Tampereen yliopisto

Aino Tarvainen

RAUTATEIDEN KULUMISEN MALLINTAMINEN MIXTURE-MALLINNUKSELLA

Informaatioteknologian ja viestinnän tiedekunta
Tilastollisen data-analyysin kandidaattitutkielma
Huhtikuu 2021

Tiivistelmä

Aino Tarvainen: Rautateiden kulumisen mallintaminen
mixture-mallinnuksella
Tilastollisen data-analyysin kandidaattitutkielma
Tampereen yliopisto
Matematiikan ja tilastollisen data-analyysin tutkinto-ohjelma
Huhtikuu 2021

Mixture-malli on tilastotieteen malli, jolla voidaan mallintaa ja tunnistaa aineiston sisäisiä osapopulaatioita. Tutkielman aiheena on mixture-mallinnuksen soveltaminen rautateistä vuosina 2008–2018 kerättyyn pitkittäisaineistoon. Tutkimuskysymyksenä on, voiko mixture-malleilla tunnistaa ne aikavälit, jolloin rataosuutta on korjattu eli tuettu. Kysymykseen vastataan sovittamalla kolmelle esimerkkirataosuudelle mixture-mallit ja sitten visuaalisesti analysoimalla saatuja sovitteita.

Tutkielmassa ensin esitellään Väylävirastolta saatu rautateiden mittausaineisto, jota tutkielmassa käytetään. Tämän jälkeen määritellään mixture-mallin yleinen esitysmuoto ja joitakin sen ominaisuuksia. Samassa osassa määritellään myös suurimman uskottavuuden menetelmä ja EM-algoritmi, joita käytetään mixture-mallin parametrien estimoinnissa. Mixture-mallien määrittelyyn on käytetty apuna Geoffrey McLachlanin ja David Peelin teosta *Finite Mixture Models*. Mixture-mallien teoriaosuuden jälkeen esitellään lyhyesti tutkielman mallien sovitukseen ja laskemiseen käytetyt R-ohjelman funktiot. Tutkielma päättyy tulosten esittelyyn ja analysointiin, ja lopuksi tehdään yhteenveto tutkimuksen tulosten merkityksestä.

Tutkielman tärkein tulos on se, että mixture-mallinnus sopi rautatieaineistoon odotettua huonommin. Mallinnus onnistui hyvin vain niissä tilanteissa, joissa aineiston arvot olivat huomattavan erisuuria. Muissa tapauksissa mallinnus ei tuottanut tutkimuskysymyksen kannalta hyödyllisiä ratkaisuja. Johtopäätös on se, että tutkielmassa esitelty menetelmä ei ole tarpeeksi luotettava keino tuentakertojen löytämiseen rautateiden mittausaineistosta.

Avainsanat: mixture model, klusterointi, EM-algoritmi, suurimman uskottavuuden menetelmä

Sisällys

1 Johdanto

Rautatiet kuluvat käytön ja luonnonvoimien vaikutuksesta, mikä johtaa epätasaiseen rataan. Siksi on tärkeää, että raiteiden tasaisuutta mitataan säännöllisesti, jotta rataa voidaan huoltaa ennen kuin se on liian vaarallinen junille kulkea. Nämä tuentakerrat voidaan havaita mittausten aikasarja-aineistossa rataosuuden epätasaisuuden arvon jyrkkänä laskuna, ja tämä onnistuu yleensä hyvin aineistosta piirrettyjen kuvaajien avulla silmämittaisesti. Tämän kandidaattitutkielman tavoite on tutkia, voidaanko tuentakerrat tunnistaa sen sijaan yhtä hyvin tilastollisin menetelmin käyttäen mixture-mallinnusta ja klusterointia. Tuentakertojen havaitseminen aineistosta on pragmaattinen tutkimusaihe, sillä tuentakertojen ajankohdasta ja paikasta ei ole saatavilla arkistoituja tietoja.

Rautatieaineiston mallinnuksen työkaluksi valittiin mixture-mallinnus, sillä sen avulla havainnot pystytään klusteroimaan helposti. Klusterit ovat tässä tapauksessa niiden havaintojen joukkoja, jotka koostuvat ajallisesti peräkkäisistä havainnoista ja joiden välillä rataa ei ole tuettu. Näin tuentakertojen ajankohta voidaan määritellä klusterien rajojen mukaan. Hypoteesina on se, että mixture-algoritmi osaa tunnistaa epätasaisuuden arvojen pudotuksen klusterin määrittäväksi ominaisuudeksi. Tämän kandidaattitutkielman keskeisin tutkimusaihe on tämän hypoteesin testaus.

Tutkielman rakenne etenee seuraavasti: Ensiksi luvussa 2 esitellään käytetyn aineiston alkuperä ja ominaisuudet ja sen jälkeen osassa 3 määritellään mixture-mallinnuksen lyhyt teoria sekä sen mallien estimoinnissa käytettävät suurimman uskottavuuden menetelmä ja EM-algoritmi. Viimeksi mainitussa osassa käytetyissä kaavoissa ja määritelmissä mukaillaan McLachlanin ja Peelin (2000) teoksen esitys- ja merkintätapaa. Luvussa 4 lyhyesti kerrotaan niistä ohjelmointimenetelmistä, joilla tulokset laskettiin tätä työtä varten. Luvussa 5 selvitetään tutkielman tuloksia ja lopuksi luvun 6 yhteenvedossa arvioidaan näiden tulosten merkitystä. Liitteestä 1 löytyy R-koodi, jota on käytetty aineiston muokkaamiseen ja liitteessä 2 on kirjoitettuna R-koodi, jolla työn esimerkkien mixture-mallit on laskettu, sekä näiden mallien informaatiokriteeritulosteet.

2 Aineisto

Suomessa rautateiden kuntoa seuraa Väylävirasto, joka suorittaa ratakunnon mittauksia noin 2-6 kertaa vuodessa. Mittauksia tehdään radantarkastuvaunulla, joka mittaa kiskojen geometrista kuntoa, eli kuinka tasainen rautatie on. Vaunun tuottamaan numeeriseen dataan pääsee käsiksi vain tietyt rajatut tahot, ja tässä työssä käytettävä aineisto on saatu Tampereen yliopiston tutkimuskeskus Terran välityksellä.

Aineisto on kerätty aikavälillä 20.10.2008 - 12.8.2018, jonka aikana mittauksia on tehty 22 kertaa, noin kaksi kertaa vuodessa. Mittaukset on otettu Luumäen ja Ilomantsin väliseltä radalta. Alkuperäisessä datassa mittaus on tehty joka 25 senttimetrin välein 65.922 ratakilometrin pituiselta matkalta. Geometria mitataan millimetreissä, ja se saa arvoja nollan kummaltakin puolelta.

Jotta voitaisiin tarkastella radanosien epätasaisuuden muutosta mittauskertojen välillä, tämän kandidaattitutkielman kirjoittaja on käsitellyt aineistoa seuraavasti: Aineisto on ensin jaettu 200 metrin peräkkäisiin otoksiin, mikä merkitsee 800 peräkkäistä havaintoa per otos, ja näistä otoksista on otettu niiden keskihajonta, yksi jokaiselle 22 mittauskerralle. Tuloksena on aineisto, joka koostuu 330 radanpätkästä, joilla kullakin on 22 keskihajonta-arvoa.

Vertailemalla radanosan peräkkäisiä keskihajontoja toisiinsa voidaan arvioida radan epätasaisuutta tietyllä ajanhetkellä: mitä suurempi keskihajonta, sitä epätasaisemmaksi rautatie on päässyt. Tavanomaisesti keskihajonnat nousevat suunnilleen lineaarisesti ajan suhteen.

Tämän tutkimuksen tarkoituksena on erottaa tästä keskihajontojen datasta ne välit, jolloin keskihajonta on pudonnut huomattavasti. Tämä pudotus on indikaattori sille, että kyseinen kohta rataa on tasoittunut huomattavasti mittauskertojen välillä. Tämänlainen suuri muutos epätasaisuudessa selittyy ainoastaan sillä, että rataa on tuettu.

Osassa 5 esiteltävissä tuloksissa mixture-mallinnusta sovelletaan aina yhden radanpätkän keskihajontoihin. Aineistosta nostetaan tarkasteluun kolme esimerkkiä, jotka ovat otettu aineiston riveiltä 59, 289 ja 307. Tämä merkitsee sitä, että esimerkki 59 sisältää järjestyksessä 59:nnen 200 metrin rataosuuden havainnot, eli sen sijainti radalla on noin 11.8 kilometrin päässä mittausten aloituspisteestä.

3 Mixturemallinnuksen teoria

3.1 Mixture-mallit

Olkoon $\mathbf{Y}_1, \ldots, \mathbf{Y}_n$ satunnaisotos, jonka koko on n, ja $\mathbf{Y}_j$ p:n havainnon pituinen satunnaisvektori, jonka tiheysfuntkio on muotoa $f(\mathbf{y}_j) \subset \mathbb{R}^p$. Olkoon $\mathbf{Y} = (\mathbf{Y}_1^T, \ldots, \mathbf{Y}_n^T)^T$, missä $\mathbf{Y}$ edustaa koko otosta.

Mixture-mallinnuksessa satunnaisvektorin Y_j havaintojen katsotaan noudattavan kahta tai useampaa eri jakaumaa - usein voidaan olettaa, että havainnot edustavat itse asiassa eri alipopulaatioiden yksilöitä tai ilmiöitä. Tällöin havaintojen tiheysfunktio täytyy esittää monen tiheysfunktion painotettuna summana, eli se voidaan ilmaista muodossa

$$f(\mathbf{y}_j) = \sum_{i=1}^{g} \pi_i f_i(\mathbf{y}_j), \tag{1}$$

missä $f_i(\mathbf{y}_j)$ ovat mixturen komponenttitiheysfunktioita ja π_i komponenttipainoja, eli komponentin j prioritodennäköisyyksiä. Painot ovat epänegatiiviisa ja summautuvat yhdeksi, eli

$$0 \leq \pi_i \leq 1 \tag{2}$$

ja

$$\sum_{i=1}^{g} \pi_i = 1. \tag{3}$$

Koska mixture-mallien komponenttijakaumat $f_i(\mathbf{y}_j)$ ovat käytännössä yleisesti oletettu kuuluvan parametrisiin perheisiin, ne voidaan kirjoittaa myös muodossa $f_i(\mathbf{y}_j; \boldsymbol{\theta}_i)$, missä $\boldsymbol{\theta}_i$ on i:nnen oletetun komponenttijakauman tuntemattomien parametrien vektori. Koko mixture-malli voidaan kirjoittaa nyt parametrisesti muodossa

$$f(\mathbf{y}_j; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i f_i(\mathbf{y}_j; \boldsymbol{\theta}_i), \tag{4}$$

missä vektori $\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_{g-1}, \boldsymbol{\xi}^T)^T$ sisältää mixture-mallin kaikki tuntemattomat parametrit. Vektoriin $\boldsymbol{\xi}$ täten sisällytetään vektoreiden $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_g$ parametrit, jotka määrittävät komponenttien jakaumat. Koska mixture-painot π_i summautuvat yhdeksi, viimeistä painoa π_g ei ole tarvetta ottaa mukaan vektoriin triviaalisuuden takia.

Koska mixture-mallin perusoletus on se, että aineiston havainnot on kerätty eri jakaumia noudattavista alipopulaatioista, on se luonteva työkalu havaintojen klusteroinnille. Sovitetun mallin komponenttien g posterioritodennäköisyydet saadaan kaavalla

$$p_{ij} = \frac{\pi_i f_i(\mathbf{y}_j)}{f(\mathbf{y}_j)},\ i = 1, \ldots, g. \tag{5}$$

Havainnon $\mathbf{y}_j$ katsotaan nousseen siitä komponenttijakaumasta i, joka saa suurimman posterioritodennäköisyysarvon tämän kyseisen havaintovektorin arvoilla. Posterioritodennäköisyyksiä vertailemalla havainnot voidaan täten suoraviivaisesti klusteroida sovitetun mixture-mallin mukaisesti.

3.2 Parametrivektorin estimointi suurimman uskottavuuden menetelmällä

Mixture-mallin parametrinen sovittaminen aineistoon edellyttää vektorin $\mathbf{\Psi}$ estimointia. Tässä työssä mallien estimointiin käytetään EM-algoritmiä (Dempster, Laird & Rubin, 1977), joka lienee tällä hetkellä suosituin käytetty mixture-estimoinnin työkalu. EM-algoritmilla parametrit estimoidaan iteratiivisesti käyttäen suurimman uskottavuuden menetelmää.

Estimaatti $\hat{\mathbf{\Psi}}$ saadaan suurimman uskottavuuden menetelmällä kehittämällä sopiva ratkaisu uskottavuusyhtälöön,

$$\frac{\delta \, log \, L(\Psi)}{\delta \, \mathbf{\Psi}} = \mathbf{0}, \tag{6}$$

missä

$$L(\mathbf{\Psi}) = \prod_{j=1}^{n} f(\mathbf{y}_j; \mathbf{\Psi}) \tag{7}$$

on vektorin $\mathbf{\Psi}$ uskottavuusfunktio ja satunnaismuuttujat $\mathbf{y}_1, \ldots, \mathbf{y}_n$ oletetaan riippumattomiksi toisistaan. Estimoinnissa käytetään uskottavuusfunktion luonnollista logaritmia laskemisen helpottamiseksi: logaritmifunktion monotonisuuden vuoksi yhtälö tuottaa samat ratkaisut estimaateille kuin lineaarinen yhtälö.

Parametrivektorin suurimman uskottavuuden estimaattori (SUE) on se juuri $\hat{\mathbf{\Psi}}$, joka maksimoi uskottavuusyhtälön, eli

$$\hat{\mathbf{\Psi}} = arg \, max \, L(\mathbf{\Psi}). \tag{8}$$

Teoriassa SU-estimaattori on parametriavaruuden juuri, joka on uskottavuusyhtälön globaali maksimi. Käytännössä globaalin maksimin löytäminen ja todistaminen, että se todella on globaali maksimi, voi olla hankalaa tai jopa mahdotonta: kaikkien mahdollisten lokaalien maksimien laskeminen voi olla liian työlästä, jolloin ei ole keinoa osoittaa, että globaali maksimi on todella löydetty. Joissain tapauksissa uskottavuusyhtälö voi olla myös rajaton, eli sille ei yksinkertaisesti ole olemassa globaalia maksimia.

Vaikka globaalin maksimin löytäminen on SU-menetelmän optimistinen tavoite, käytännössä mallinnuksessa voidaan käyttää myös lokaalin maksimin määrittelemää estimaattivektoria. On osoitettu (McLachlan & Peel, 2000, 41), että myös lokaalien maksimien estimaattorit voivat silti noudattaa niitä ominaisuuksia, nimellisesti tehokkuuden, tarkentuvuuden ja asymptoottisen normaalisuuden ominaisuuksia, jotka tekevät suurimman uskottavuuden menetelmästä hyvän estimointimenetelmän. Tiedetään myös, että aloitusarvot voivat vaikuttaa ratkaisevasti esimaattiarvoihin (mm. Seidel, Mosler & Manfred, 2000). Tämän ja edellä mainitun syyn vuoksi eri juurien määrittämien estimointiratkaisuiden tarkastelu mixture-mallien sovittamisessa voi olla hyödyllistä, jos globaali maksimi ei ole selviö.

3.3 EM-algoritmi

Mixture-mallien tapauksessa uskottavuusfunktioon (7) sijoitetaan yhtälössä (1) kuvattu mixture-mallin määritelmä. Näin mixture-mallien log-uskottavuus on

$$log\, L(\mathbf{\Psi}) = \sum_{j=1}^{n} log f(\mathbf{y}_j; \mathbf{\Psi}) = \sum_{j=1}^{n} log[\sum_{i=1}^{g} \pi_i f_i(\mathbf{y}_j; \boldsymbol{\theta}_i)]. \tag{9}$$

SU-estimointia ei pystytä tekemään suoraan mixture-mallinnuksessa kompleksisuuden takia, joten estimointi suoritetaan EM-algoritmillä (*engl.* expectation-maximization algorithm). Algoritmilla on kaksi askelta, E- ja M-askel, jotka yksinkertaistettuna koostuvat ensin estimaattien odotusarvon (*expectation*) laskemisesta ja sitten uskottavuusyhtälön maksimoimisesta (*maximization*).

EM-algoritmin soveltaminen mixture-mallinnukseen voidaan nähdä puuttuvan datan ongelmana. Havaintovektoria $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_n^T)^T$ käsitellään datana, josta määritellään puuttuvan komponenttiluokkien jäsenyysindikaattorivektorit $\mathbf{z}_1, \ldots, \mathbf{z}_n$. Puuttuvan datan kehyksessä $\mathbf{z}_j$ ovat g-ulotteisia vektoreita, joille pätee

$$z_{ij} = (\mathbf{z}_j)_i = \begin{cases} 1, & \text{kun } \mathbf{y}_j \text{ kuuluu mixturen } i\text{:nteen komponenttiin.} \\ 0, & \text{kun } \mathbf{y}_j \text{ ei kuulu mixturen } i\text{:nteen komponenttiin.} \end{cases} \tag{10}$$
$$(i = 1, \ldots, g; j = 1, \ldots, n)$$

Nyt havaintovektorin täydennetty muoto on

$$\mathbf{y}_C = (\mathbf{y}^T, \mathbf{z}^T)^T, \tag{11}$$

missä $\mathbf{z} = (\mathbf{z}_1^T, \ldots, \mathbf{z}_n^T)$ ovat realisoituneita arvoja riippumattomille satunnaismuuttujille $\mathbf{Z}_j$, joille pätee

$$\mathbf{Z}_1, \ldots, \mathbf{Z}_n \overset{i.i.d.}{\sim} Mult_g(1, \boldsymbol{\pi}), \tag{12}$$

missä π on mixture-painojen vektori.

Näin täydennetyn datan log-uskottavuusfunktio määritellään muodossa

$$log\, L_C(\mathbf{\Psi}) = \sum_{i=1}^{g} \sum_{j=1}^{n} z_{ij}[log\, \boldsymbol{\pi}_i + log\, f_i(\mathbf{y}_j; \boldsymbol{\theta}_i)]. \tag{13}$$

EM-algoritmissä edellisen iteraation $\mathbf{\Psi}$:n ja $\mathbf{z}$:n estimaattiarvot ovat pohja seuraavan iteraation E- ja M-askelille, eli algoritmi iteratiivisesti päivittää estimaatteja konvergoituen näin lopulta johonkin parametriavaruuden lokaaliin maksimiin. Askelia toistetaan niin kauan, kunnes estimaatit saavuttavat etukäteen määritetyn suppenemistason, eli kun erotus $L(\mathbf{\Psi}^{(k+1)}) - L(\mathbf{\Psi}^{(k)})$ ei enää parane merkityksellisesti iteraatioiden välillä, tai kunnes iteraatioiden maksimimäärä saavutetaan tai jokin muu ehto täyttyy.

3.3.1 E-askel

E-askeleessa lasketaan täydennetyn datan log-uskottavuuden $log\, L_c(\mathbf{\Psi})$ ehdollinen odotusarvo ehdolla $\mathbf{y}$ käyttämällä parametrien $\mathbf{\Psi}$ senhetkistä sovitetta. Ensimmäiseen iteraatioon sovelletaan $\mathbf{\Psi}$:n erikseen määriteltyjä aloitusarvoja $\mathbf{\Psi}^{(0)}$, ja sitä seuraavissa iteraatioissa käytetään hyväksi edellisen iteraation tuottamia estimaatteja $\mathbf{\Psi}^{(k)}$. E-askeleessa estimoitava odotusarvo iteraatiolla $k+1$ olkoon täten

$$Q(\mathbf{\Psi}; \mathbf{\Psi}^{(k)}) = E[log\, L_C(\mathbf{\Psi}^{(k)}) \mid \mathbf{y}]. \tag{14}$$

Koska mixture-mallin täydennetyn datan määritelmään (11) on otettu mukaan puuttuva data $\mathbf{z}$, sen log-uskottavuusfunktio $log\, L_C(\mathbf{\Psi})$ on lineaarinen z_{ij}:ssä. Tämän ansiosta E-askeleessa tarvitaan laskea ehdollinen odotusarvo ainoastaan satunnaismuuttujalle Z_{ij} ehdolla $\mathbf{y}$. Koska määritelmän (12) mukaan komponenttijäsenyysarvot z_{ij} noudattavat multinomijakaumia, joille $n = 1$, niiden odotusarvo on $E(Z_{ij}) = P(Z_{ij} = 1)$. Täten log-uskottavuuden ehdollinen odotusarvo määritetään

$$\begin{aligned} E_{\mathbf{\Psi}^k}(Z_{ij} \mid \mathbf{y}) &= P_{\mathbf{\Psi}^{(k)}}(Z_{ij} = 1 \mid \mathbf{y}) \\ &= \tau_i(\mathbf{y}_j; \mathbf{\Psi}^{(k)}), \end{aligned} \tag{15}$$

missä

$$\begin{aligned} \tau_i(\mathbf{y}_j; \mathbf{\Psi}^{(k)}) &= \pi_i^{(k)} f_i(\mathbf{y}_j; \boldsymbol{\theta}_i^{(k)}) / f(\mathbf{y}_j; \mathbf{\Psi}^{(k)}) \\ &= \pi_i^{(k)} f_i(\mathbf{y}_j; \boldsymbol{\theta}_i^{(k)}) / \sum_{h=1}^{g} \pi_h^{(k)} f_h(\mathbf{y}_j; \boldsymbol{\theta}_h^{(k)}) \end{aligned} \tag{16}$$

on mixture-mallin posterioritodennäköisyys sille, että otoksen j:nnes jäsen kuuluu mallin i:nteen komponenttiin sen havaintojen $\mathbf{y}_j$ arvojen perusteella. Nyt täydennetyn datan log-uskottavuuden k:nnen iteraation ehdollinen odotusarvo ehdolla $\mathbf{y}$ määritellään

$$Q(\mathbf{\Psi}; \mathbf{\Psi}^{(k)}) = \sum_{i=1}^{g} \sum_{j=1}^{n} \tau_i(\mathbf{y}_j; \mathbf{\Psi}^{(k)})[log\, \pi_i + log\, f_i(\mathbf{y}_j; \boldsymbol{\theta}_i)]. \tag{17}$$

Kiteytettynä E-askeleessa siis korvataan täyden datan log-uskottavuusfunktion komponenttijäsenyysarvot z_{ij} niiden posterioritodennäköisyyksillä $\tau_i(\mathbf{y}_j)$, jotka saadaan laskettua sijoittamalla estimaatit $\mathbf{\Psi}^{(k)}$ kaavaan (16).

3.3.2 M-askel

M-askeleessa otetaan E-askeleessa laskettu päivitettyjen posterioritodennäköisyyksien log-uskottavuusyhtälö $Q(\mathbf{\Psi}; \mathbf{\Psi}^{(k)})$ ja maksimoidaan se $\mathbf{\Psi}$:n suhteen. Saatu SU-estimaatti on $k+1$:nnen EM-iteraation mixture-mallien tuntemattomien parametrien estimaatti $\mathbf{\Psi}^{(k+1)}$, toisin sanoen

$$\mathbf{\Psi}^{(k+1)} = arg\, max\, Q(\mathbf{\Psi}; \mathbf{\Psi}^{(k)}). \tag{18}$$

Mixture-mallin log-uskottavuusfunktion määritelmässä (13) nähdään, että komponenttipainoille π_i ja parametrivektoreille $\boldsymbol{\theta}_i$ ei ole kaavassa yhteisiä tekijöitä. Tämän vuoksi parametrien SU-estimointi tehdään erikseen.

Komponenttipainot kuvaavat mixture-mallin kunkin komponenttiluokan teoreettista osuutta otoksessa. Koska z_{ij} saa arvon 1, jos havainto kuuluu komponenttiin i, ja muutoin 0, komponenttipainovektorien π_i SU-estimaattori on komponenttijäsenyysarvojen otoskeskiarvo, eli

$$\hat{\pi}_i = \frac{1}{n} \sum_{j=1}^{n} z_{ij}. \tag{19}$$

Huomaa, että komponenttijäsenyysarvot z_{ij} ovat puuttuvaa dataa, joten ne täytyy korvata niiden posterioritodennäköisyyksillä $\tau(.)$, jotka estimoitiin E-askeleessa (ks. kaava (16)). Näin komponenttipainoestimaatit $\pi^{(k+1)}$ saadaan kaavasta

$$\pi_i^{(k+1)} = \frac{1}{n} \sum_{j=1}^{n} \tau_i(\mathbf{y}_j; \boldsymbol{\Psi}^{(k)}). \tag{20}$$

Komponenttijakaumien parametrivektorin $\boldsymbol{\xi}$ maksimoiminen tapahtuu ratkaisemalla log-uskottavuusyhtälölle $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$ sopiva juuri $\boldsymbol{\xi}$:n suhteen, eli

$$\boldsymbol{\xi}^{(k+1)} = arg\, max \sum_{i=1}^{g} \sum_{j=1}^{n} \tau_i(\mathbf{y}_j; \boldsymbol{\Psi}^{(k)})\, log\, f_i(\mathbf{y}_j; \boldsymbol{\theta}_i). \tag{21}$$

3.4 Mixture-mallin komponenttien määrän valitseminen

Tähän asti esitellyt kaavat ja määritelmät käyttävät komponenttiluokkien määrää g ennalta määriteltynä vakiona. Klusterointiongelmissa tilanne on useasti se, että luokkien todellista määrää ei ole ennalta saatavissa, joten aineistoon sovitettava mixture-komponenttien lukumäärä g_0 täytyy arvioida erikseen.

Tässä työssä käytetään komponenttilukumäärän valitsemiseen mallin informaatiokriteerejä. Analyysissa vertaillaan erityisesti Akaiken (1973, 1974) informaatiokriteeriä (AIC), Schwarzin (1978) Bayesiläistä informaatiokriteeriä (BIC) ja Biernackin ym. (2000) ICL-kriteeriä, joidenka arvot R-ohjelma laskee automaattisesti, kun mixture-malleja sovitetaan flexmix-komennolla.

4 R ja FlexMix-pakkaus

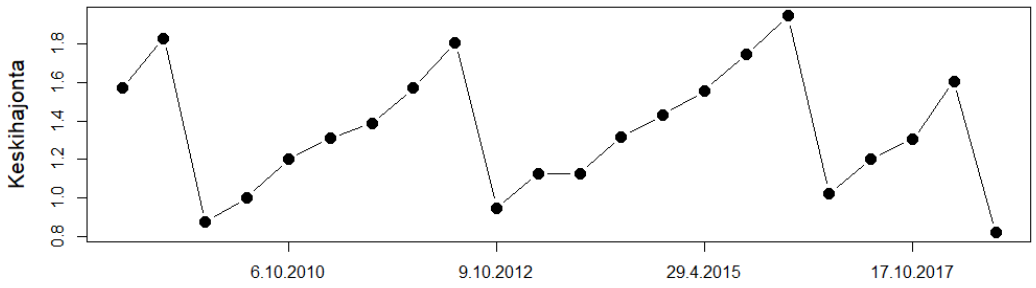
Tässä työssä mixture-mallien sovitukseen käytetään R-kirjaston flexmix-lisäosan versiota 2.3-15 (ks. Leisch 2004, Gruen & Leisch 2007, 2008). Pakkauksen funktio `flexmix` palauttaa käyttäjän määrittämällä mallilla dataan sovitetun flexmix-olion, joka on käytännössä k-komponenttinen mixture-malli. Sijoittamalla aineisto funktioon `stepFlexmix` saadaan palautusarvona stepFlexmix-olio, joka sisältää monta mixture-mallia, joiden sovitukseen on käytetty eri komponenttiluokkamääriä g_0. Funktion parametri `k` on määritelty tässä funktiossa kokonaisarvovektoriksi, joka sisältää nämä komponenttiluokkamäärät. stepFlexmix kutsuu flexmix-funktiota `nrep` kertaa jokaiselle parametrin `k` määrittämälle komponenttiluokkien lukumäärälle ja valitsee funktion palauttamista flexmix-olioista jokaiselle arvolle `k` sen mallin, joka tuottaa pienimmän loguskottavuuden.

stepFlexmix sopii hyvin mallien vertailuun, sillä se palauttaa eri komponenttimääräisten mallien lisäksi niiden vastaavat informaatiokriteeriarvot. Analysoimalla näitä arvoja voidaan valita se komponenttimäärä, jolla saadaan verrattain parhain mixture-malli. Malli saadaan kutsutuksi stepFlexmix-oliosta sijoittamalla se funktioon `getModel(object, which = )`, jossa parametrin which arvoksi laitetaan sen informaatiokriteerin nimi, jonka perusteella malli halutaan valita (esim. which = "BIC").

5 Tulokset

5.1 Vakioselittäjän malli

Intuition mukaan rautatien epätasaisuus kasvaa lineaarisesti ajan suhteen. Oletus pitänee hyvin paikkansa, kun katsotaan kuvaajaa 1, jossa on piirrettynä radan kohdan 59 keskihajonnat aikajärjestyksessä. Kuvaajasta löytyy ainakin neljä lineaarisesti kasvavaa suoraa sekä viimeisin havainto, jonka jälkeisestä kehityksestä ei ole dataa, mutta jonka voisi kuvitella olevan seuraavan lineaarisen nousukehityksen alku. Tämän vuoksi havaintoihin on aiheellista sovittaa ensimmäisen asteen lineaarinen malli.



Kuvaaja 1: Rataosuuden 59 mittausten keskihajonnan muutos ajallisesti

Koska käytetyssä aineistossa jokaiselle ratakohdalle on saatavilla vain yhdenlaisia havaintoja (sen mittausarvojen keskihajonta tiettynä mittauspäivänä), sovitettavassa mixture-mallissa on selittävänä tekijänä ainoastaan vakio. Yksityiskohtaisemmin esitettynä mittauskerran t keskihajonnan suuruuden mallinnusformula on muotoa

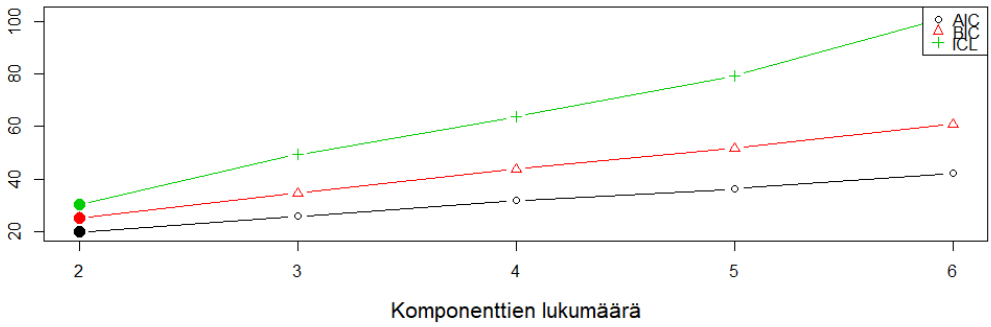
$$keskihajonta \; = \; \beta_0 + \epsilon_t,$$

missä residuaalit ϵ_t on oletettu riippumattomiksi.

Kun Kuvaajan 1 havainnot annetaan stepFlexMix-funktiolle (tarkemmat funktioparametrit ja tulosteet malleille löytyvät liitteestä 2), saadaan tuloksena viisi mixture-mallisovitetta, joiden informaatiokriteerit ovat listattuna taulukkoon 1 ja piirrettynä kuvaajaan 2.

Taulukko 1: Rataosuuden 59 mixture-mallien informaatiokriteeriarvot

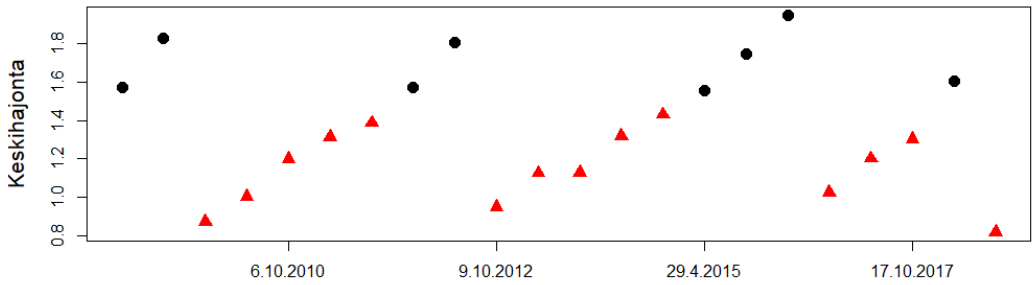
g	Konvergoituminen	AIC	BIC	ICL
2	EI	19.876	25.331	30.163
3	KYLLÄ	25.811	34.539	49.272
4	EI	31.772	43.773	63.577
5	EI	36.280	51.555	79.223
6	EI	42.280	60.827	102.005



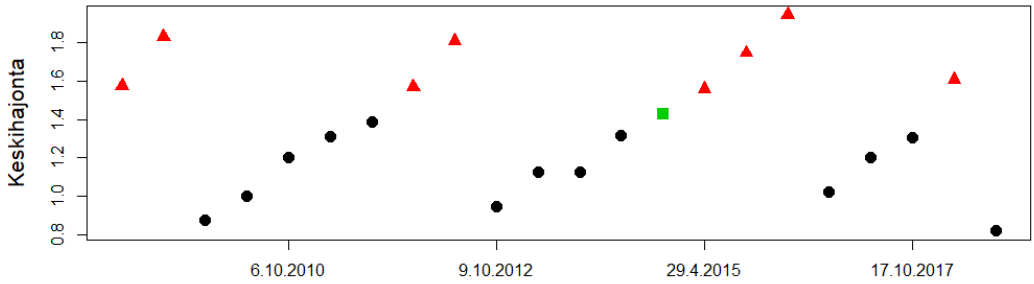
Kuvaaja 2: Rataosuuden 59 mixture-mallien informaatiokriteerit

Taulukon 1 ja Kuvaajan 2 mukaan kahden komponentin malli saa pienimmät informaatiokriteeriarvot jokaisella menetelmällä. Toisaalta Taulukossa 1 on huomattavaa, että EM-algoritmi konvergoitui ainoastaan sovitettaessa kolmen komponentin mallia. Tämän vuoksi tarkastellaan kahden ja kolmen komponentin mallia.

Kuvaajassa 3 on esitettynä kahden komponentin mallilla klusteroidut havainnot ja kuvaajassa 4 vastaava kolmen mallin tuottama erottelu. Kahden komponentin mallissa klusterikoot ovat 8 ja 14, ja kolmen komponentin mallissa suuremmasta klusterista on erotettu yksi havainto omaksi klusterikseen kahden komponentin mallin 14 havainnon klusterista. Komponenttien määrän nostaminen ei vaikuta tässä tilanteessa klusteroinnin laatuun ratkaisevasti.

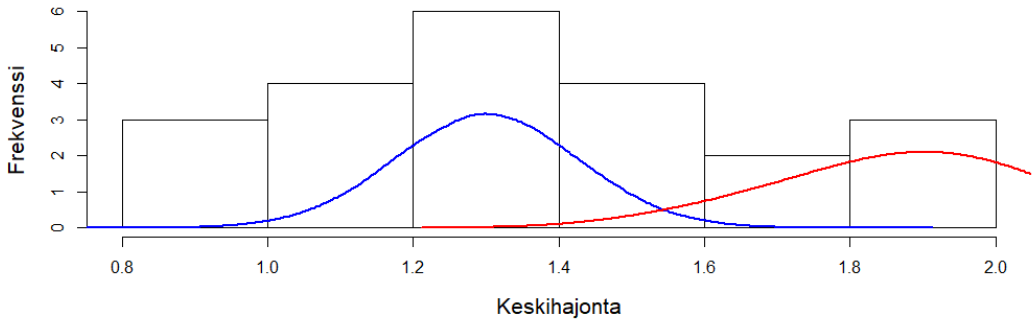


Kuvaaja 3: Rataosuuden 59 havaintojen klusterointi kahden komponentin mixture-mallilla



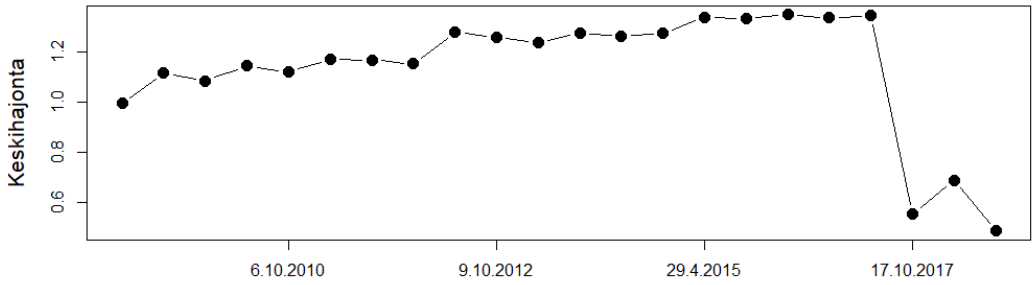
Kuvaaja 4: Rataosuuden 59 havaintojen klusterointi kolmen komponentin mixture-mallilla

Vaikka ihmissilmä näkee havainnoissa selkeän trendin, algoritmi ei ole saanut aikaan samansuuntaista ryhmittelyä. On ilmeistä, että keskiarvohavainnot on ryhmitelty niiden suuruden mukaan, eli kuvaajissa niiden y-akselin arvojen mukaan. Itse asiassa, kun katsotaan kuvaajaa 5, joka on kohdan 59 havaintojen histogrammi, voisi havaintojen kuvitella noudattavan kahden eri normaalijakauman mixture-jakaumaa, joiden odotusarvot voisivat olla noin 1.3 ja 1.9.



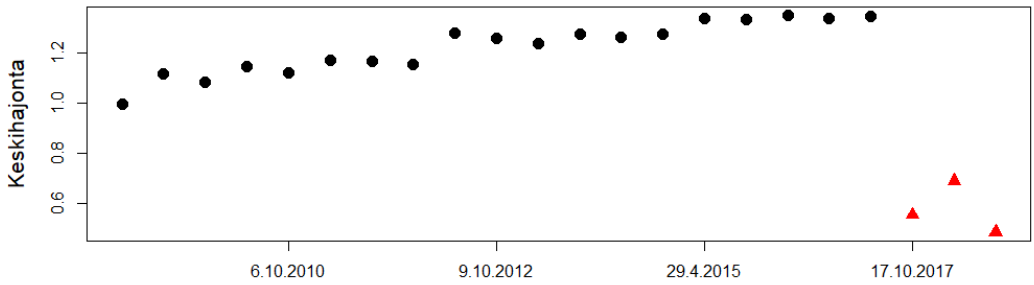
Kuvaaja 5: Rataosuuden 59 keskihajontojen frekvenssijakauma

Kuvaajassa 6 on aineiston kohdan 307 keskihajontojen aikasarjakuvaaja. On selkeää, että tällä alueella on mitä todennäköisimmin tehty korjaustoimenpide ennen lokakuuta 2017, minkä lisäksi tuentakerran jälkeen mitatut arvot ovat täysin eri tasoisia kuin aikaisempi trendi.



Kuvaaja 6: Rataosuuden 307 mittausten keskihajonnan muutos ajallisesti

Kun kohtaan 307 sovitetaan kahden komponentin mixture-sovitus saadaan kuvaajan 7 mukaiset klusterit. Koska tämän kohdan eri ajankohtien havainnot ovat selkeästi eri suuruusluokkaa, löytää algoritmi järkevät klusterit helposti.



Kuvaaja 7: Rataosuuden 307 havaintojen klusterointi kahden komponentin mixture-mallilla

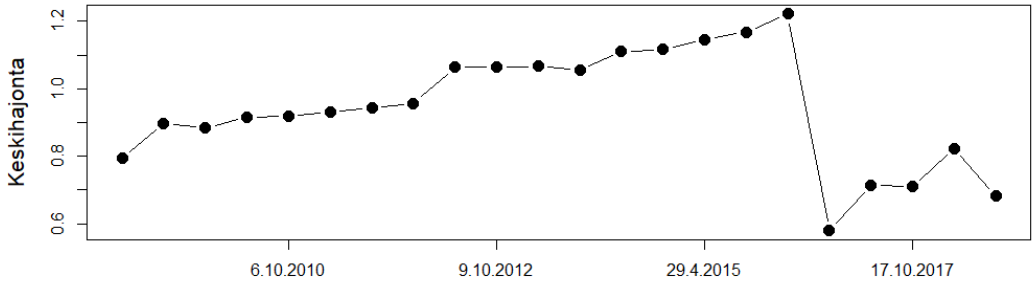
5.2 Aikaselittäjän malli

Vakioselittäjän mallissa ongelma on siinä, että algoritmilla ei ole käytössään informaatiota siitä, että havainnot ovat aikasarjasta. Eräs tapa yrittää ratkaista tämä ongelma on liittää jokaiselle keskiarvohavainnolle aikaindikaattoriarvo: koska havaintojen mittausajankohdat ovat noin tasaisin väliajoin, indikaattoriksi asetetaan arvot 1:stä 22:een. Tällöin mallin yhtälöksi määritetään

$$keskihajonta \ = \ \beta_0 + \beta_1 t + \epsilon_i,$$

missä t on keskihajonnan mittauskerran indikaattori ja $t = 1, \ldots, 22$.

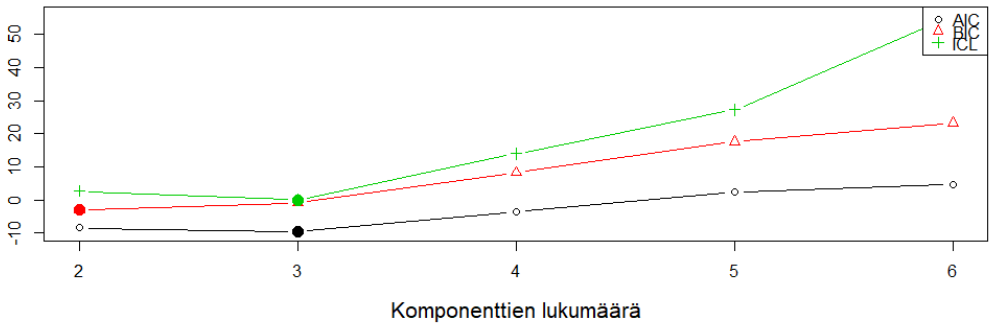
Tutkitaan tämän mallin sopivuutta aineistoon ottamalla tarkasteluun kohta 289 (kuvaaja 8). Kuvaajasta voisi päätellä, että tätä radankohtaa on tuettu ainakin kerran mittauskertojen aikana noin vuonna 2016, jolloin sen keskihajonta puolittui 1.2:sta 0.6:een mittauskertojen välillä. Sovitetaan havainnoille ensin vakiomuuttujan tuottama mixture-malli, ja verrataan sen määrittämiä klustereita aikamuuttujan sisältämän mixture-sovituksen tuloksiin.



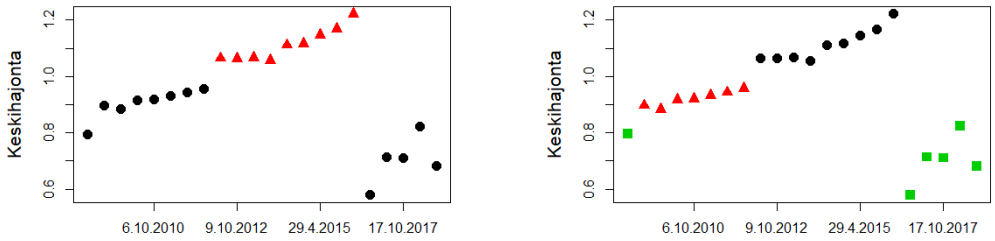
Kuvaaja 8: Rataosuuden 289 mittausten keskihajonnan muutos ajallisesti

Kuvaajassa 9 on yksinkertaisen mallin eri komponenttilukumäärillä sovitettujen mallien informaatiokriteerit. Kahden ja kolmen komponentin mallit näyt-

tävät pärjäävän yhtä hyvin, joten klusteroidaan havainnot kahden ja kolmen komponentin malleilla. Kuviossa 10 nähdään, että kumpikin malli luokittelee havainnot jälleen niiden suuruuden mukaan, vaikka suuri osa havainnoista klusterien sisällä ovat peräkkäisiä.



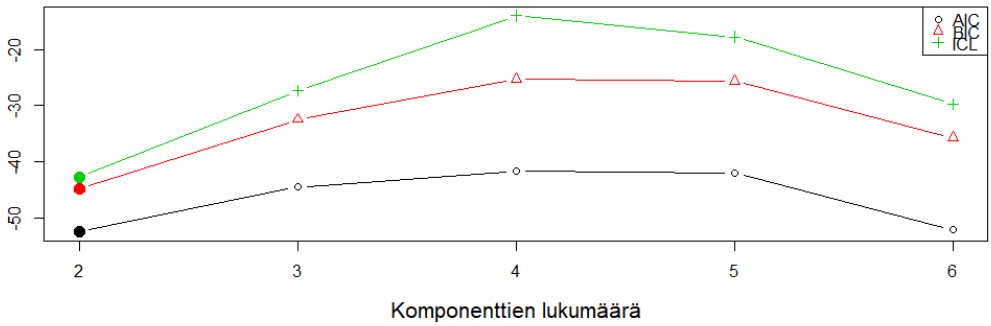
Kuvaaja 9: Rataosuuden 289 mixture-mallien informaatiokriteerit



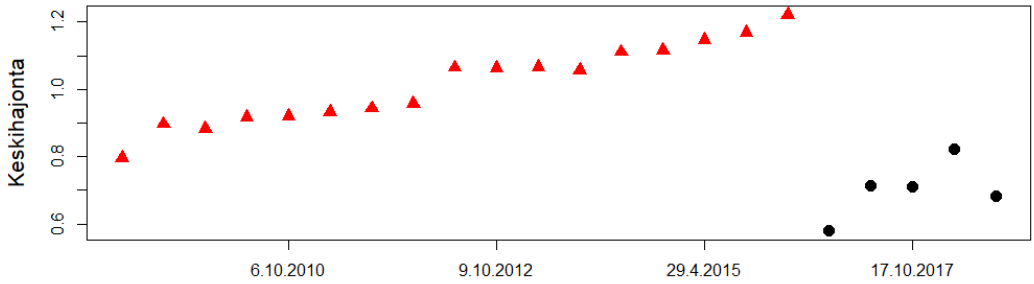
(a) Kahden komponentin mixture-malli (b) Kolmen komponentin mixture-malli

Kuvaaja 10: Rataosuuden 289 havaintojen klusterointi vakioselittäjän mixture-mallilla

Kuvaajassa 11 on informaatikriteerit malleille, joihin oli lisätty aikaindikaattori selittäväksi muuttujaksi. Arvojen mukaan kahden komponentin malli on parhain, joten klusteroidaan havainnot sen mukaan. Kuvaajasta 12 näkee, että klusterointi on onnistunut nyt verrattain paremmin kuin yksinkertaisessa mallissa: klusterien havainnot ovat aikajärjestyksessä perättäisiä.



Kuvaaja 11: Rataosuuden 289 informaatiokriteerit mixture-malleille, joissa aika t selittäjänä

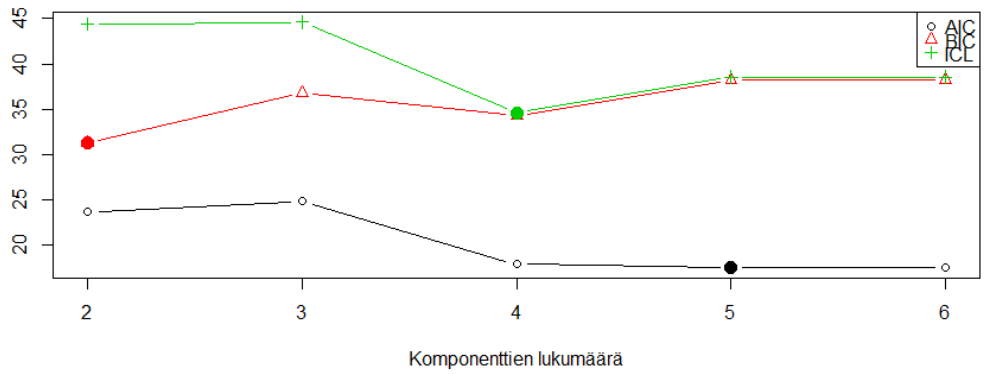


Kuvaaja 12: Rataosuuden 289 havaintojen klusterointi kahden komponentin aikaselittäjän mixture-mallilla

Toisaalta on aiheellista kyseenalaistaa aikamuuttujaa käyttävän mallin hyödyllisyys yleisesti, kun sitä käytetään eri radankohtien havaintoihin. Aikamuuttuja voi auttaa joidenkin havaintojen klusteroinnissa, kuten edellä osoitettiin kohdan 289 esimerkissä, mutta tämä ei tarkoita, että aikamuuttujamalli suoriutuu huomattavasti paremmin kaikenlaisissa kohdissa.

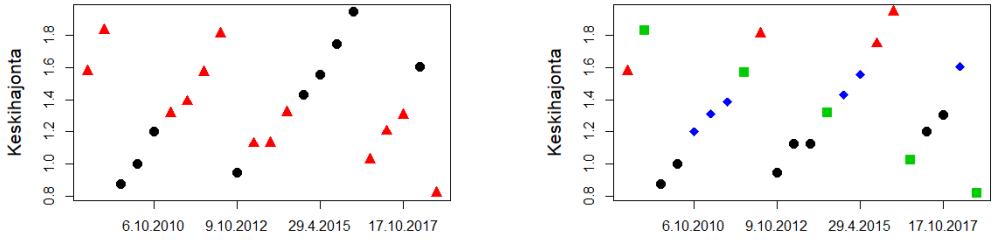
Kun aikamuuttujamalli sovitetaan esimerkiksi osan 5.1 kohtien 59 ja 307 havaintoihin (kuvaajat 1 ja 6), saadaan vähintään yhtä hedelmättömiä klusterointituloksia kuin vakioselittäjän mallilla.

Estimointi aikaselittäjämallilla antaa kohdan 59 havainnoille kuvaajan 13 mukaiset informaatiokriteerit. Eri tavoin saadut kriteeriarvot antavat tässä tapauksessa eriäviä ehdotuksia sopivimmalle komponenttien lukumäärälle. Valitaan klusterointiin silti komponenttimäärät 2 ja 4, sillä AIC-arvot eivät eroa huomattavasti toisistaan komponenttimäärille 4 ja 5.



Kuvaaja 13: Rataosuuden 59 informaatiokriteerit mixture-malleille, joissa aika t selittäjänä

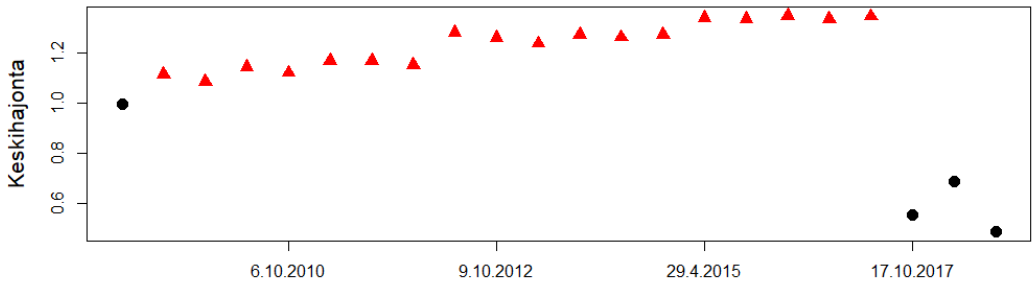
Kahden ja neljän komponenttien klusteroinnit on kuvattuna kuvaajassa 14. Nähdään, että aikamuuttuja ei ole auttanut löytämään sopivia ryhmittelyjä. Se näyttää silti noudattavan eri periaatetta kuin vakiomuuttujamalli, jonka taipumus on luokitella havainnot niiden y-akselin arvojen mukaan.



(a) Kahden komponentin mixture-malli (b) Neljän komponentin mixture-malli

Kuvaaja 14: Rataosuuden 59 havaintojen klusterointi aikaselittäjän mixture-mallilla

Kuvaajassa 15 nähdään aikaselittäjämallin kahden komponentin mixture-mallin klusterointi. Havainnot ovat jakautuneet lähes samoin kuin vakioselittäjän malli (kuvaaja 7), mutta tässä tapauksessa ensimmäinen havainto on sijoitettu "väärään"klusteriin. Aikaindikaattoriarvo ei vaikuta auttavan tarpeeksi merkittävästi, jotta algoritmi päätyisi klustereihin, joiden arvot olisivat ajallisesti peräkkäisiä ja kuvaisivat yhtä keskihajonnan nousukautta.



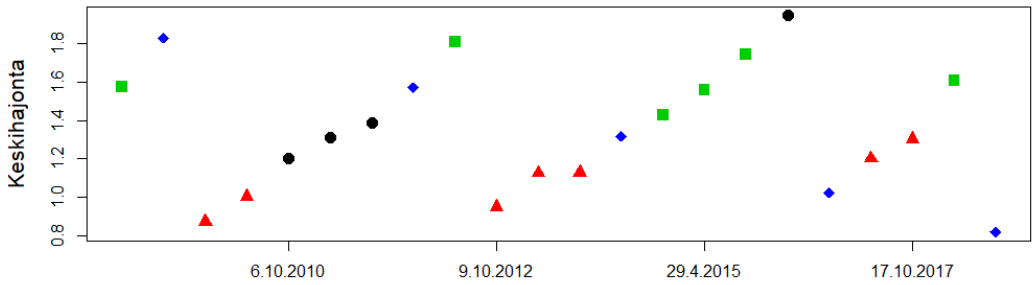
Kuvaaja 15: Rataosuuden 307 havaintojen klusterointi kahden komponentin aikaselittäjän mixture-mallilla

5.3 Huomioita aikasarja-aineiston mixture-mallinnuksesta ja algoritmin aloitusarvoista

Rautateiden kulumisen mixture-mallintamisen suurin ongelma on se, että mallit eivät huomioi havaintojen korreloituneisuutta. Pitkittäisdatassa edellinen havainto vaikuttaa yleensä seuraavan mittauksen havaintojen kehitykseen. Oletus on, että rataosion peräkkäiset keskihajontahavainnot vaikuttavat toistensa posterioritodennäköisyyteen kuulua samaan klusteriin, eli tuentakertojen väliin jäävään kulumisen aiheuttaman keskihajontojen nousun käyrään. Ajallisesti lähekkäiset kaksi havaintoa kuulunevat samaan kehityskäyrään todennäköisemmin kuin havainnot, joiden välillä on monta mittauskertaa.

Tämä oletus rikkoo tosin mixture-mallinnuksen oletuksia. Määritelmän 12 mukaan mixture-mallin parametrien estimoinnissa EM-algoritmilla havainnon komponenttijäsenyyden arvot z_{ij} noudattavat riippumattomia multinomijakaumia. Toisin sanoen havaintojen komponenttijäsenyyden posterioritodennäköisyys oletetaan itsenäiseksi muiden havaintojen komponenttijäsenyyksistä. Tätä riippumattomuuden oletusta käytetään hyväksi täten myös FlexMix-pakkauksen funktioissa. Tämä tarkoittaa sitä, että edellä esitellyt klusteroinnit on estimoitu olettamuksilla, jotka eivät päde kyseiseen käytettyyn aineistoon.

Kuten osassa 3.2 todetaan, EM-algoritmi löytää aina yhden parametriavaruuden lokaalin maksimin tietyillä aloitusarvoilla. Edellä esitetyt tulokset on saavutettu käyttämällä vain yhden siemenen tuottamia arvoja. Eri maksimien määrittämien klustereiden analysoiminen voi olla usein hyödyllistä. Toisaalta tässä tutkielmassa käytettyjen havaintojen lukumäärät ovat verrattain pienet, joten eri estimaattien tuottamat klusterit voivat erota toisistaan vain rajatun verran ja usein erot eri klusterointien välillä ovat merkityksettömiä.



Kuvaaja 16: Rataosuuden 59 havaintojen klusterointi neljän komponentin aikaselittäjän mixture-mallilla käyttäen eri siemenlukua

Jos esimerkiksi kohtaan 59 estimoidaan komponentit toisilla aloitusarvoilla käyttäen samaa osassa 5.2 esiteltyä aikaselittäjän yhtälöä, saadan kuvaajan 16 mukaiset klusterit. Havainnot ovat jakautuneet eri tavoin verrattuna kuvaajaan 14b, mutta klusterit eivät silti anna tämän työn tutkimuskysymyksen kannalta informatiivisia ryhmittelyjä.

6 Johtopäätökset

Johdannossa asetettiin tämän tutkielman tavoitteeksi selvittää, onko mixture-mallinnus sopiva työkalu tuentakertojen havaitsemiseksi rautateiden mittausaineistosta. Työhön otettiin esimerkeiksi kolmen eri rataosuuden keskihajontahavainnot, jotka oli kerätty noin kymmenen vuoden ajalta. Kun näitä havaintoja oli klusteroitu sekä vakioselittäjän että aikaselittäjän sisältävillä mixture-malleilla, havaittiin että menetelmä jakoi havainnot parhaiten niissä otoksissa, joissa tuentakerran jälkeiset keskihajonta-arvot olivat huomattavasti pienempiä aikaisempiin havaintoihin verrattuna. Tapauksissa, joissa arvioitujen tuentakertojen väliset intervallit saivat samantasoisia keskihajonta-arvoja, algoritmi ei löytänyt hyviä klusterointiratkaisuja.

Mallinnus ja klusterointi tuottivat huonompia tuloksia kuin tutkimuksen alussa oli odotettu. Tähän pystyttiin tunnistamaan ainakin kaksi syytä: Ensiksikin, käytetyt aineistot olivat pieniä. Analyysi oli rajoitettu 22 havainnon klusterointiin, sillä aineiston rautatietä oli mitattu vain noin pari kertaa vuodessa, minkä lisäksi algoritmi sai syötteekseen ainoastaan selitettävän muuttujan, eli mittauksien keskihajonnat. Aineiston informaation määrää oli pyritty lisäämään määrittämällä havainnoille ajankohtaindikaattorit, joiden avulla havainnot klusteroitiin uudestaan nk. aikamuuuttujan mallilla. Ajan muuttujan lisäämisen hyödyttävä vaikutus klusteroinnin laatuun jäi silti kyseenalaiseksi — esimerkeissä se toimi noin yhtä hyvin kuin vakiomuuttujan malli tai hieman paremmin. Toiseksi, tutkielman esimerkkimallien estimointiin käytetty R-funktio flexmix sisältää sellaisia aineiston oletuksia, jotka eivät päteneet tämän työn aineistossa. Yleiseen mixture-mallin määritelmään sisältyy, että havaintojen oletetaan olevan keskenään korreloimattomia. Rautateiden säännöllinen mittaus taas on luonnollisesti autokorreloitunutta, sillä se on aikasarja-aineisto.

Jos mixture-mallinnus onnistuu löytämään rautateiden mittausaineistosta tuentakerrat vain silloin, kun keskihajontojen erot ovat huomattavan suuret, herää kysymys, onko tämän menetelmän käytöllä käytännön hyötyä. Tässä tutkielmassa analysoitujen esimerkkien perusteella voitaisiin jopa väittää, että yksi ihminen olisi pärjännyt algoritmia paremmin aineiston klusteroinnissa, varsinkin kun havaintojen määrä on näin pieni. Mixture-mallien käyttö siis osoittautui epäluotettavaksi keinoksi havaita tuentakertoja rautateiden mittausaineistosta — ainakin sillä tavalla, miten se oli toteutettu tässä tutkielmassa. Parempia klusterointituloksia voisi saada kenties käyttämällä menetelmiä, jotka huomioivat autokorreloituneisuuden tai omaavat löyhemmät aineisto-oletukset.

Mixture-mallinnus voi silti osoittautua hyödylliseksi muilla tavoin rautateitä analysoidessa. Tässä tutkielmassa käytettyä aineistoa voitaisiin esimerkiksi mallintaa käyttäen kaikkia rataosuuksia, jolloin klusterit koostuisivat niistä radanpätkistä, joiden keskihajontojen kehityskäyrät ovat samankaltaiset. Saman klusterin jäsenille voisi täten etsiä yhteisiä vaikuttavia syitä vaikkapa sille, miksi niiden tasaisuus on samaa luokkaa tai miksi ne kuluvat samalla tahdilla.

Lähteet

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *Proceedings of the Second International Symposium on Information Theory*, 267—281.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19, 716—723.

Biernacki, C, Celeux, G. & Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(7), 719—725.

Dempster, A., Laird, N. & Rubin, D. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1—38.

Gruen, B & Leisch, F. (2007). Fitting finite mixtures of generalized linear regressions in R. *Computational Statistics & Data Analysis*, 51(11), 5247—5252.

Gruen, B. & Leisch, F. (2008). FlexMix Version 2: Finite mixtures with concomitant variables and varying and constant parameters. *Journal of Statistical Software*, 28(4), 1—35.

Leisch, F. (2004), FlexMix: A general framework for finite mixture models and latent class regression in R. *Journal of Statistical Software*, 11(8).

McLachlan, G. & Peel, D. (2000). *Finite Mixture Models*, Wiley, New York.

Schwarz, G. (1978). Estimating the dimension of a model, *Annals of Statistics*, 6(2), 461—464.

Seidel, W., Mosler, K. & Manfred, A. (2000). A Cautionary Note on Likelihood Ratio Tests in Mixture Models. *Annals of the Institute of Statistical Mathematics*, 52, 481—487.

Liitteet

Liite 1. Keskihajontojen laskeminen alkuperäisestä datasta

```
# Matriisiin rautatie.khajonnat.all tallennetaan aineiston 200 metrin keskihajonnat.
# Alkuperäinen aineisto on matriisissa rautatie.dat, jonka kolme ensimmäistä saraketta
# sisältävät havainnon sijaintiin liittyvää ei-numeerista dataa.
rautatie.khajonnat.all <- matrix(0, 330, 22)
for (i in 1:329) {
  alku <- (i-1)*800
  loppu <- alku + 800
  for (j in 4:25) {
    rautatie.khajonnat.all[i,j-3] <- sd(rautatie.dat[(alku:loppu),j])
  }
}
# Koska alkuperäisen aineiston rivien lukumäärä ei ole jaollinen 800:lla, viimeinen
# rataosuus on 122.25 metrin pituinen.
for (j in 4:25) {
  rautatie.khajonnat.all[330,j-3] <- sd(rautatie.dat[263200:263689,j])
}
```

Liite 2. Esimerkkien mixture-mallinnus ja mallien informaatiokriteerit

```
library(flexmix)
```

```
## Warning: package 'flexmix' was built under R version 3.6.2
```

```
## Loading required package: lattice
```

```
# Rataosuus 59
ro59 <- rautatie.khajonnat.all[59,]
ro59 <- data.frame(matrix(data= c(1:22, as.numeric(t(ro59))), ncol = 2))
colnames(ro59) <- c("aika", "khajonta")

# Vakioselittäjän mixture-malli (kohta 59)
set.seed(1)
vakio.flexmix <- stepFlexmix(khajonta ~ 1, k = 2:6, nrep=10, dat = ro59)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * * * * * * *
## 6 : * * * * * * * * * *
```

```
vakio.flexmix # Kohdan 59 informaatiokriteerit, vakioselittäjän malli
```

```
##
## Call:
## stepFlexmix(khajonta ~ 1, dat = ro59, k = 2:6, nrep = 10)
##
##   iter converged k k0    logLik      AIC      BIC       ICL
## 2  200     FALSE 2  2 -4.937850 19.87570 25.33091  30.16251
## 3   82      TRUE 3  3 -4.905509 25.81102 34.53936  49.27217
## 4  200     FALSE 4  4 -4.885848 31.77170 43.77316  63.57667
## 5  200     FALSE 5  5 -4.140012 36.28002 51.55462  79.22327
## 6  200     FALSE 6  6 -4.139828 42.27966 60.82738 102.00489
```

```
# Aikaselittäjän mixture-malli
set.seed(1)
aika.flexmix <- stepFlexmix(khajonta ~ aika, k = 2:6, nrep=10, dat = ro59)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * *Error in FLXfit(model = model, concomitant = concomitant, control = control,  :
##    124 Log-likelihood: NaN
##  * * * * *
## 6 : * * * * * * * * * *
```

```
aika.flexmix # Kohdan 59 informaatiokriteerit, aikaselittäjän malli
```

```
##
## Call:
```

```
## stepFlexmix(khajonta ~ aika, dat = ro59, k = 2:6, nrep = 10)
##
##   iter converged k k0    logLik      AIC      BIC      ICL
## 2   57      TRUE 2  2 -4.836357 23.67271 31.31001 44.39130
## 3  108      TRUE 3  3 -1.419507 24.83901 36.84048 44.59930
## 4   66      TRUE 4  4  6.047073 17.90585 34.27149 34.61842
## 5  112      TRUE 5  5 10.243921 17.51216 38.24196 38.56285
## 6  123      TRUE 5  6 10.243918 17.51216 38.24197 38.56289
```

```
# Aikaselittäjän malli toisella siemenluvulla
set.seed(12)
aika.flexmix <- stepFlexmix(khajonta ~ aika, k = 2:6, nrep=10, dat = ro59)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * * * * * *Error in FLXfit(model = model, concomitant = concomitant, control = control,  :
##    165 Log-likelihood: NaN
##  *
## 6 : * * * * * * * * *Error in FLXfit(model = model, concomitant = concomitant, control = control,  :
##    112 Log-likelihood: NaN
##  *
```

```
aika.flexmix # Kohdan 59 informaatiokriteerit, aikaselittäjän malli toisella siemenellä
```

```
##
## Call:
## stepFlexmix(khajonta ~ aika, dat = ro59, k = 2:6, nrep = 10)
##
##   iter converged k k0    logLik      AIC      BIC      ICL
## 2   71      TRUE 2  2 -4.836356 23.67271 31.31001 44.39272
## 3   85      TRUE 3  3 -1.664132 25.32826 37.32973 45.08383
## 4  117      TRUE 4  4  5.813108 18.37378 34.73942 36.52076
## 5  108      TRUE 5  5 10.243920 17.51216 38.24197 38.56287
## 6   85      TRUE 5  6 10.244579 17.51084 38.24065 38.55506
```

```
# Rataosuus 289
ro289 <- rautatie.khajonnat.all[289,]
ro289 <- data.frame(matrix(data= c(1:22, as.numeric(t(ro289))), ncol = 2))
colnames(ro289) <- c("aika", "khajonta")

# Vakioselittäjän mixture-malli
set.seed(1)
vakio.flexmix <- stepFlexmix(khajonta ~ 1, k = 2:6, nrep=10, dat = ro289)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * * * * * * *
## 6 : * * * * * * * * * *
```

```
vakio.flexmix # Kohdan 289 informaatiokriteerit, vakioselittäjän malli
```

```
##
## Call:
## stepFlexmix(khajonta ~ 1, dat = ro289, k = 2:6, nrep = 10)
##
##   iter converged k k0    logLik       AIC       BIC          ICL
## 2  117      TRUE 2  2  9.258896 -8.517791 -3.062579  2.617414909
## 3   73      TRUE 3  3 12.866022 -9.732043 -1.003704 -0.008470742
## 4   74      TRUE 4  4 12.866022 -3.732044  8.269423 13.937413140
## 5   95      TRUE 5  5 12.866031  2.267938 17.542533 27.207701149
## 6   98      TRUE 6  6 14.656258  4.687483 23.235205 55.626318014
```

```
# Aikaselittäjän mixture-malli
set.seed(1)
aika.flexmix <- stepFlexmix(khajonta ~ aika, k = 2:6, nrep=10, dat = ro289)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : *Error in FLXfit(model = model, concomitant = concomitant, control = control,  :
##    155 Log-likelihood: NaN
##  * * * * * * * * *
## 6 : * * * * * * * * * *
```

```
aika.flexmix # Kohdan 289 informaatiokriteerit, aikaselittäjän malli
```

```
##
## Call:
## stepFlexmix(khajonta ~ aika, dat = ro289, k = 2:6, nrep = 10)
##
##   iter converged k k0   logLik       AIC       BIC       ICL
## 2   16      TRUE 2  2 33.24274 -52.48549 -44.84819 -42.70653
## 3   15      TRUE 3  3 33.24362 -44.48724 -32.48578 -27.39498
## 4   63      TRUE 4  4 35.81365 -41.62730 -25.26166 -14.05061
## 5  200     FALSE 4  5 35.99285 -41.98570 -25.62007 -17.77231
## 6   99      TRUE 4  6 41.03842 -52.07685 -35.71121 -29.78896
```

```
# Rataosuus 307
ro307 <- rautatie.khajonnat.all[307,]
ro307 <- data.frame(matrix(data= c(1:22, as.numeric(t(ro307))), ncol = 2))
colnames(ro307) <- c("aika", "khajonta")

# Vakioselittäjän mixture-malli
set.seed(1)
vakio.flexmix <- stepFlexmix(khajonta ~ 1, k = 2:6, nrep=10, dat = ro307)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * * * * * * *
## 6 : * * * * * * * * * *
```

```
vakio.flexmix # Kohdan 307 informaatiokriteerit, vakioselittäjän malli
```

```
##
## Call:
## stepFlexmix(khajonta ~ 1, dat = ro307, k = 2:6, nrep = 10)
##
##   iter converged k k0   logLik       AIC       BIC       ICL
## 2   17      TRUE 2  2 11.17250 -12.34500 -6.889792 -6.889723
## 3   59      TRUE 3  3 13.88284 -11.76568 -3.037340 -1.683749
## 4   56      TRUE 3  4 13.87638 -11.75276 -3.024417 -1.384703
## 5   93      TRUE 3  5 13.87637 -11.75275 -3.024408 -1.384559
## 6   97      TRUE 3  6 13.87637 -11.75275 -3.024408 -1.384560
```

```
# Aikaselittäjän mixture-malli
set.seed(1)
aika.flexmix <- stepFlexmix(khajonta ~ aika, k = 2:6, nrep=10, dat = ro307)
```

```
## 2 : * * * * * * * * * *
## 3 : * * * * * * * * * *
## 4 : * * * * * * * * * *
## 5 : * * * * * * * * * *
## 6 : * *Error in FLXfit(model = model, concomitant = concomitant, control = control,  :
##     38 Log-likelihood: NaN
##  * * * * * * * *
```

```
aika.flexmix # Kohdan 307 informaatiokriteerit, aikaselittäjän malli
```

```
##
## Call:
## stepFlexmix(khajonta ~ aika, dat = ro307, k = 2:6, nrep = 10)
##
##   iter converged k k0   logLik       AIC       BIC       ICL
## 2   35      TRUE 2  2 34.35196 -54.70391 -47.06662 -46.68692
## 3   34      TRUE 3  3 43.58810 -65.17620 -53.17474 -52.21410
## 4   83      TRUE 4  4 52.82662 -75.65325 -59.28761 -57.41887
## 5   44      TRUE 4  5 48.98395 -67.96790 -51.60226 -49.36317
## 6   39      TRUE 4  6 56.20972 -82.41943 -66.05379 -65.15699
```