

Markovin ketju Monte Carlo -menetelmien diagnostiikka

TURUN YLIOPISTO
Tietotekniikan laitos
LuK-tutkielma
Tietojenkäsittelytiede
Lokakuu 2025
Katja Kaulo

TURUN YLIOPISTO
Tietotekniikan laitos

KATJA KAULO: Markovin ketju Monte Carlo -menetelmien diagnostiikka

LuK-tutkielma, 27 s.
Tietojenkäsittelytiede
Lokakuu 2025

Markovin ketju Monte Carlo -menetelmillä eli MCMC-menetelmillä simuloidaan näytteitä jakaumasta, ja näin saadaan näytejono, jossa kukin näyte riippuu edellisen näytteen arvosta. Menetelmiä voidaan hyödyntää esimerkiksi Bayes-päätelyssä, kun posteriorijakauma on liian monimutkainen selvittää muilla keinoin. Tarkoituksena on saada aikaan jakauma, joka konvergoi eli suppenee kohti kohdejakaumaa. Konvergenssin saavuttaminen tarkoittaa sitä, että näytteillä kerätty jakauma edustaa kiinnostuksen kohteena olevaa jakaumaa. Tämän onnistumista voidaan selvittää konvergenssidiagnostiikoilla. Diagnostiikkamenetelmistä voidaan mainita esimerkiksi \hat{R} -diagnostiikka (eli Gelmanin-Rubinin diagnostiikka), tehollinen otoskoko (ESS) ja jälkikuvio (trace plot).

Tämä tutkielma on muodoltaan kirjallisuuskatsaus. Sen tutkimuskysymyksinä ovat: onko konvergenssidiagnostiikka ratkaistu ongelma ja mitä konvergenssidiagnostiikan kehittämisessä on ajankohtaista 2020-luvulla. Tutkimusaineistona on yhdeksän tutkimusartikkelia.

Tutkimusartikkelien perusteella konvergenssidiagnostiikkoja tutkitaan edelleen eikä kaikkia konvergenssidiagnostiikan ongelmia ole ratkaistu. Viime vuosina on paranneltu aiemmin olemassa olleita menetelmiä, mutta myös kehitelty uusia diagnostiikkoja. Esimerkkinä uudesta menetelmästä, voidaan mainita R^* -diagnostiikka, jossa hyödynnetään koneoppimista. Vakiintuneemmista diagnostiikoista erityisesti \hat{R} -diagnostiikkaa tutkitaan edelleen, ja siitä onkin kehitetty lähivuosina uusia versioita kuten folded-split- \hat{R} ja nested \hat{R} . \hat{R} -menetelmän uusien versioiden lisäksi artikkeleissa käsitellään diagnostiikkoja, joita on kehitelty rajatumpiin tilanteisiin kuten bayesiläiseen klusterointiin ja bayesiläisiin hermoverkkoihin. Myös graafisia diagnostiikkoja on kehitelty. On mahdollista, että tulevaisuudessa koneoppimista hyödyntävät menetelmät valtaavat alaa myös konvergenssidiagnostiikoissa.

Asiasanat: konvergenssidiagnostiikka, Markovin ketju Monte Carlo

Sisällys

1	Johdanto	1
2	Markovin ketju Monte Carlo	3
2.1	Bayesiläinen näkökulma	3
2.2	MCMC-menetelmät	5
2.3	Kohdejakauman saavuttaminen	8
2.4	Konvergenssidiagnostiikkamenetelmät	10
3	Diagnostiikan kehittelyn nykysuuntauksia	13
3.1	Gelmanin-Rubinin eli \hat{R} -diagnostiikka	15
3.2	Muut diagnostiikat	18
3.3	Artikkelien vertailu	22
4	Yhteenveto	25
	Lähdeluettelo	28

Kuvat

1.1	Tiedonhakuprosessi.	2
2.1	MCMC-menetelmän eteneminen.	9
2.2	Jälkikuvio (engl. trace plot).	11

Taulukot

3.1 Tutkimusartikkelit	14
----------------------------------	----

1 Johdanto

Markovin ketju Monte Carlo eli MCMC on ollut merkittävä kehitysaskel bayesiläisessä päättelyssä. Pelkkä teorian ja menetelmien kehittäminen ei ole yksistään ollut ratkaisevaa, vaan lisäksi on tarvittu tietokoneiden, laskentatehon ja ohjelmistojen kehitystä. MCMC on tärkeä väline, joka on mahdollistanut monenlaisten ongelmien ratkaisemisen. [1, s. 3–4] [2] Kuten minkä tahansa työvälineen myös sen on toimittava, ja tämän toimimisen tarkastaminen ei ole täysin ongelmatonta (esim. [1, s. 21]).

Tutkielmani on muodoltaan kirjallisuuskatsaus ja käsittelee MCMC-menetelmien konvergenssidiagnostiikkaa eli tapoja tutkia toimiiko käytetty MCMC-menetelmä. Tutkimuskysymyksinäni ovat:

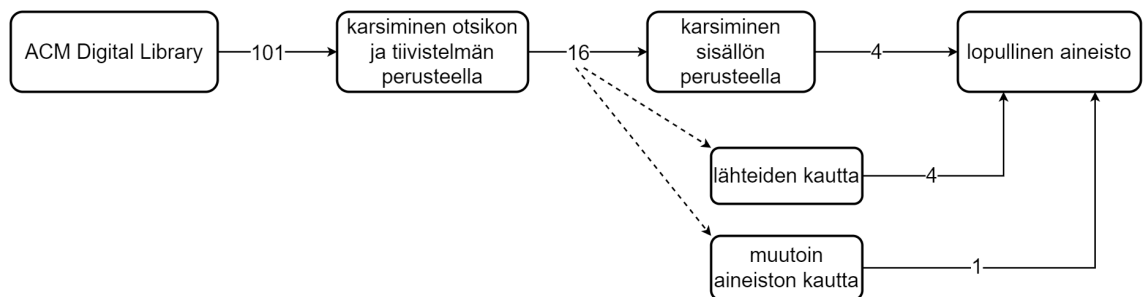
TK1 Onko konvergenssidiagnostiikka ratkaistu ongelma?

TK2 Mitä konvergenssidiagnostiikan kehittämisessä on ajankohtaista 2020-luvulla?

Tarkoitukseni on siis tarkastella, kehitelläänkö konvergenssimenetelmiä tällä hetkellä, ja jos näin on, niin ovatko jotkut tietyt konvergenssidiagnostiikat tutkimuksen keskiössä vai onko kyse täysin uusista diagnostikoista. Tavoitteenani ei ole kovin syvälinen ja tarkka selvitys vaan enemmän pääpiirteinen ja yleisluontoinen katsaus siihen, millaisia suuntaviivoja konvergenssidiagnostiikan tutkimuksessa ja kehittämisessä on tällä hetkellä havaittavissa.

Tutkielmani tiedonhaussa hakulausekkeenani oli "convergence diagnostics" AND ("markov chain monte carlo" OR mcmc). Lisäksi käytin aikarajausta

1.1.2020–28.2.2025. Tiedonhaun tein ACM Digital Library -tietokannasta käyttäen laajempaa hakua *The ACM Guide to Computing Literature*. Sain 101 tulosta, joista karsin otsikon ja tiivistelmän perusteella pois tutkimukseni kannalta epäoleelliset. Näin löytämistäni 16 artikkelista valitsin lopulta neljä artikkelia tutkimukseeni. Täydensin tätä aineistoa myös tutkimusten lähdeluetteloista löytämilläni rajaukseen sopivilla tutkimuksilla. Lisäksi löysin yhden artikkelin, sillä siinä oli käytetty lähteenä erästä tarkasteluun löytämäni artikkelia. Näin pääaineistoni koostuu yhteensä yhdeksästä artikkelista. Tiedonhakuprosessi on esitettyä kuvassa 1.1.



Kuva 1.1: Tiedonhakuprosessi. Yhteensä 101 hakutuloksesta lopulliseksi aineistoksi seuloutui ja löytyi yhdeksän artikkelia.

Tutkielman rakenne on seuraavanlainen: Luvussa 2 on taustatietoa kirjallisuuskatsaustani varten. Kerron siinä MCMC:n toiminnasta, siihen liittyvistä käsitteistä ja siitä, miten voidaan selvittää, toimiiko MCMC toivotulla tavalla. Luvussa 3 käsittelemän tutkimusaineistoani. Pyrin siinä nostamaan esiin tutkimuskysymysteni kannalta oleellisia asioita. Luvussa 4 kokoan vastaukset tutkimuskysymyksiini käsittelemäni aineiston perusteella ja pohdin konvergenssidiagnostiikan tulevaisuutta.

Käytän tutkielmassani englannin kielen sanasta *Bayesian* suomenkielistä sanaa *bayesiläinen*, jota suositellaan Tilastotieteen sanastossa [3, s. 12]. Myös muissa käsitteissä pyrin käyttämään mahdollisuuksien mukaan Tilastotieteen sanastossa esitettyjä nimityksiä. Käytän suppenemisesta termiä *konvergenssi*, suppenemisen tarkastelusta *konvergenssin diagnosoiminen* ja menetelmistä tarkastella suppenemistä (*konvergenssi*)*diagnostiikat*.

2 Markovin ketju Monte Carlo

Tässä luvussa selvennän tutkimukseni kannalta oleellisia käsitteitä. Alaluvussa 2.1 kerron bayesiläisestä päättelystä, jossa MCMC-menetelmät ovat erittäin hyödyllisiä monimutkaisten posteriorijakaumien selvittämisessä, vaikkakin niitä voidaan käyttää myös bayesiläiseen päättelyn ulkopuolella, kuten optimoinnissa. Tämän jälkeen kerron alaluvussa 2.2 MCMC-menetelmistä aloittaen lyhyesti Markovin ketjuista ja Monte Carlo -menetelmästä. MCMC-menetelmissä simuloidaan näytteitä kohdejakaumasta, kuten esimerkiksi kiinnostuksen kohteena olevasta posteriorijakaumasta. Käytän tästä esimerkkinä Metropolisin algoritmia ja sen kautta nykyään yleisempää Metropolisin-Hastingsin algoritmia. Algoritmin lopputuloksena on kohdejakaumaa kohti suppeneva eli konvergoituva jakauma. Alaluvussa 2.3 kerron konvergenssista ja lämmittelyjaksosta, joka liittyy konvergenssiin, sekä stationaarisuudesta ja sekoittumisesta ennen siirtymistä käsittelemään konvergenssidiagnostiikan menetelmiä alaluvussa 2.4. Diagnostiikat ovat tärkeitä, sillä niillä tarkkaillaan konvergoituvatko MCMC-menetelmällä saadut ketjut kohdejakaumaa kohti.

2.1 Bayesiläinen näkökulma

Bayesiläisessä näkökulmassa keskeisessä osassa on Bayesin kaava, joka tunnetaan myös esimerkiksi nimellä Bayesin teoreema (engl. esim. Bayes' theorem, Bayes' rule,

Bayes' formula) [3, s. 12] [4, s. 7]:

$$p(\theta|y) = \frac{p(\theta, y)}{p(y)} = \frac{p(\theta)p(y|\theta)}{p(y)}. \quad (2.1)$$

Kaavalla 2.1 lasketaan *posteriorijakauma* (engl. posterior density). Ensimmäisessä vaiheessa osoittajassa on yhteisjakauma $p(\theta, y)$, joka on priorijakauman $p(\theta)$ ja havaintojen jakauman $p(y|\theta)$ tulo, kuten kaavan jälkimmäisessä osassa voidaan nähdä. Nimittäjässä esiintyvä $p(y)$ on reunajakauma (engl. marginal distribution), josta voidaan tässä tilanteessa käyttää kuvaavampaa nimitystä *prioriprediktiiivinen jakauma* (engl. prior predictive distribution). Se on tuntemattoman, mutta havaittavan y :n jakauma. Kyseessä on diskreetissä tapauksessa summa yli kaikkien θ :n arvojen eli $p(y) = \sum_{\theta} p(\theta)p(y|\theta)$ ja jatkuvassa tapauksessa θ :n todennäköisyystiheys eli $p(y) = \int p(\theta)p(y|\theta)d\theta$. [4, s. 6–7] Posterioria voidaan tarkastella myös ilman reunajakaumaa:

$$p(\theta|y) \propto p(\theta)p(y|\theta) \quad (2.2)$$

Kaavassa 2.2 oikealla puolella on posteriorijakauma ilman normalisoivaa tekijää (engl. unnormalized posterior density). Normalisoiva tekijä tässä on $p(y)$. Se on pois jätetty vakiokerroin, sillä se ei riipu θ :sta ja havainnot y ovat kiinnitettyinä. Kun $p(y|\theta)$ nähdään parametrin θ funktiona, kutsutaan sitä nimellä uskottavuusfunktio (engl. likelihood function). Kaavan 2.2 tulontekijöinä ovat siis priorijakauma (eli ennakkotiedot asiasta) ja havaintojen jakauma (eli uskottavuusfunktio, joka kertoo kuinka todennäköisiä havainnot y ovat eri θ :n arvoilla). Lopputuloksena on posteriorijakauma, jolla arvioidaan θ :n epävarmuutta, kun havainnot y on havaittu. Kyse on siis ennakkotietojen ja asiantuntemuksen päivittämisestä uudella datalla ja epävarmuuden huomioimisesta. [4, s. 7] [5, s. 21–22]

Bayesiläisessä päättelyssä voi monesti tulla vastaan tilanteita, joissa tarkkaa pos-

teriorijakaumaa ei pystytä laskemaan johtuen siitä, että ne voivat olla monimutkaisia laskea. Esimerkiksi hierarkkisten mallien posteriorijakaumat voivat olla ongelmallisia. MCMC-menetelmät ovat yksi ratkaisu tällaisiin tilanteisiin. Tarkan tuloksen ja suljetun muodon sijaan turvaudutaankin approksimointiin ja simulointiin. [2, s. 110] [4, s. 261–262]

2.2 MCMC-menetelmät

MCMC-menetelmissä yhdistyy kaksi osaa: Markovin ketju (engl. Markov Chain) ja Monte Carlo -menetelmä. Markovin ketju on stokastisen prosessin erikoistapaus. Se on jono, jossa alkiot ovat satunnaismuuttujia ja kukin alkio riippuu vain juuri sitä edeltävästä alkioista. Näitä alkioita kutsutaan tiloiksi, ja kaikkien mahdollisten tilojen joukkoa tila-avaruudeksi (engl. state space). Tila-avaruus voi olla diskreetti tai jatkuva. Pääosa MCMC-menetelmien tila-avaruuksista on jatkuvia. [1, s. 4–5]

Monte Carlo -menetelmä on hyödyllinen tilanteessa, jonka selvittäminen on laskennallisesti liian raskasta. Suoranaisen laskemisen sijaan simuloidaan näytteitä, joista voidaan laskea vaikkapa odotusarvolle estimaatti. Monte Carlo -menetelmässä simulaation näytekoko on oleellinen. Esimerkiksi tarkkuutta voidaan parantaa kymmenkertaiseksi nostamalla Monte Carlo -näytekoko satakertaiseksi. [1, s. 6–7]

Nämä MC (Markov chain) ja MC (Monte Carlo) yhdistämällä siis saadaan MCMC eli *Markovin ketju Monte Carlo*. Näytteet otetaan järjestyksessä perä perää ja seuraava näyte riippuu vain sitä aiemman arvosta. Näin syntyy ketju näytteitä, jotka lopulta edustavat kiinnostuksen kohteena olevaa jakaumaa. Käytän tästä seuraavaksi esimerkkinä Metropolisin algoritmia (engl. Metropolis algorithm). [4, s. 275]

Metropolisin algoritmi

Käytän seuraavassa Metropolisin algoritmin kuvailussa lähteenä teoksen *Bayesian Data Analysis* lukua 11 *Basics of Markov chain simulation* [4, s. 275–292]. Seuraan myös sen merkintätapaa. Muut käytetyt lähteet on merkitty erikseen.

Metropolisin algoritmi etenee seuraavalla tavalla:

1. Valitaan aloituspiste θ^0 aloitusjakaumasta $p_0(\theta)$.
2. Jokaiselle $t = 1, 2, \dots$
 - (a) Poimitaan ehdokaspiste θ^* ehdokasjakaumasta $J_t(\theta^*|\theta^{t-1})$.
 - (b) Lasketaan tiheyksien suhdeluku r :

$$r = \frac{p(\theta^*|y)}{p(\theta^{t-1}|y)}. \quad (2.3)$$

- (c) Asetetaan

$$\theta^t = \begin{cases} \theta^*, & \text{todennäköisyydellä } \min(r, 1) \\ \theta^{t-1}, & \text{muulloin.} \end{cases} \quad (2.4)$$

Metropolisin algoritmin suoritus alkaa aloituspisteen valinnalla. Valinta tehdään aloitusjakaumasta (engl. starting distribution) $p_0(\theta)$. Aloituspisteelle tulee päteä $p(\theta^0|y) > 0$, eli sen tulee olla mahdollinen aineistolla y . Algoritmin suoritus jatkuu tämän jälkeen ehdokaspisteen valinnalla (a), suhdeluvun r laskemisella (b) ja ehdokaspisteen hyväksymisellä tai hylkäämisellä, eli seuraavan pisteen valitsemisella Markovin ketjuun (c). Vaihe 2 toistuu algoritmin suorituksessa useita kertoja.

Ehdokaspiste (engl. candidate sample) poimitaan ehdokasjakaumasta (engl. jumping distribution, proposal distribution, candidate distribution). Metropolisin algoritmissa ehdotusjakauma on symmetrinen eli $J_t(\theta_a|\theta_b) = J_t(\theta_b|\theta_a)$ kaikille arvoille θ_a , θ_b ja t . Tämä tarkoittaa sitä, että todennäköisyyden siirtyä pisteestä θ_a pisteeseen θ_b on oltava sama kuin todennäköisyyden siirtyä pisteestä θ_b pisteeseen θ_a .

Metropolisin-Hastingsin algoritmille ehdotusjakauma voi olla epäsymmetrinen. Se on siis yleistys Metropolisin algoritmista epäsymmetrisille jakaumille.

Kaavan 2.3 ehdokaspisteen ja nykyisen pisteen tiheyksien suhdelukua r voidaan kutsua myös nimityksellä *Metropolis ratio* [1, s. 24], jonka voisi suomentaa Metropolisin suhdeluvuksi. Suhdeluku on toinen eroavaisuus Metropolisin ja Metropolisin-Hastingsin algoritmien välillä. Metropolisin-Hastingsin algoritmin suhdelukua (*Hastings ratio* [1, s. 26]) laskettaessa otetaan huomioon ehdokasjakauman epäsymmetrisyys:

$$r = \frac{p(\theta^*|y)/J_t(\theta^*|\theta^{t-1})}{p(\theta^{t-1}|y)/J_t(\theta^{t-1}|\theta^*)}. \quad (2.5)$$

Ehdokaspisteen hyväksymistodennäköisyyttä $\min(r, 1)$ verrataan tasajakautuneeseen satunnaislukuun u väliltä $(0, 1)$. Ehdotettuun pisteeseen siis siirrytään, jos $\min(r, 1) > u$. Mikäli siis tiheys ehdokaspisteessä on suurempi kuin nykyisessä pisteessä, siirrytään ehdokaspisteeseen todennäköisyydellä 1. [5, s. 538] Tästä voidaankin huomata, että u :n satunnaisesta arvosta johtuen ei ehdokaspistettä välttämättä silti hyväksytä, vaikka sen todennäköisyys olisi varsin korkea. Samoin osa ehdokaspisteistä, joiden hyväksymistodennäköisyys on pieni, saatetaan tämän satunnaisuuden vuoksi hyväksyäkin.

Hyväksytystä ehdokaspisteestä tulee uusi piste Markovin ketjussa, ja algoritmin suoritusta jatketaan kohdasta 2. Jos ehdokaspiste hylätään, pysytään nykyisessä pisteessä, ja tämä lasketaan yhdeksi iteraatioksi. Piste on siis näytteissä peräkkäin kahdesti, jos ehdokaspiste hylätään, tai jopa useampaankin kertaan, jos hylkääminen sattuu tapahtumaan moneen kertaan peräkkäin [5, s. 538].

Algoritmin suoritus jatkuu aina viimeisimmästä hyväksytystä pisteestä: joko uudesta hyväksytystä pisteestä tai pisteestä, jossa vierailaan toistamiseen. Näin syntyy ketju, jossa kukin piste riippuu vain sitä aiemmasta pisteestä. Kyseessä on, kuten todettua, todellakin Markovin ketju. Metropolisin algoritmin toiminnasta johtuen

tässä ketjussa on enemmän pisteitä alueilta, joiden tiheys on suurempi. Tällaisiin pisteisiin siirrytään todennäköisemmin, ja niissä pysytään todennäköisemmin pidempään. Kuitenkin myös niiden ympäristöstä etsitään uusia pisteitä, joiden ympäristöstä voi taas löytyä tiheydeltään entistä suurempia pisteitä. Lopulta muodostuva jakauma konvergoituu kohti kohdejakaumaa (engl. target distribution).

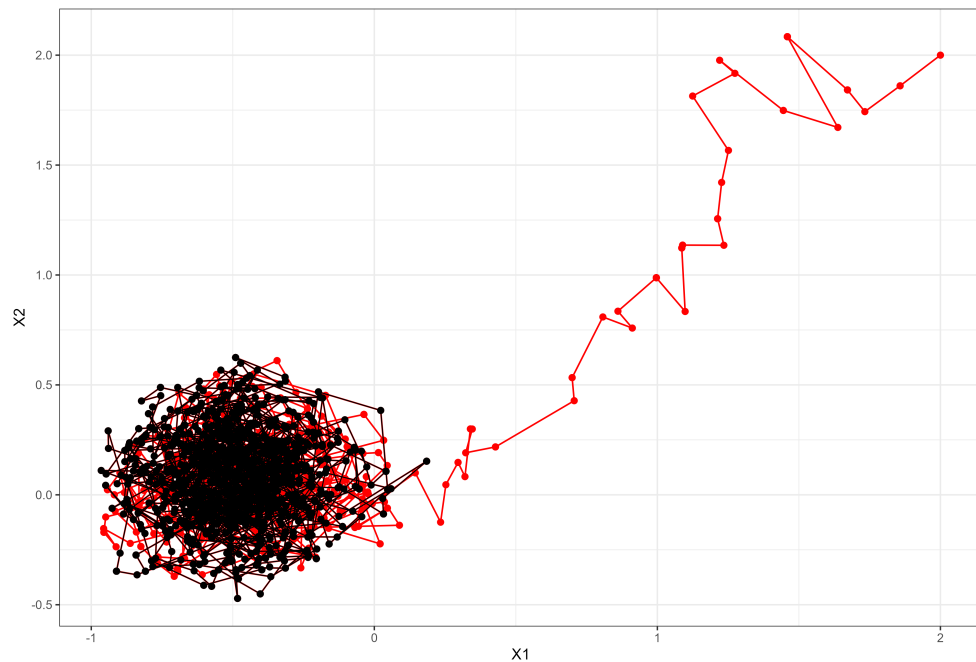
2.3 Kohdejakauman saavuttaminen

Konvergenssilla (engl. convergence) tarkoitetaan MCMC-simulaatiolla saatavan approksimoivan jakauman suppenemista kohti kohdejakaumaa [4, s. 275]. Jos konvergenssi saavutetaan, edustaa näytteillä kerätty jakauma siis sitä todellista jakaumaa, jota halutaan tarkastella, esimerkiksi posteriorijakaumaa.

MCMC-menetelmien käytössä on haasteensa. Jos iteraatioita ei ole tarpeeksi monta, ei saatu jakauma edusta kohdejakaumaa. Lisäksi ongelmana on, että vaikka konvergenssi saavutettaisiinkin, niin alun iteraatioiden näytteet eivät kuvaa kohdejakaumaa vaan niissä näkyy aloituspisteen vaikutus. Näihin ongelmakohtiin on kehitetty erilaisia ratkaisuja, kuten esimerkiksi lämmittelyjakso. [4, s. 281–282]

Lämmittelyjakso (engl. warm-up period, aiemmin burn-in period) tarkoittaa käytäntöä, jossa varhaisia MCMC-menetelmän iteraatioita hylätään, sillä ne eivät vielä kuvasta kohdejakaumaa. Hylättyjen iteraatioiden määrä riippuu monesta tekijästä, muun muassa käytetystä menetelmästä. Esimerkiksi ensimmäinen puolikas iteraatioista voidaan hylätä. [4, s. 282] Kuvassa 2.1 lämmittelyjakso on esitettynä punaisella. Siinä on nähtävissä algoritmin eteneminen kohti tavoitejakaumaa.

Kaikki eivät välttämättä suosittelen ensimmäisen iteraatiopuolikkaan jättämistä käyttämättä, sillä sen voidaan ajatella haaskaavan liikaa näytteitä (esim. [6, s. 394]). Geyer [1, s. 20] toteaa, että lämmittelyjakso voidaan nähdä vain jokseenkin kehnona menetelmänä löytää aloituspiste MCMC-iteraatioille. Hänen mukaansa se on vakiintunut käytäntö mahdollisesti sen vuoksi, että se on enimmäkseen harmiton. Vuonna



Kuva 2.1: MCMC-menetelmän eteneminen. Otetut näytteet ovat kuvassa pisteinä. Alun näytteet eivät edusta hyvin kohdejakaumaa, ja siksi ne hylätäänkin. Tämä ns. lämmittelyjakso on kuvattuna punaisella. Lämmittelyjakson jälkeiset näytteet, joita voidaan hyödyntää, on merkitty mustalla.

1992 Geyer [7, s. 480] on arvellut 1–2 % alun iteraatioista riittävän lämmittelyjakson pituudeksi rutiininomaisesti ja että tätä pidemmän lämmittelyjakson tarve voidaan saada selville autokovarianssin avulla.

MCMC-menetelmien hyödyllisyys nojaa siihen, että kiinnostuksen kohteena olevaa jakaumaa edustavia näytteitä saadaan tarkasteluun. Poistamalla lämmittelyjakson aikaiset näytteet ollaan jo (mahdollisesti) paremmassa tilanteessa [4, s. 282]. Gelmanin ym. [4, s. 276] mukaan tarvitaan ehdottomasti myös konvergenssin tarkistamista.

Konvergenssiin liittyy kaksi oleellista tarkastelukohdetta: *sekoittuminen* (engl. mixing) ja *stationaarisuus* (engl. stationarity). Sekoittuminen käsittelee sitä, kuinka hyvin ja nopeasti ketju liikkuu tila-avaruudessa eli käy läpi mahdollisten tilojen alueen [6, s. 396]. Stationaarisuus taas tarkoittaa sitä, että ketju löytää lopulta haetun jakauman eikä muutu, vaikka uusia näytteitä otettaisiin [4, s. 284] [1, s. 11].

Jos konvergenssi on saavutettu, on näytejono myös stationaarinen ja hyvin sekoittunut. Näytejono voi olla hyvin sekoittunut mutta epästationaarinen tai stationaarinen mutta huonosti sekoittunut. Tällöin ei voi olla kyse konvergoituneesta ketjusta. Näiden kahden tunnuspiirteen testaaminen on siis käyttökelpoista konvergenssin diagnosoimisessa. [4, s. 283–284, 288]

2.4 Konvergenssidiagnostiikkamenetelmät

Käytän tässä alaluvussa päälähteenäni Vivekananda Royn artikkelia *Diagnostics for Markov Chain Monte Carlo* vuodelta 2020 [6]. Keskityn erityisesti konvergenssin tarkasteluun stationaarisuuden ja sekoittumisen kautta ja sivuutan lämmittelyjakson pituuden ja pysäyttämissäännöt. Selitän diagnostiikoista tarkemmin ne Royn käsittelemät menetelmät, joiden arvelen olevan yleisimmin käytössä.

Gelmanin-Rubinin diagnostiikka (GR-diagnostiikka) lienee yksi suosituimmista tavoista diagnosoida MCMC-menetelmien konvergenssia. Sen tuloksena on yksittäinen lukuarvo \hat{R} , jolla voidaan arvioida konvergenssia.

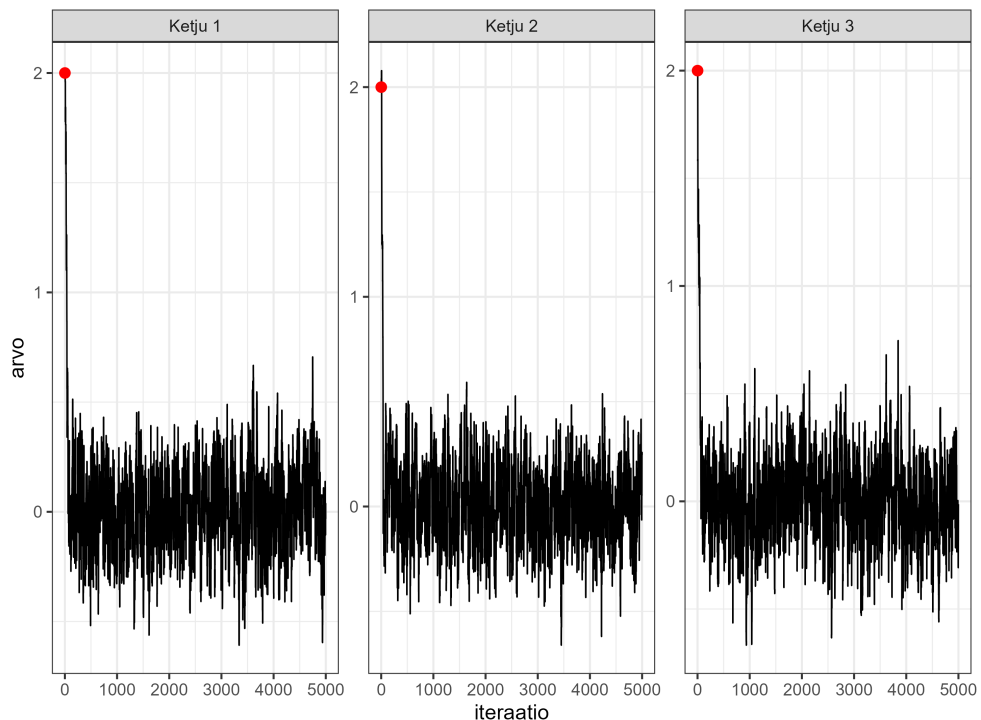
Yksinkertaistettuna sen toimintaperiaate etenee seuraavalla tavalla. Diagnostiikassa käytetään useita Markovin ketjuja. Näiden konvergoitumista tarkastellaan käyttäen ketjun sisäisen varianssin estimaattia W ja ketjujen yhdistetyn varianssin (engl. pooled variance) estimaattia \hat{V} . Näiden välistä suhdetta käytetään arvioimaan konvergoitumista. [6, 8]

$$\hat{R} = \frac{\hat{V}}{W} \tag{2.6}$$

Kaavalla 2.6 lasketaan siis suhdeluku \hat{R} (engl. the potential scale reduction factor, PSRF), joka pienenee, kun ketjujen välinen varianssi \hat{V} lähestyy ketjujen sisäisiä variansseja W . Tätä suhdetta verrataan lukuun yksi. MCMC-menetelmällä simuloitujen ketjujen todetaan konvergoituneen (tai joissain tapauksissa simulaatio

pysäytetään), kun \hat{R} on tarpeeksi lähellä sitä. [6, 8] Yleisesti käytettynä rajana on 1.1, jonka alle siis lukuarvon \hat{R} tulisi olla [4, s. 287] [6].

Useamman muuttujan tapauksiin on olemassa monimuuttujainen \hat{R} (engl. the multivariate PSRF, MPSRF), jota merkitään \hat{R}_p . Myös \hat{R}_p -suhdelukua verrataan lukuarvoon yksi. [9]



Kuva 2.2: Jälkikuvio (engl. trace plot). Kuvassa on kolmen eri ketjun jälkikuvioita. Jälkikuvio on graafinen diagnostiikkamenetelmä MCMC-menetelmällä tuotetun ketjun konvergoitumisen tutkimiseen.

Trace plot on suosituin graafisista diagnostiikkamenetelmistä, ja sen voisi suomentaa esimerkiksi jälkikuvioiksi. Sen avulla voidaan tutkia silmämääräisesti ketjun sekoittumista ja stationaarisuutta. Jälkikuvio on aikasarjakuvio, joka näyttää Markovin ketjun realisaatiot yhdistettynä kuhunkin iteraatioon. Tavoitteena on saada aikaan "karvaisen toukan" näköinen kuvio. Kuvion alussa näkyy aloituspisteen vaikutus. Jos kuvio on jo alusta saakka samanlainen kuin myöhemminkin, ei lämmittelyjaksolle ole välttämättä tarvetta. Kuvassa 2.2 on nähtävissä esimerkki jälkikuvioista

kolmessa eri ketjussa. Kuvassa aloituspiste on merkittynä punaisella. Vaaka-akselilla on näkyvissä iteraationumero ja pystyakselilla Markovin ketjun realisaatiot.

Tehollinen otoskoko (ESS, engl. effective sample size) on myös varsin yleinen diagnostiikka. Se lasketaan yksittäisestä ketjusta [10, s. 690–691]. Mitä suurempi lukuarvo saadaan, sitä tehokkaampaa näytteistys on, sillä näytteistä suurempi osa on riippumattomia. Tehollisesta otoskoosta on olemassa erilaisia laskutapoja, mutta yleisin on näkyvissä kaavassa 2.7 ([11] viitattu lähteessä [6]).

$$ESS = \frac{n}{1 + 2 \sum_{i=1}^{\infty} \text{Corr}_{\pi}(g(X_0), g(X_i))} , \quad (2.7)$$

jossa n on ketjun näytteiden määrä, joka jaetaan viipeiden¹ autokorrelaatiolla [10, s. 690–691].

Näiden lisäksi Roy [6] nostaa esiin muutamia muita konvergenssidiagnostiikkoja, joita en esittele tarkemmin tutkielmassani. Graafisista menetelmistä hän mainitsee autokorrelaatiokuvion (engl. autocorrelation plot) ja spektritiheyteen pohjautuvista menetelmistä Geweken konvergenssidiagnostiikan ja Heidelbergin ja Welchin konvergenssidiagnostiikan. Myös Raftery-Lewisin diagnostiikka ja ydintiheyteen pohjautuvat menetelmät käsitellään. Näiden lisäksi on olemassa vielä muita diagnostisia menetelmiä, kuten pysäyttämissääntöjä.

¹suom. [3, s. 78]

3 Diagnostiikan kehittelyn nykysuuntauksia

Tässä luvussa käsittelen kirjallisuuskatsaukseni tutkimusaineistoa. Näkökulmani on tutkia, kehitelläänkö konvergenssidiagnostiikkaa tällä hetkellä ja, jos kehitellään, mitkä konvergenssidiagnostiikat ovat pinnalla. Kiinnostavaa on myös, parannellaanko vanhoja menetelmiä vai onko peräti kehitelty uusia.

Tutkielmani tutkimusaineisto on esiteltynä taulukossa 3.1 tiivistetysti. Käsittelen tässä luvussa siis yhdeksää artikkelia, joiden aiheena on konvergenssidiagnostiikka ja joista pyrin löytämään vastauksia tutkimuskysymyksiini. Artikkelit ovat aikaväliltä 1.1.2020–28.2.2025. Aineistonhankinnan ja sen rajauksen olen käsitellyt luvussa 1.

Kuten mainitsin alaluvussa 2.4, Roy [6] käyttää Gelmanin ja Rubinin kehittämästä diagnostiikasta nimeä Gelmanin-Rubinin diagnostiikka ja GR-diagnostiikka sekä diagnostiikan lukuarvosta nimitystä \hat{R} ja kirjainlyhennettä PSRF (the potential scale reduction factor). Käsittelemissäni artikkeleissa kyseiseen diagnostiikkaan viitattiin pääosin \hat{R} :lla. Muutamissa oli käytössä myös \hat{R} -diagnostiikka tai eri \hat{R} -versioista \hat{R} -diagnostiikat [12, 17, 19]. Selkeyden ja yksinkertaisuuden vuoksi käytän myös tässä luvussa kyseisestä diagnostiikasta nimeä \hat{R} -diagnostiikka.

Taulukossa 3.1 on nähtävissä tiivistettynä, mitä ongelmia artikkelien tutkimuksissa on pyritty ratkaisemaan tai mihin tilanteisiin on yritetty kehitellä toimivam-

Taulukko 3.1: Tutkimusartikkelit

Artikkeli	Käsiteltävät ongelmat ja tilanteet	Ratkaisuehdotukset
Vehtari ym. 2021 [12]	Split- \hat{R} :n ja ESS:n ongelmia.	Järjestyslukunormalisoidut bulk-ESS ja split- \hat{R} , folded-split- \hat{R} , tail-ESS sekä graafiset (Rank plots).
Lambert ja Vehtari 2022 [13]	\hat{R} :n rajoitteet.	R^* (hyödyntää ML).
Moins ym. 2025 [14]	\hat{R} :n rajoitteet.	\hat{R}_∞ ($\hat{R}(x)$ n pienin yläraja).
Margossian ym. 2024 [15]	Rinnakkaisen MCMC:n ja lyhyiden ketjujen diagnostiikka.	\hat{R}_v , eli nested \hat{R} , jossa vertailaan ketjuklustereita.
Lysy ym. 2021 [16]	Diagnostiikka bayesiläisessä klusteroinnissa.	Yhden ketjun vaativa konvergenssidiagnostiikka.
Säilynoja ym. 2022 [17]	Diagnostiikat, jotka antavat yhden lukuarvon, mutta eivät kerro ketjujen välisten erojen luonteesta.	Graafinen testi lisädiagnostiikkana (täydentämään esim. \hat{R} tai R^*).
Sommer ym. 2024 [18]	Diagnostiikka bayesiläisissä hermostoverkoissa.	$\hat{R}^{(\kappa)}$ ja $\widehat{cR}^{(\kappa)}$ (κ kpl aliketjuja). Parametriavaruuden konvergenssi ketjuittain ja kerroksittain.
Cooper ym. 2024 [19]	Rinnakkaisen MCMC:n lämmitelyjakso ja diagnostiikka.	Lämmittelyjakson uudelleenkäyttö. Yksinkertaistukset diagnostiikkaan (ESS ja \hat{R}).
Duttweiler ym. 2024 [20]	Poiminta-avaruus diskretisoitu, epäeuklidiset poimijan siirtymät, näytteiden monimutkaisuus ja moniulotteisuus.	Näytteiden esikäsittely reaaliarvoisiksi ja standardidiagnostiikat (Trace plot, ESS ja \hat{R}).

paa konvergenssidiagnostiikkaa sekä ehdotetut ratkaisut ja menetelmät. Taulukossa esiintyy varsin usein \hat{R} -diagnostiikka, ja aloitan käsittelemällä näitä artikkeleita.

3.1 Gelmanin-Rubinin eli \hat{R} -diagnostiikka

Vehtari ym. [12] käsittelevät artikkelissaan *Rank-Normalization, Folding, and Localization: An Improved \hat{R} for Assessing Convergence of MCMC (with Discussion)* \hat{R} -menetelmän ja tehollisen otoskoon ongelmia. Artikkelissa käytettynä \hat{R} -versiona on split- \hat{R} , jossa ketju katkaistaan nimensä mukaisesti kahtia. Ketjun alkuosaa verrataan ketjun loppuosaan, ja pyritään näin paljastamaan konvergenssin puutteellisuus kyseisen ketjun sisällä. Tämä tehdään erikseen kullekin ketjulle.

Artikkelissa käytetään \hat{R} :n rajana lukuarvoa 1.01 aiemmin yleisempänä olleen 1.1 sijaan, sillä joissain tapauksissa \hat{R} voi laskea arvon 1.1 alle ennen konvergenssia. Kun rajana käytetään lukua 1.01 tapahtuu tätä huomattavasti harvemmin. Lisäksi vaikka split- \hat{R} -diagnostiikka mahdollistaisi diagnosoinnin vain yhdestä ketjusta (kun ollaan varmoja yhden ketjun riittävydestä diagnosointiin), artikkelissa suositellaan käytettäväksi vähintään neljää ketjua, sillä se esimerkiksi paljastaa helpommin huonon sekoittumisen ja toimii paremmin monihuippuisilla jakaumilla. Laskenta-aikakaan ei kasva, jos ketjut ajetaan rinnakkain. [12]

Vaikka split- \hat{R} -menetelmä onkin parannus \hat{R} -diagnostiikkaan, on sillä siis silti edelleen tiettyjä rajoituksia, joihin artikkelissa pyritään löytämään ratkaisuja. Ongelmallisimmat tilanteet split- \hat{R} -diagnostiikalle ovat, kun kohdejakauman keskiarvo tai varianssi ovat äärettömiä tai kun ketjun varianssi on äärellinen, mutta jakauma on hyvin paksuhäntäinen tai ketjuilla on samat keskiarvoparametrit, mutta varianssit eroavat. Ratkaisuksi artikkelissa ehdotetaan järjestyslukuihin pohjautuvaa diagnostiikkaa. Arvon sijaan käytetään siis sen järjestyslukua ja normalisoidaan arvot, jolloin saadaan järjestysnormalisoidut (engl. rank-normalized) bulk-ESS ja split- \hat{R} . Järjestyslukunormalisoinnin lisäksi voidaan käyttää järjestyslukunormalisoitua it-

seisarvoa näytteen erotuksesta ketjun mediaanista. Tällöin saadaan folded-split- \hat{R} , joka on hyödyllinen diagnosoitaessa ketjun jakauman häntiä. Jotta saataisiin yksi konservatiivinen \hat{R} -estimaatti, voidaan raportoida suurempi järjestysnormalisoidusta split- \hat{R} :stä ja folded-split- \hat{R} :stä. Hännistä ehdotetaan laskettavaksi tail-ESS, joka saadaan laskemalla tehollinen otoskoko 5 %:n ja 95 %:n kvantiileissa ja valitseamalla pienempi luku. Näin saadaan mahdollisesti selvitettyä ongelmia, jotka johtuvat arvojen vaihtelusta, jotka eroavat ketjujen välillä. [12]

Vehtari ym. [12] ehdottavat konvergenssin visuaaliseen tarkasteluun kullekin ketjulle järjestyslukuihin pohjautuvaa histogrammia, *rank plot*, jälkikuvioiden sijaan. Samasta jakaumasta olevien ketjujen histogrammien oletetaan olevan yhdenmukaisia. Toisiinsa nähden samanlaiset kuviot kertovat siis hyvästä sekoittumisesta. Jos jokin ketju eroaa sijainti- tai paikkaparametriltaan muista, myös sen histogrammi on erilainen. Toisena visuaalisena keinona tarkastella konvergenssia on kvantiili- ja pienten intervallien kuvio (engl. quantile and small-interval plot). Tästä kuviosta voidaan havaita alueet, joita ei ole tutkittu kunnolla. Kolmantena visuaalisena keinona mainitaan tehokkuus per iteraatio -kuvio (engl. efficiency per iteration plot). Koska tehollisen otoskoon pitäisi kasvaa kokonaisotoskoon kasvaessa, kuviossa tulisi näkyä lineaarista kasvua.

Lambert ja Vehtari [13] esittelevät artikkelissaan *R*: A Robust MCMC Convergence Diagnostic with Uncertainty Using Decision Tree Classifiers* uuden diagnostiikan R^* . Se perustuu koneoppimisluokittelijoihin, jotka on opetettu ennustamaan, mistä ketjusta havainnot ovat peräisin. Jos luokittelija selvittää hyvin, mistä ketjusta havainnot ovat peräisin, viittaa se konvergenssiongelmiin. Kun ketjut ovat sekoittuneet hyvin, ei havaintojen ketjuja onnistuta selvittämään paremmalla tarkkuudella kuin puhtaasti satunnaisella arvauksella. Toisin kuin \hat{R} -menetelmä, joka lasketaan kullekin parametrille erikseen, R^* -diagnostiikka palauttaa yksittäisen lukuarvon, joka kertoo onko konvergenssi saavutettu. R^* saadaan luokittelijan ja nollamallin en-

nustustarkkuuksien suhteesta, eli vertaamalla saatua tarkkuutta lukuun $1/N$, jossa N on käytettyjen ketjujen määrä. Näin saadaan luku, joka on samankaltainen kuin alkuperäinen \hat{R} . Kun $R^* \approx 1$, viittaa se konvergenssiin.

Lambert ja Vehtari [13] suosittelevat käyttämään R^* :n laskemisessa kahta koneoppimislukittelijaa: gradienttitehostettua regressiopuuta ja satunnaismetsää. Näillä saatujen luokittelun todennäköisyyksien kautta saadaan laskettua myös R^* :n epävarmuus. Huomioitavaa on, että koulutus- ja testiaineiston luomisen sekä mallin opettamisen satunnaisuudesta johtuen R^* -menetelmä palauttaa eri lukuarvon vaikka se ajettaisiin samoilla näytteillä (ellei ns. siemenluku (engl. seed) ole kiinnitetty). Lisäksi R^* -diagnostiikkaan ehdotetaan käytettäväksi split- \hat{R} -diagnostiikasta tuttua ketjujen puolittamista, sillä se vaikuttaisi tekevän menetelmästä tarkemman.

Moins ym. [14] käsittelevät artikkelissaan *On the Use of a Local \hat{R} to Improve MCMC Convergence Diagnostic* alkuperäisen \hat{R} -diagnostiikan rajoitteita. Rajoituksina ja ongelmakohtina mainitaan esimerkiksi kynnsarvon mielivaltaisuus ja \hat{R} -arvon tulkittavuuden puutteellisuus. Se ei myöskään ole monessa tilanteessa robusti menetelmä tai käsittele tiettyä kiinnostavaa suuretta.

Moins ym. [14] nostavat esiin Vehtarin ym. [12] pyrkimykset korjata joitain näistä puutteista. He toteavat esimerkiksi, ettei Vehtarin ym. \hat{R} :n rajana käyttämä lukuarvo 1.01 korjaa sitä, että raja ei pohjaudu teoriaan. Moins ym. huomauttavat myös, että vaikka Vehtarin ym. kehittämä rank- \hat{R} on robusti, se ei silti korjaa tulkittavuuden ongelmaa. Teoreettista arvoa R , jota \hat{R} -arvolla estimoitaisiin, ei ole tutkittu. \hat{R} :sta voidaan tehdä johtopäätöksiä, kun arvo on suurempi kuin 1, mutta kun $\hat{R} \approx 1$, ei ketjujen yhteisistä tekijöistä tiedetä juurikaan.

Moins ym. [14] huomauttavat, että vaikka useammalle parametrille on olemassa monimuuttujainen laajennus \hat{R} -diagnostiikasta, ei se välttämättä huomioi parametrien välistä riippuvuutta. He toteavat, että Lambertin ja Vehtarin [13] R^* on toimivampi useammalle muuttujalle, mutta sitäkin vaivaa muun muassa sama aiemmin

mainittu tulkittavuuden puute. Lisäksi R^* -diagnostiikassa voi olla tarpeen kiinnittää huomiota yksittäisen diagnostisen lukuarvon ohella myös sen epävarmuudesta kertovaan jakaumaan.

Ratkaisuksi mainitsemiinsa \hat{R} -diagnostiikan ongelmiin Moins ym. [14] esittelevät uuden paikallisen version $\hat{R}(x)$, jolla voidaan arvioida konvergenssia halutulla kvanttiililla, sekä $R(x)$:n, jonka voidaan ajatella olevan etäisyys ketjujen jakaumien välillä. Uudeksi diagnostiikaksi he ehdottavat näihin pohjautuvaa \hat{R}_∞ -menetelmää, joka on $\hat{R}(x)$:n pienin yläraja ($\hat{R}(x)$ käy läpi $m(n+1)$ arvoa, jossa m on ketjujen määrä ja n ketjun pituus). \hat{R}_∞ on helpommin tulkittava, ja sillä voidaan selvittää paremmin monia konvergenssiongelmia. Monimuutujaisten tapausten diagnosoimiseen he ehdottavat kaksiaskeleista algoritmia, jossa hyödynnetään artikkelissa esiteltyä \hat{R}_∞ -diagnostiikkaa.

Moinsin ym. [14] kehittämän $\hat{R}(x)$:n raja liittyy tyypin I virheeseen eli hylkäysvirheeseen. Heidän laskelmiensa perusteella lukuarvo 1.01 toimii varsin hyvin neljän ketjun tapauksessa. Jos ketjuja on yli 100, lukuarvo 1.1 voi olla heidän mukaansa toimivampi.

Margossian ym. [15] nostavat artikkelissaan *Nested \hat{R} : Assessing the convergence of Markov chain Monte Carlo when running many short chains* esiin \hat{R} -menetelmän ketjujen pituuden ja esittelevät uuden diagnostiikan nested \hat{R} eli \hat{R}_ν . Siinä tarkastellaan yksittäisten ketjujen sijaan ketjuklustereita eli superketjuja. Useamman ketjun käyttämisen etuna on, että sopivan lämmittelyjakson jälkeen tarvitaan lyhyempi ketju.

3.2 Muut diagnostiikat

Alaluvussa 3.1 esittelin artikkeleita, joissa lähdettiin liikkeelle \hat{R} -diagnostiikan rajoitteista ja kehiteltiin menetelmää eteenpäin. Tässä alaluvussa käsittelem artikkelia, joissa pohditaan diagnostiikkaa rajatumpiin tilanteisiin ja käyttötarkoituksiin,

kehitellään lisädiagnostiikkoja vakiintuneiden diagnostiikkojen täydennykseksi tai esitellään toimia, joiden ansiosta vakiintuneet diagnostiikat ovat käyttökelpoisia. \hat{R} -menetelmä saattaa olla myös mukana osassa näistä artikkeleista, vaikkakin enemmän ehkä sivuroolissa.

Lysy ym. [16] tarkastelevat artikkelissaan *A convergence diagnostic for Bayesian clustering* diagnostiikkaa bayesiläisessä klusteroinnissa ja esittelevät yhden ketjun diagnostiikan diskreetissä avaruudessa. Tehokkaasti tila-avaruutta tutkivien siirtymäytimien suunnittelu on haasteellista, ja esimerkiksi monet siirtymäytimet jaottelevat uudelleen vain yhden havainnon kerrallaan, jolloin konvergoiminen on hidasta. Konvergoitumisen tarkkailu ja toimiva diagnostiikka ovat siis hyvin oleellisia. Lysin ym. kehittänyt diagnostiikka perustuu siihen, että MCMC-menetelmällä muodostunutta jakaumaa ja kohdejakaumaa (stationaarista jakaumaa, engl. stationarity distribution), joka tässä tapauksessa tunnetaan normalisointivakiota vaille, verrataan keskenään. Tässä käytetään p-arvoja, eli tutkitaan, kuinka todennäköistä on, että kohdejakaumasta voidaan saada niin paljon poikkeavia arvoja kuin MCMC-simuloidut arvot, jos MCMC-menetelmällä muodostettu jakauma on konvergoitunut. P-arvot lasketaan menetelmässä käyttäen Hotelling-tyyppistä testiä.

Säilynoja ym. [17] esittelevät artikkelissaan *Graphical test for discrete uniformity and its applications in goodness-of-fit evaluation and multiple sample comparison* graafisen testin, jota voidaan hyödyntää esimerkiksi lisädiagnostiikkana diagnostiikoille kuten \hat{R} tai R^* . Menetelmä pohjautuu kertymäfunktioimuunnokseen (engl. probability integral transformation, PIT) ja empiiriseen kertymäfunktioon (engl. empirical cumulative distribution function, ECDF). Säilynoja ym. ovat kehittäneet kaksi eri menetelmää: He esittelevät menetelmän, jossa simulaatioiden avulla lasketaan empiiriselle kertymäfunktiolle (samanaikaiset) luottamusvyöhykkeet, ja nopeamman optimointimenetelmän. Monet diagnostiikat saattavat antaa MCMC-ketjulle yksittäisen lukuarvon, joka kertoo konvergoitumisesta, mutta eivät anna välttämättä li-

sätietoa konvergenssiongelman luonteesta. Artikkelissa esitelty graafinen esitys taas tarjoaa käytännöllistä tietoa poikkeavuuksien laadusta.

Sommer ym. [18] käsittelevät artikkelissaan *Connecting the Dots: Is Mode-Connectedness the Key to Feasible Sample-Based Inference in Bayesian Neural Networks?* näytteisiin pohjautuvaa päättelyä (engl. sample-based inference, SBI) ja konvergenssidiagnostiikkaa bayesiläisissä hermoverkoissa (BNN). He toteavat, etteivät perinteiset diagnostiikat toimi bayesiläisissä hermoverkoissa johtuen symmetrioista ja ketjujensisäisistä suurista vaihteluista eri kerroksissa. Symmetriat tarkoittavat samanlaista ulostuloa, vaikka mallin rakennetta vaihdettaisiin. Niiden tuloksena on monihuippuinen posteriorijakauma. Symmetria voi johtua esimerkiksi yliparametrisoinnista (engl. overparameterization) eli kun neuroverkossa on enemmän parametreja kuin tarvittaisiin.

Sommer ym. [18] ehdottavat, että bayesiläisissä hermoverkoissa parametriavaruuden konvergenssia mitattaisiin sekä ketjuittain että kerroksittain. He ovat kehittäneet diagnostiikat $\hat{R}^{(\kappa)}$ ja ketjuttaiseen tarkasteluun $\widehat{cR}^{(\kappa)}$, joka on siis $\hat{R}^{(\kappa)}$ yhdelle ketjulle. Menetelmissä ketjut jaetaan κ samankokoiseen aliketjuun \hat{R} :n laskemista varten. Sommerin ym. mukaan, kun $\kappa = 4$, niin diagnostiikan lukuarvo ≤ 1.1 viittaa konvergenssiin. Parametriavaruuden konvergenssin lisäksi Sommer ym. käsittelevät myös funktioavaruuden konvergenssia, jonka sivuutan tässä kirjallisuuskatsauksesani. Kiinnostavaa kuitenkin on, että myös funktioavaruuden konvergenssin tarkastelussa on saatettu käyttää diagnostiikkana \hat{R} -menetelmää. Sommer ym. ehdottavat artikkelissaan sen rajoitteista ja ongelmista johtuen tilalle toisenlaista kehittämäänsä menetelmää.

Sommer ym. [18] käsittelevät lyhyesti myös tehollista otoskokoa. He arvelevat, että se ei sovellu hyvin diagnostiikaksi tilanteisiin, joissa malli on erittäin yliparametrisoitu. Tällöin on odotettavissa, että näytteet ovat autokorreloituneita.

Cooper ym. [19] käsittelevät artikkelissaan *Bayesian cross-validation by paral-*

lel *Markov chain Monte Carlo* bayesiläisten mallien ristiinvalidointia, jossa aineisto jaetaan osioiksi, joista kukin vuorollaan toimii testiaineistona. He ehdottavat käytettäväksi rinnakkain ajettavia MCMC-simulaatiota ja esittelevät samalla myös tällaiseen tapaukseen sopivaa MCMC-menetelmien lämmittelyjaksoa ja diagnostiikkaa. Jos lämmittelyjakso olisi erillinen kaikille osiolle, olisi se laskennallisesti kallista. Cooper ym. suosittelevatkin samaa koko mallista lasketun lämmittelyjakson käyttöä kullekin osiolle. Diagnostiikassa Cooper ym. käyttävät tehollista otoskokoa (ESS) ja \hat{R} -menetelmää, joihin on tehty rinnakkaisajoon sopivia muutoksia ja yksinkertaistuksia. Esimerkiksi tehollisessa otoskoossa huomioidaan erien keskiarvo, kun taas \hat{R} -diagnostiikassa hyödynnetään logaritmia ja yksinkertaistamiseksi jätetään käyttämättä ketjunpuolitukset ja järjestyslukunormalisoinnit (kuten esim. [12]). Ajatuksena on, ettei yksittäisiä parametreja tai kutakin ristiinvalidoinnin posterioria tarkastella erikseen vaan kokonaisuutena eli keskitytään posterioriprediktiiviseen estimaattiin.

Konvergenssin diagnosoimista voidaan kehittää myös muuttamatta itse diagnostiikkaa esimerkiksi laajentamalla yleisesti käytettyjen standardidiagnostiikkojen käyttömahdollisuuksia muulla tavoin. Duttweiler ym. [20] esittelevät artikkelissaan *The Traceplot Thickens: MCMC Diagnostics for Non-Euclidean Spaces* yleistyksen tavanomaisille diagnostiikoille, jolloin konvergenssia voidaan tutkia myös epäeuklidisten siirtymien ja diskreettien otosavaruuksien tilanteissa. Tämä tehdään käyttämällä etäisyysfunktioita (engl. distance function) ja läheisyyskarttafunktioita (engl. proximity-map function), jolloin monimutkaisemmat rakenteet ja moniulotteiset etäisyydet saadaan kuvattua reaalityyppisiksi. Näille reaalityyppisille ketjuille voidaan käyttää tavanomaisia konvergenssidiagnostiikkoja, kuten jälkikuviota, tehollista otoskokoa ja \hat{R} -menetelmää. Artikkelissa ei siis varsinaisesti käsitellä diagnostiikkojen kehittämistä, mutta yleistetään diagnostiikat toimimaan tilanteissa, joissa se ei aiemmin ole ollut mahdollista tai on ollut hankalaa. Eritelty menetelmä on myös hel-

posti saatavissa käyttöön R-paketissa genMCMCDiag. Se ei välttämättä suoriudu paremmin kuin juuri tutkimusmenetelmää varten luodut diagnostiikat, mutta se on varsin helppo ottaa käyttöön ja antaa hyvän käsityksen konvergenssiongelmissa viemättä kallista tutkimusaikaa. Hyviä jatkotutkimuksen aiheita olisivat Duttweilerin ym. mukaan etäisyysfunktion ja läheisyyskarttafunktion kehittäminen, jolloin saataisiin entistä tehokkaampaa diagnostiikkaa ja pienennettyä laskenta-aikaa.

3.3 Artikkelien vertailu

Edellä olen jakanut tutkimusaineistoni kahteen eri alalukuun. Alaluvussa 3.1 käsitelin neljää artikkelia, joissa lähtökohtana ovat \hat{R} -menetelmän rajoitteet tai ratkaisuna luoda uusi \hat{R} -versio. Alaluvussa 3.2 taas käsitelyvuoron saivat artikkelit, joissa ongelmina ovat rajatun tilanteet, joihin tarvitaan toimivampia diagnostiikkoja, ja ratkaisuna ovat monesti keinot, joiden ansiosta vakiintuneempia diagnostiikkoja voidaan käyttää tehokkaammin.

Uusina \hat{R} -versioina artikkeleissa ovat järjestyslukunormalisoitu split- \hat{R} ja folded-split- \hat{R} [12], \hat{R}_∞ [14] ja \hat{R}_ν (eli nested \hat{R}) [15]. R^* -diagnostiikka ei tarkasti ottaen ole \hat{R} -menetelmä, vaikka sillä on tiettyjä yhteneväisyyksiä \hat{R} -diagnostiikkaan. Lambert ja Vehtari [13] mainitsevatkin sen \hat{R} :n serkkuna. Lisäksi alaluvussa 3.2 mainitut $\hat{R}^{(\kappa)}$ ja $\widehat{cR}^{(\kappa)}$ [18] ovat \hat{R} -menetelmiä, mutta ne on suunniteltu käytettäväksi bayesiläisissä hermoverkoissa.

Alaluvun 3.2 viidestä artikkelista kolmessa käsitellään konvergenssin diagnoosimista erityistilanteissa. Näissä artikkeleissa esitellään diagnostiikkoja bayesiläiseen klusterointiin [16], bayesiläisiin hermoverkkoihin [17] ja MCMC:n rinnakkaisajoon [19]. Kahdessa muussa artikkelissa taas esitellään käsittelyjä ja täydennyksiä diagnosoimisen parantamiseksi. Duttweiler ym. [20] tarkastelevat tilannetta, jossa poiminta-avaruus on diskretisoitu tai poimijan siirtymät ovat epäeuklidisia, ja ratkaisuna on näytteiden esikäsittely, jonka ansiosta on mahdollista käyttää vakiintu-

neita diagnostiikkoja. Diagnosoimista voidaan myös parantaa kehittämällä lisätietoa antavia diagnostiikkoja, joita voi käyttää vakiintuneempien diagnostiikkojen ohella. Säilynoja ym. [17] esittelevät artikkelissaan tällaisen graafisen täydennyksen, joka voi auttaa hahmottamaan paremmin mahdollisia ongelmia konvergenssissa.

Tarkastelemistani artikkeleista kaksi käsittelee MCMC:n rinnakkaisajoa ja sen erityisvaatimukseen vastaamista. Niiden ratkaisut asiaan eroavat toisistaan, kuten Cooper ym. [19] artikkelissaankin toteavatkin. Margossian ym. [15] ratkaisevat asian käyttämällä lyhyiden ketjujen muodostamia ketjuklustereita. Cooper ym. taas käyttävät pidempiä ketjuja kuhunkin osioon, uudelleenkäyttävät yhteisen lämmittelyjakson ja tekevät yksinkertaistuksia diagnostiikkaan. Jaoin nämä eri alalukuihin, sillä Margossianin ym. vastauksena on uusi \hat{R} -versio kun taas Cooperin ym. ratkaisussa tehdään ennemminkin valintoja, jotka tehostavat diagnostiikkaa.

Tutkielmani aineiston perusteella näyttäisi siltä, että myös graafisia menetelmiä ja diagnostiikkojen visualisoimista kehitellään jonkin verran. Vehtari ym. [12] esittelevät artikkelissaan kolme uutta visuaalista työkalua: rank plotin, kvantiili- ja pienten intervallien kuvion ja tehokkuus per iteraatio -kuvion. Säilynoja ym. [17] ovat kehitelleet graafisen lisädiagnostiikan, jota voidaan hyödyntää esimerkiksi \hat{R} -menetelmän kanssa. Duttweilerin ym. [20] kehittelemät menetelmät taas mahdollistavat jälkikuvion käyttämisen tilanteissa, joissa sen käyttäminen ei ole toiminut hyvin.

Olen edellä esitellyt konvergenssidiagnostiikan kehitystä, joka vaikuttaa olevan varsin aktiivista. On kuitenkin hyvä huomata, että lisäksi kehitteillä on myös muita tapoja löytää ja selvittää konvergenssiongelmiä. Esimerkiksi Huang ym. [21] esittelevät artikkelissaan *Debugging convergence problems in probabilistic programs via program representation learning with SixthSense* erilaisen lähestymistavan. Pro-

babilistinen¹ ohjelmointi mahdollistaa bayesiläisen päättelyn laajemmalle joukolle tutkijoita, mutta esimerkiksi juuri konvergenssiongelmiin tunnistamisessa tarvittaisiin silti tilastotieteellistä osaamista. Ratkaisuksi Huang ym. tarjoavat SixthSense-järjestelmän, jolla voidaan ennustaa probabilistisen ohjelman konvergenssia ja korjata konvergenssivirheitä. Huangin ym. tutkimuksissa SixthSense saavutti yli 78 prosentin tarkkuuden konvergenssin ennustamisessa. Lisäksi heidän mukaansa se kertoo laajalti käytössä olevan Stan-ohjelman sisäänrakennettuja varoituksia paremmin konvergenssiongelmiin syitä.

¹Käytän *probabilistic* sanasta suomennosta *probabilistinen* sen käytännöllisyyden vuoksi. Esimerkiksi Joni Kämäräinen [22, s. 55] on päätenyt käyttämään nimitystä *todennäköisyyksiin perustuva*, ja mainitsee muista vaihtoehdoista *probabilistisen* ja *stokastisen*.

4 Yhteenveto

Tässä luvussa vastaan luvussa 1 esittämiini tutkimuskysymyksiin luvussa 3 käsittelemieni yhdeksän artikkelin pohjalta. Lisäksi pohdin saavuttiko tutkielmani sille asettamani tavoitteet. Lopuksi luon lyhyesti katseen konvergenssidiagnostiikan tulevaisuuteen.

TK1. Onko konvergenssidiagnostiikka ratkaistu ongelma?

Käsittelemieni artikkelien perusteella näyttäisi siltä, että konvergenssidiagnostiikka ei ole selvitetty ongelma. Sitä tutkitaan edelleen, ja lähiaikoina onkin kehitelty lisäyksiä ja muutoksia ennestään olemassa olleisiin menetelmiin sekä uusia menetelmiä.

TK2. Mitä konvergenssidiagnostiikan kehittämisessä on ajankohtaista 2020-luvulla?

Näyttäisi siltä, että \hat{R} -diagnostiikalla on varsin keskeinen rooli konvergenssin tutkimisessa. Monet uudet ideat käsittelevät juuri se parantelemista ja uudistamista (esim. [14] ja [15]), hyödyntämistä muutoksin [19] tai kokonaan uusia menetelmiä, jotka ainakin jossain määrin pohjautuvat siihen [13]. Suurimmassa osassa käsittelemiäni artikkeleita \hat{R} -menetelmää hyödynnetään tai se huomioidaan jollain tavoin. Myöskään graafisten menetelmien kehittäminen ei ole täysin loppunut [12, 17].

Diagnostiikkoja kehitellään myös erityistilanteisiin kuten esimerkiksi bayesiläiseen klusterointiin [16]. Rinnakkain ajettavien MCMC-simulaatioiden diagnostiikkaa käsittelee kaksi artikkelia [15, 19]. Itse diagnostiikan kehittelyn sijaan saatetaan

konvergenssin tarkastelemista parantaa esimerkiksi näyttöiden esikäsittelyllä [20] tai täydentämällä vakiintuneita diagnostiikkoja lisädiagnostiikalla [17].

Tutkielmani on pääpiirteinen ja rajallinen katsaus konvergenssidiagnostiikan kehityksen suuntaviivoihin. Jo kandidaatintutkielman pituus asettaa omat rajoitteensa sille, kuinka laajasti tai syvällisesti aihettaan on mahdollista tutkia. En ole tutkielmassani kuvannut kovin perinpohjaisesti esimerkiksi artikkeleissa esiteltyjä uusia diagnostiikkoja tai muutoksia vanhoihin diagnostiikkoihin, sillä olen pyrkinyt pitämään sen yleisluontoisena kirjallisuuskatsauksena viimeaikojen tutkimukseen. Kehitellyt menetelmät ja muokkaukset aiempiin menetelmiin ovat lisäksi varsin uusia, ja vasta aika näyttää niiden todellisen suosion ja kuinka kauan ne ovat ajankohtaisia. Tästä huolimatta koen työn saavuttaneen sille asettamani tavoitteet ja löytäneeni tutkimuskysymyksiin tyydyttävät vastaukset.

On hyvin kiinnostavaa nähdä, miten diagnostiikkoja tullaan kehittämään tulevaisuudessa. Voisi olla mahdollista, että koneoppiminen valtaa alaa. Tästä hyvänä esimerkkinä on jo R^* -diagnostiikka, joka hyödyntää koneoppimista [13]. Toisaalta tekoälyä voidaan hyödyntää myös suoraan ohjelmistoissa, mistä esimerkkinä on SixthSense [21]. Ehkäpä konvergenssin diagnosoimista automatisoidaan ja piiloteetaan suoraan ohjelmien sisään enenevässä määrin. Tämä voisi herättää huolen siitä, onko konvergenssin ja diagnostiikan ymmärtäminen tulevaisuudessa vaarassa. Asiaa voisi katsoa toiseltakin kantilta ja miettiä, olisiko kaikki tehtävä itse alusta alkaen. Teknologian kehityksen ja uusien innovaatioiden tarkoituksena on kuitenkin helpottaa arkea ja tehdä työstä sujuvampaa. Se, että diagnostiikkaa ei tällä hetkellä ole jätetty täysin tekoälyn suoritettavaksi, ei kerro siitä, etteikö joissain tapauksissa diagnostiikkaa suoritettaisi varsin rutiininomaisesti miettimättä sitä tarkemmin tai varsinaisesti ymmärtämättä sitä syvällisemmin. Esimerkiksi Geyer [1] on kritisoinut joitain MCMC-menetelmien käytännöksi muodostuneita tapoja. Ehkäpä vastaavia löytyy myös diagnostiikan puolelta. Diagnostiikan tutkimuksen ja diagnostiikan arki-

päiväisen käytön väliset mahdolliset erot olisivat kiinnostava tutkimusaihe. Ne voisivat nostaa esiin enemmän diagnostiikan ongelmia. Samalla ne tarjoaisivat kiinnostavan kurkistusikkunan aikaan ennen kuin koneoppimista hyödyntävät automatisoidut diagnostiset ratkaisut valtaavat alaa (jos valtaavat).

Lähdeluettelo

- [1] C. J. Geyer, "Introduction to Markov Chain Monte Carlo", teoksessa *Handbook of Markov Chain Monte Carlo*, 1. painos, New York: Chapman ja Hall/CRC, toukokuu 2011, s. 3–48, ISBN: 978-0-429-13850-8. DOI: 10.1201/b10905-2.
- [2] C. Robert ja G. Casella, "A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data", *Statistical Science*, vol. 26, nro 1, helmikuu 2011, ISSN: 0883-4237. DOI: 10.1214/10-STS351.
- [3] J. Alho, E. Arjas, E. Läärä ja P. Pere, *Tilastotieteen sanasto* (Suomen Tilastoseuran julkaisuja 8), 2. laitos. Helsinki: Suomen Tilastoseura ry, 2023, ISBN: 978-952-9795-14-7.
- [4] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari ja D. B. Rubin, *Bayesian Data Analysis*. lokakuu 2024. url: <https://users.aalto.fi/~ave/BDA3.pdf>.
- [5] C. M. Bishop, *Pattern recognition and machine learning* (Information science and statistics). New York: Springer, 2006, ISBN: 978-0-387-31073-2.
- [6] V. Roy, "Convergence Diagnostics for Markov Chain Monte Carlo", *Annual Review of Statistics and Its Application*, vol. 7, nro 1, s. 387–412, maaliskuu 2020, ISSN: 2326-8298, 2326-831X. DOI: 10.1146/annurev-statistics-031219-041300.

-
- [7] C. J. Geyer, "Practical Markov Chain Monte Carlo", *Statistical Science*, vol. 7, nro 4, s. 473–483, 1992, ISSN: 08834237.
- [8] A. Gelman ja D. B. Rubin, "Inference from Iterative Simulation Using Multiple Sequences", *Statistical Science*, vol. 7, nro 4, marraskuu 1992, ISSN: 0883-4237. DOI: 10.1214/ss/1177011136.
- [9] S. P. Brooks ja A. Gelman, "General Methods for Monitoring Convergence of Iterative Simulations", *Journal of Computational and Graphical Statistics*, vol. 7, nro 4, s. 434–455, joulukuu 1998, ISSN: 1061-8600, 1537-2715. DOI: 10.1080/10618600.1998.10474787.
- [10] L. Gong ja J. M. Flegal, "A Practical Sequential Stopping Rule for High-Dimensional Markov Chain Monte Carlo", en, *Journal of Computational and Graphical Statistics*, vol. 25, nro 3, s. 684–700, heinäkuu 2016, ISSN: 1061-8600, 1537-2715. DOI: 10.1080/10618600.2015.1044092.
- [11] C. P. Robert ja G. Casella, *Monte Carlo Statistical Methods* (Springer Texts in Statistics). New York, NY: Springer New York, 2004, ISBN: 978-1-4419-1939-7 978-1-4757-4145-2. DOI: 10.1007/978-1-4757-4145-2.
- [12] A. Vehtari, A. Gelman, D. Simpson, B. Carpenter ja P.-C. Bürkner, "Rank-Normalization, Folding, and Localization: An Improved \hat{R} for Assessing Convergence of MCMC (with Discussion)", *Bayesian Analysis*, vol. 16, nro 2, kesäkuu 2021, ISSN: 1936-0975. DOI: 10.1214/20-BA1221.
- [13] B. Lambert ja A. Vehtari, "R*: A Robust MCMC Convergence Diagnostic with Uncertainty Using Decision Tree Classifiers", *Bayesian Analysis*, vol. 17, nro 2, kesäkuu 2022, ISSN: 1936-0975. DOI: 10.1214/20-BA1252.
- [14] T. Moins, J. Arbel, A. Dutfoy ja S. Girard, "On the Use of a Local \hat{R} to Improve MCMC Convergence Diagnostic", *Bayesian Analysis*, vol. 20, nro 1, maaliskuu 2025, ISSN: 1936-0975. DOI: 10.1214/23-BA1399.

-
- [15] C. C. Margossian, M. D. Hoffman, P. Sountsov, L. Riou-Durand, A. Vehtari ja A. Gelman, *Nested \hat{R} : Assessing the convergence of Markov chain Monte Carlo when running many short chains*, toukokuu 2024. DOI: 10.48550/arXiv.2110.13017.
- [16] M. Lysy, M. Asgharian ja V. Partovi Nia, ”A convergence diagnostic for Bayesian clustering”, *WIREs Computational Statistics*, vol. 13, nro 4, e1536, heinäkuu 2021, ISSN: 1939-5108, 1939-0068. DOI: 10.1002/wics.1536.
- [17] T. Säilynoja, P.-C. Bürkner ja A. Vehtari, ”Graphical test for discrete uniformity and its applications in goodness-of-fit evaluation and multiple sample comparison”, *Statistics and Computing*, vol. 32, nro 2, s. 32, huhtikuu 2022, ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-022-10090-6.
- [18] E. Sommer, L. Wimmer, T. Papamarkou, L. Bothmann, B. Bischl ja D. Rügamer, *Connecting the Dots: Is Mode-Connectedness the Key to Feasible Sample-Based Inference in Bayesian Neural Networks?*, toukokuu 2024. DOI: 10.48550/arXiv.2402.01484.
- [19] A. Cooper, A. Vehtari, C. Forbes, D. Simpson ja L. Kennedy, ”Bayesian cross-validation by parallel Markov chain Monte Carlo”, *Statistics and Computing*, vol. 34, nro 4, s. 119, elokuu 2024, ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-024-10404-w.
- [20] L. Duttweiler, J. Klus, B. Coull ja S. W. Thurston, *The Traceplot Thickens: MCMC Diagnostics for Non-Euclidean Spaces*, elokuu 2024. DOI: 10.48550/arXiv.2408.15392.
- [21] Z. Huang, S. Dutta ja S. Misailovic, ”Debugging convergence problems in probabilistic programs via program representation learning with SixthSense”, *International Journal on Software Tools for Technology Transfer*, vol. 26, nro 3,

s. 249–268, kesäkuu 2024, ISSN: 1433-2779, 1433-2787. DOI: 10.1007/s10009-024-00737-2.

- [22] J. Kämäräinen, *Koneoppimisen perusteet*. Helsinki: Otatieto, 2023, ISBN: 978-951-672-379-5.