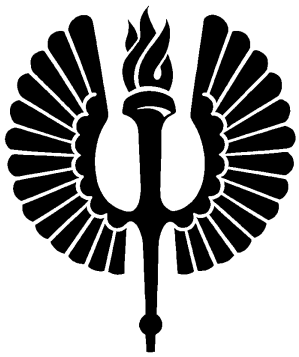


Ostotapahtuman tekevien pelaajien määrän ennustaminen elinaika-analyysia ja EM-algoritmia soveltamalla

Riikka Numminen

Pro gradu -tutkielma
Toukokuu 2018

MATEMATIIKAN JA TILASTOTIETEEN LAITOS
TURUN YLIOPISTO

TURUN YLIOPISTO
Matematiikan ja tilastotieteen laitos

NUMMINEN, RIIKKA: Ostotapahtuman tekevien pelaajien määrän ennustaminen elinaika-analyysia ja EM-algoritmia soveltamalla
Pro gradu -tutkielma, 61 s.
Sovellettu matematiikka
Toukokuu 2018

Tässä työssä tutkitaan menetelmää, jolla pyritään ennustamaan rajoitetulla aikavälillä kerätystä mobiilipelien pelaajadatasta ostotapahtuman tekevien pelaajien määrä. Kirjallisuuskatsauksessa esitellään elinaika-analyysin käsitteitä ja yleisimpiä malleja sekä johdettavan mallin optimiratkaisun selvittämiseen käytettävä Expectation Maximization -algoritmi. Ostotapahtuman tekevien pelaajien määrän ennustamisessa käytetään elinaika-analyysin mixture cure -mallia. Teoriaosuuden lopuksi johdetaan mallille funktiot, jotka tarvitaan EM-algoritmin käyttämiseen. Menetelmä implementoidaan R-ohjelmointikielellä.

Mallia käytetään Hipster Sheep -nimisestä pelistä [1] kerättyyn dataan ja se toimii pääasiassa odotusten mukaisesti. Joillain harvoilla otoksilla havaittiin ongelma mixture cure -mallin ja eksponenttijakauman toisistaan erottamisessa. Ongelman aiheuttavasta datasta ei pystytä erottamaan, tekevätkö kaikki pelaajat ostotapahtuman (eksponenttijakauma hitaalla ostotahdilla) vai onko datassa pelaajia, jotka eivät sitä tee, vaikka niitä seurattaisiin äärettömän pitkään (mixture cure -malli kohtuullisella ostotahdilla).

Lisäksi tutkitaan generoidulla datalla otoskoon ja ostotapahtuman tekevien pelaajien määrän vaikutusta mallin toimivuuteen. Huomataan, että malli toimii sitä paremmin, mitä enemmän pelaajia on, mitä useampi pelaaja tekee ostotapahtuman ja mitä isompi osa ostotapahtuman tekevistä pelaajista tekee sen tarkasteluhetkeen mennessä. Liian pienellä otoksella ja liian pienellä ostotapahtuman tekevien pelaajien sekä tarkasteluhetkeen mennessä tehtyjen ostotapahtumien määrällä mallin antama ratkaisu vaihtelee melko paljon otoksesta riippuen. Yhteenvedossa esitetään kehitysidea, jolla saadaan poistettua edellä mainitut ongelmat.

Avainsanat: mixture cure model, EM-algoritmi, pelidata.

Sisältö

1 Johdanto

Tämän työn tavoite on tutkia elinaika-analyysissa käytetyn mixture cure -mallin toimivuutta pelialalta kerättyyn dataan. Elinaika-analyysi on tilastotieteen menetelmä, jolla pyritään ennustamaan lopputulos rajoitetulla aikavälillä kerätystä datasta. Alttiit yksilöt tekevät tarkastelun kohteena olevan tapahtuman äärellisellä seuranta-ajalla ja immuunit yksilöt eivät sitä tee, vaikka niitä seurattaisiin äärettömän pitkään. Pelidata, johon tätä sovelletaan, on netistä puhelimeen ilmaiseksi ladattavasta pelistä kerättyjä tietoja. Osa pelien ominaisuuksista on saatavilla vain jos niistä maksaa.

Tarkoituksena on ennustaa ostotapahtuman tekevien pelaajien määrä, kun käytössä on rajoitetulla aikavälillä kerättyä dataa. Uusien pelaajien saaminen edellyttää pelin mainostamista, joten mainostuskulujen minimoimiseksi halutaan mahdollisimman lyhyellä seuranta-ajalla pystyä ennustamaan pelin tuottavuus pitkällä aikavälillä. Käytettävässä mallissa on kaksi selittävää muuttujaa: ostavien pelaajien määrä ja ostotahti.

Pelialan data on muodoltaan samantapaista kuin muutenkin elinaika-analyysin sovelluskohteissa käytetty data. Data koostuu pelaajista, joista osa on tehnyt ostotapahtuman ja osa ei. Pelaajat, jotka eivät ole ostotapahtumaa tehneet, ovat sensuroituja. Sensuroiduista pelaajista ei kuitenkaan eroteta ostotapahtuman myöhemmin tekeviä pelaajia ostotapahtuman tekemiselle immuuneista pelaajista. Pelaajien jako immuuneihin ja alttiisiin yksilöihin on siis latentti muuttuja. Osa alttiista pelaajista tekee useita ostotapahtumia, mutta tässä tutkielmassa keskitytään ensimmäisen ostotapahtuman tekevien pelaajien määrään eikä kaiken kaikkiaan tehtävien ostotapahtumien määrään.

Ennen varsinaisen käsiteltävän mallin johtamista käydään läpi sen muotoilemiseen tarvittavaa elinaika-analyysin ja Expectation Maximization -algoritmin teoriaa. Elinaika-analyysista esitellään siihen liittyviä käsitteitä ja tärkeimpiä malleja. Tämän jälkeen käydään läpi vaillinaisen datan tilanteeseen soveltuvan optimointialgoritmin toiminta. Data on vaillinaista, kun siinä on latentteja muuttujia, joiden arvoja ei tiedetä. Latentteja muuttujia sisältävän mallin optimaalisen parametrivektorin selvittämiseen käytettävä optimointimenetelmä on nimeltään EM-algoritmi.

Kirjallisuuskatsauksen jälkeen johdetaan tässä tutkielmassa käytettävälle mallille EM-algoritmin käyttämiseen tarvittavat funktiot. Tämän lisäksi saatu algoritmi toteutetaan R-ohjelmointikielellä (R), jolla havainnollistetaan algoritmin konvergenssia. Mallin toimivuutta testataan oikeasta pelistä kerätyllä datalla sekä mallin oletuksien mukaisesti generoidulla datalla. Todellisena datana käytetään Hipster sheep -nimisestä pelistä [1] kerättyä dataa. Lopuksi vielä tutkitaan generoidulla datalla datan ominaisuuksien vaikutus-

ta siihen, miten suurimman uskottavuuden estimaatti konvergoi todelliseen arvoonsa.

2 Elinaika-analyysi

Aloitetaan tämän tutkielman teoriaosuus perehtymällä elinaika-analyysiin. Käydään ensin läpi tärkeimpiä peruskäsitteitä ja ominaisuuksia. Jatketaan sitten käytetyimpien parametristen ja epäparametristen mallien esittelemiseen ja keskitytään lopuksi vielä elinaika-analyysin osa-alueen cure-mallien ja erityisesti mixture cure -mallin perusteisiin.

Elinaika-analyysi on tilastotieteen ala, jossa tarkastellaan yksilöitä, jotka suorittavat tarkastelun kohteena olevan tapahtuman jollakin ajanhetkellä. Tällaisesta tapahtumasta käytetään nimitystä *vikaantuminen*. Aikaa, joka kuluu ennen kuin tapahtuma tehdään, kutsutaan *vikaantumisajaksi*. Elinaika-analyysissä jokainen yksilö voi tehdä tarkastelun kohteena olevan tapahtuman enintään kerran. [2]

Tarkastelun kohteena olevan tapahtuman tyyppi riippuu alasta, johon elinaika-analyysiä sovelletaan. Tästä syystä tapahtuma voi kuvata hyvinkin erilaisia asioita. Data voi olla kerätty lääketieteellisestä tutkimuksesta, jolloin tapahtuma voi olla potilaan kuolema, tai jos kerätty data on teollisuudesta, niin tapahtuma voi olla esimerkiksi jonkin laitteen tietyn osan hajoaminen. [2]

Elinaika-analyysin menetelmät riippuvat *välttöjakaumasta*, joka voidaan määritellä välttöfunktion ja riskifunktion avulla. Olkoon satunnaismuuttuja T vikaantumisaika. Merkitään, että sen tiheysfunktio on $f(t)$ ja kertymäfunktio on $F(t)$. Tällöin voidaan määritellä *välttöfunktio*

$$S(t) = 1 - F(t) = P(T > t), \quad 0 < t < \infty, \tag{1}$$

joka määrittää todennäköisyyden, että vikaantuminen tapahtuu vasta ajanhetken t jälkeen [3]. Todennäköisyysfunktiona välttöfunktio luonnollisestikin saa arvoja väliltä [0,1]. Välttöfunktion arvo ajanhetkellä $t = 0$ on 1, sillä kertymäfunktion arvo $F(0) = 0$, ja välttöfunktio on vähenevä tai vakio ajan funktiona. Välttöfunktio määritellään usein riskifunktion avulla. Riskifunktio

$$\begin{aligned} h(t) &= \frac{f(t)}{1 - F(t)} \\ &= \lim_{\Delta t \to 0} \frac{P(t < T < t + \Delta t \mid T > t)}{\Delta t} \end{aligned} \tag{2}$$

kuvaa todennäköisyyttä, että vikaantuminen tapahtuu seuraavan ajanjakson Δt aikana, kun yksilö on selvinnyt ajanhetkeen t asti. [4]

Integroimalla riskifunktio muodossa (2) saadaan *kumulatiivinen riskifunktio*

$$H(t) = \int_0^t h(u)\,\mathrm{d}u = \int_0^t \frac{f(u)}{1-F(u)}\mathrm{d}u = -\int_0^t \log\left[1-F(u)\right] \\ = -\log\left[1-F(t)\right] = -\log S(t), \tag{3}$$

joka on riskifunktion kertymäfunktio ja täten kuvaa riskifunktion alle jäävää pinta-alaa ajanhetkeen t asti tarkasteltuna. Kaavan (3) integraalia laskettaessa tulee muistaa, että tiheysfunktio $f(t)$ on kertymäfunktion $F(t)$ derivaatta. Kaavasta (3) ratkaisemalla $S(t)$ saadaan, että välttöfunktio voidaan kirjoittaa muodossa

$$S(t) = e^{-\int_0^t h(u)\mathrm{d}u} = e^{-H(t)}. \tag{4}$$

Yhtälö (4) osoittaa riskifunktion yhteyden välttöfunktioon. Yhtälö (2) on mahdollista kirjoittaa myös muodossa, jossa käytetään välttöfunktiota $S(t)$ riskifunktion $h(t)$ määrittelemiseen. Tällöin riskifunktio lasketaan yhtälön

$$h(t) = -\frac{\mathrm{d}S(t)}{\mathrm{d}t}\frac{1}{S(t)} \tag{5}$$

avulla. [4]

Vikaantumisajan tarkkaan määrittämiseen tarvitaan kolme ehtoa: ajan nollakohdan tulee olla yksikäsitteinen, ajankulun mitta-asteikko pitää olla sovittuna ja vikaantumisen määritelmän tulee olla täysin selvä. Ajan nollakohta pitää olla tarkkaan määritelty jokaiselle yksilölle. Yleensä se ei ole kalenterin mukaisessa ajassa sama jokaiselle yksilölle. Esimerkiksi lääketieteellisessä tutkimuksessa jokaisella potilaalla on oma nollakohtansa, joka kuvaa esimerkiksi ajanhetkeä, jolloin potilaalla todetaan tarkasteltavana oleva sairaus. Tällaisessa tapauksessa vikaantumisaika tarkoittaa aikaa sairauden toteamisesta potilaan kuolemaan kyseisestä sairaudesta johtuen. [2]

Tästä huomataan myös vikaantumisen tarkkaan määrittelemisen tärkeys. Edellisessä esimerkissä olisi voitu todeta, että vikaantumisaika on aika sairauden toteamisesta potilaan kuolemaan. Tällöin vikaantumista ei olisi tarkkaan määritelty, sillä potilas voi diagnoosin saamisen jälkeen kuolla muustakin syystä kuin tutkittavana olevasta sairaudesta johtuen. [2]

Ajan etenemisen mitta-asteikkona käytetään usein reaaliaikaa, mutta on olemassa muitakin asteikkoja ajan etenemisen mittaamiseen. Esimerkiksi teollisuudessa aikaa mitataan usein kumulatiivisena käyttönä. Mitta-asteikon tulee kuitenkin olla sellainen, että vikaantumisaika saa vain positiivisia arvoja. Sen pitää olla jokaiselle yksilölle sama myös siitä syystä, että kahden samanlaisen yksilön pitäisi olla samanlaisessa tilassa, kun kumpaakin on seurattu yhtä kauan. [2]

Vikaantumisajan tarkan määrittelemisen lisäksi toinen olennainen piirre elinaika-analyysissä on *sensurointi*. Sensuroinnilla tarkoitetaan sitä, että ajan nollakohtaa $t = 0$ tai tapahtumahetkeä T ei pystytä tarkkaan määrittämään; aloitustapahtuma tai vikaantuminen on siis tapahtunut havainnointiajan ulkopuolella. Sensurointi voi olla joko *informatiivista* tai *epäinformatiivista*. Sensurointi on informatiivista, kun seurannasta poistuminen johtuu vikaantumiseen liittyvistä syistä, ja epäinformatiivista, kun se johtuu muusta kuin vikaantumiseen liittyvästä syystä. [4]

Yleisin sensurointitapa on *oikealta sensurointi*, joka tarkoittaa, että tapahtumahetkeä T^* ei tiedetä tarkkaan, mutta sen tiedetään olevan myöhemmin kuin ajanhetki C. Ajanhetki C kuvaa hetkeä, jona sensurointi tapahtuu, eli hetkeä, jolloin yksilön seuraaminen lopetetaan. Tällöin havaittu aika on $T = \min(T^*, C)$ ja muuttuja $\delta = \mathbb{I}(T^* \leq C)$ on *sensurointi-indeksi*, joka kuvaa, onko sensurointi tapahtunut vai ei. Muuttujan δ määritelmässä käytetyn *indikaattorifunktion* $\mathbb{I}(A)$ arvo on 1, kun A on tosi, ja 0, kun se on epätosi. Vastaavasti on mahdollista, että ajan nollakohtaa $t = 0$ ei tiedetä tarkasti. Tällöin kyseessä on *vasemmalta sensurointi*. [4]

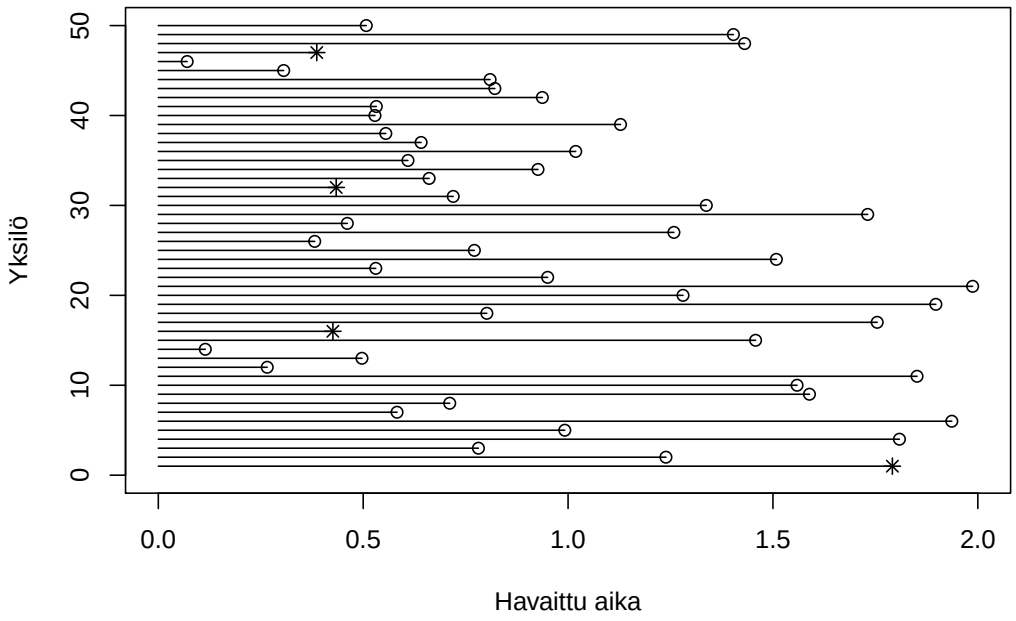
Sensurointi voidaan myös jakaa kolmeen tyyppiin: tyypin I, tyypin II ja satunnainen sensurointi. Tyypin I sensuroinnissa sensurointiaika C on päätetty etukäteen ja se on sama jokaiselle yksilölle. Tyypin II sensuroinnissa sensurointiajankohtaa ei päätetä vaan yksilöitä seurataan siihen asti, että etukäteen päätetty prosentti on vikaantunut. Satunnaisessa sensuroinnissa jokaiselle yksilölle i on oma sensurointiaikansa c_i. Sensurointiajat ovat toisistaan riippumattomia ja samoin jakautuneita. Satunnaisen sensuroinnin tapauksessa havaitaan siis vikaantumisaikojen t_i^* sijaan parit (t_i, δ_i), joissa

$$t_i = \min(t_i^*, c_i) \quad \text{ja} \tag{6}$$
$$\delta_i = \mathbb{I}(t_i^* \leq c_i). \tag{7}$$

[3]

Havainnollistetaan sensuroinnin vaikutusta kuvalla 1, josta nähdään, että osaa yksilöistä ei ole seurattu koko havainnointiaikaa, sillä ne ovat vikaantuneet jo aiemmin. Vikaantuneille yksilöille havaittu aika on tietysti lyhempi kuin sensurointiin asti seuratuille yksilöille

Elinaika-analyysissä havainnot noudattavat usein jotakin todennäköisyysjakaumaa, jonka muotoa kuvaillaan parametreilla $\boldsymbol{\theta}$. Käytetyimmät jakaumat esitellään osiossa 2.1. *Uskottavuusfunktio* $L(\boldsymbol{\theta})$ kuvaa jakauman parametrien



Kuva 1: Esimerkki sensuroidusta 50 yksilön datasta.

$\boldsymbol{\theta}$ todennäköisyyttä ja elinaika-analyysissa se on muotoa

$$\begin{aligned} L(\boldsymbol{\theta}) &= \prod_{i=1}^{n} L_i(\boldsymbol{\theta}) \\ &= \prod_{i=1}^{n} f(t_i;\boldsymbol{\theta},\boldsymbol{x}_i)^{\delta_i} S(t_i;\boldsymbol{\theta},\boldsymbol{x}_i)^{1-\delta_i}, \end{aligned} \tag{8}$$

missä aika t_i on kaavan (6) mukainen havaittu aika, sensurointi-indeksi δ_i on yhtälön (7) määräämä arvo ja muuttuja $\boldsymbol{x}_i$ on muut havaitut arvot. Parametrien arvoja, jotka maksimoivat uskottavuusfunktion, kutsutaan *suurimman uskottavuuden estimaatiksi*. Suurimman uskottavuuden estimoinnissa määritettävät parametrien arvot suppenevat todellisiin arvoihin [5]. Suurimman uskottavuuden estimaattia $\hat{\boldsymbol{\theta}}$ laskettaessa usein maksimoidaan logaritmista

uskottavuusfunktiota

$$\begin{aligned} l(\boldsymbol{\theta}) &= \log L(\boldsymbol{\theta}) = \log \left\{ \prod_{i=1}^{n} L_i(\boldsymbol{\theta}) \right\} \\ &= \sum_{i=1}^{n} \log L_i(\boldsymbol{\theta}) \\ &= \sum_{i=1}^{n} \{\delta_i \log f(t_i; \boldsymbol{\theta}, \boldsymbol{x}_i) + (1 - \delta_i) \log S(t_i; \boldsymbol{\theta}, \boldsymbol{x}_i)\} \end{aligned} \tag{9}$$

uskottavuusfunktion (8) sijasta. [6]

2.1 Parametriset mallit

Vikaantumisajan T jakauma noudattaa usein jonkinlaista mallia. Tunnistettavien jakaumien pohjalta on kehitetty useita parametrisia malleja, joilla kuvataan eloonjäämisprosessia. Esitellään seuraavaksi kirjan [6] mukaisesti yleisimmin käytetyt parametriset mallit, jotka ovat eksponentti-, Weibull-, gamma-, log-normaali-, log-logistinen ja Gompertz-jakauma.

2.1.1 Eksponenttijakauma

Aloitetaan esittelemällä eksponenttijakauma, sillä se on parametrisista malleista yksinkertaisin. Eksponenttijakauman määrittämiseen riittää yksi parametri, jota merkitään kirjaimella λ. Siitä käytetään nimitystä *skaalausparametri* ja se on vakiovikaantumisnopeus. Eksponenttijakauman välttöfunktio

$$S(t; \lambda) = e^{-\lambda t}. \tag{10}$$

Välttöfunktion muodosta (10) ja riskifunktion kaavasta (5) saadaan johdettua, että riskifunktio

$$h(t; \lambda) = \lambda. \tag{11}$$

Kumulatiivinen riskifunktio $H(t; \lambda)$ on tällöin kaavan (3) perusteella muotoa

$$H(t; \lambda) = \lambda t.$$

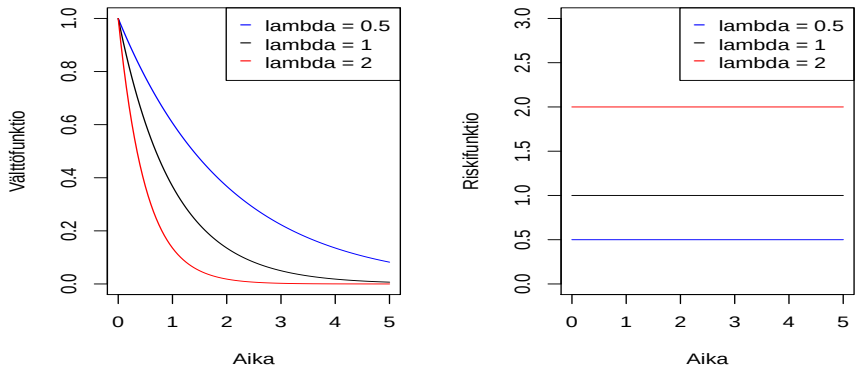
Eksponenttijakauman tiheysfunktio $f(t)$ voidaan määritellä myös yhtälöiden (10) ja (11) avulla:

$$f(t; \lambda) = h(t; \lambda) S(t; \lambda) = \lambda e^{-\lambda t}. \tag{12}$$

Jakauman kertymäfunktio $F(t)$ puolestaan saadaan laskettua yhtälöstä (1), jolloin saadaan, että

$$F(t;\lambda) = 1 - S(t;\lambda) = 1 - e^{-\lambda t}.$$

Kuvassa 2 havainnollistetaan parametrin λ vaikutusta välttöfunktion (10) muotoon. Huomataan, että välttöfunktio on sitä loivempi, mitä pienempi muuttujan λ arvo on. Riskifunktion (11) muotoon parametrin λ arvon muuttaminen ei tietenkään vaikuta riskifunktion ollessa vakio.



Kuva 2: Skaalausparametrin λ vaikutus eksponenttijakauman välttö- ja riskifunktioihin.

Eksponenttijakauman välttöfunktion logaritmi on lineaarinen ajan suhteen

$$\log S(t;\lambda) = -\lambda t.$$

Tätä ominaisuutta voidaan hyödyntää kokeellisen datan kanssa, kun halutaan graafisesti selvittää, noudattaako se eksponenttijakaumaa.

Eksponenttijakauman toinen tärkeä ominaisuus on muistittomuus. Jakauman muistittomuus tarkoittaa sitä, että

$$P(T > \Delta t) = P(T > t + \Delta t \mid T > t),$$

eli todennäköisyys selvitä vielä aika Δt ei muutu vaikka yksilön tiedettäisiin jo selvinneen aika t. Jokaisesta tarkasteluhetkestä katsottuna on siis sama todennäköisyys selvitä aika Δt. Elinaika-analyysissä eksponenttijakauman käyttö tarkoittaa, että yksilön jäljellä olevan elinajan odotusarvo on sama minkä tahansa ikäisenä. Elinaika-analyysin prosesseissa riskifunktio ei yleensä ole vakio, joten eksponenttijakauman käyttö elinaika-analyysissä on rajoitettua. Sitä kuitenkin käytetään tilanteissa, joissa tarkastellaan vikaantumisia lyhyellä aikavälillä.

2.1.2 Weibull-jakauma

Esitellään seuraavaksi Weibull-jakauma, joka on eksponenttijakauman tapaan yksinkertainen, mutta mukautuva. Weibull-jakauma määritellään kahdella parametrilla, joten se soveltuu useampaan tilanteeseen kuin eksponenttijakauma. Skaalausparametrin λ lisäksi Weibull-jakaumaa määrittää muotoparametri α.

Vikaantumisajan T noudattaessa Weibull-jakaumaa (merkitään $T \sim \text{Weib}(\lambda, \alpha)$) välttöfunktio on muotoa

$$S(t; \lambda, \alpha) = \exp\left[-(\lambda t)^{\alpha}\right]. \tag{13}$$

Vastaavasti kuin eksponenttijakauman tapauksessa, riskifunktio $h(t)$ saadaan johdettua välttöfunktion muodosta (13) ja riskifunktion määritelmästä (5):

$$h(t; \lambda, \alpha) = \lambda\alpha(\lambda t)^{\alpha-1}. \tag{14}$$

Kumulatiivinen riskifunktio $H(t; \lambda, \alpha)$ saadaan laskettua kaavalla (3). Weibull-jakauman tapauksessa

$$H(t; \lambda, \alpha) = (\lambda t)^{\alpha}. \tag{15}$$

Kuten eksponenttijakaumankin tapauksessa, myös Weibull-jakaumaa käytettäessä pystytään testaamaan, noudattaako data oletettua jakaumaa. Weibull-jakauman tapauksessa se tapahtuu ottamalla logaritmi kumulatiivisesta riskifunktiosta (15). Kumulatiivisen riskifunktion logaritmin $\log H(t; \lambda, \alpha)$ arvot ajan t logaritmin funktiona muodostavat suoran, sillä

$$\log H(t; \lambda, \alpha) = \log\lambda + \alpha \log t.$$

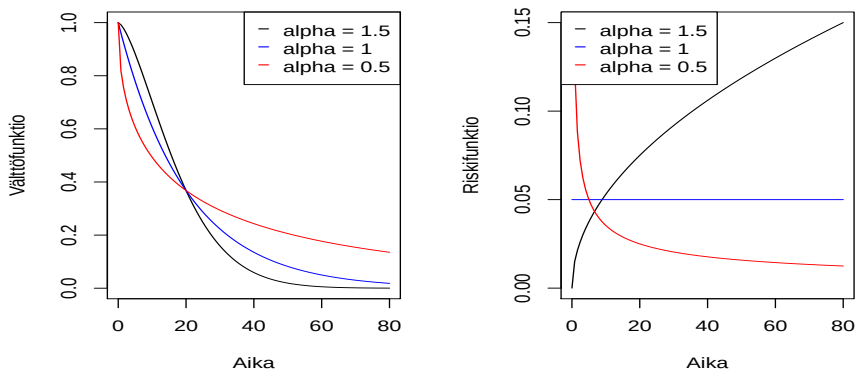
Jakauman tiheysfunktio saadaan jälleen välttöfunktion (13) ja riskifunktion (14) tulona

$$f(t; \lambda, \alpha) = h(t)\, S(t) = \lambda\alpha(\lambda t)^{\alpha-1} \exp\left[-(\lambda t)^{\alpha}\right]$$

ja kertymäfunktio saadaan välttöfunktion (13) avulla

$$F(t; \lambda, \alpha) = 1 - \exp\left[-(\lambda t)^{\alpha}\right].$$

Kuvassa 3 havainnollistetaan muotoparametrin α vaikutusta välttö- ja riskifunktioiden muotoon. Huomataan, että eksponenttijakauma on Weibull-jakauman erikoistapaus, kun $\alpha = 1$. Parametrin α arvoilla $\alpha > 1$ riskifunktio on epälineaarisesti kasvava ajan t funktiona ja arvoilla $\alpha < 1$ huomataan riskifunktion olevan epälineaarisesti vähenevä ajan t funktiona. Weibull-jakauman tapauksessa siis riskifunktio on joko monotonisesti kasvava tai vähenevä. Tämä onkin eräs Weibull-jakauman tärkeistä ominaisuuksista.



Kuva 3: Muotoparametrin α vaikutus Weibull-jakauman välttö- ja riskifunktioihin.

2.1.3 Gammajakauma

Esitellään kolmantena parametrisena mallina gammajakauma. Sitä voidaan käyttää mallinnettaessa sellaista eloonjäämisprosessia, joka ei noudata mitään symmetristä jakaumaa. Gammajakauma määritellään kahden parametrin avulla samoin kuin Weibull-jakauma. Merkinnällä $\Gamma(\alpha)$ tarkoitetaan gammafunktiota ja se määritellään kaavan (16) mukaisena integraalina

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} \mathrm{d}t. \tag{16}$$

Gamma-jakauman tiheysfunktio on muotoa

$$f[t; T \sim \Gamma(\lambda, \alpha)] = \frac{\lambda (\lambda t)^{\alpha-1} e^{-\lambda t}}{\Gamma(\alpha)}, \quad t > 0. \tag{17}$$

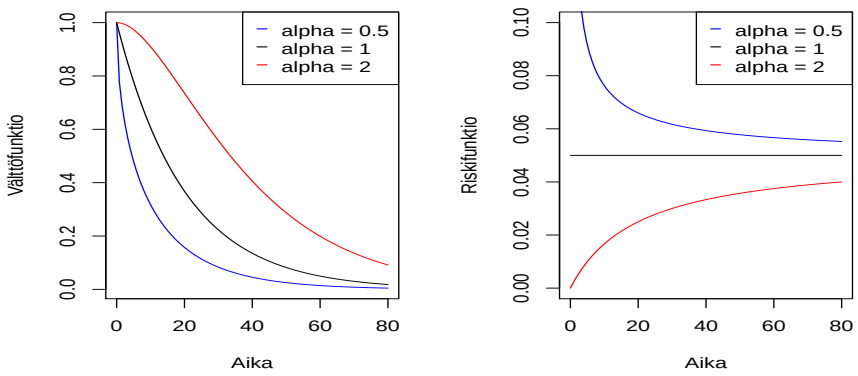
Jälleen huomataan, että jos $\alpha = 1$, niin gammajakauma redusoituu eksponenttijakaumaksi, joten eksponenttijakauma on myös gammajakauman erikoistapaus. Tämä nähdään myös kuvasta 4, jossa havainnollistetaan gammajakauman välttö- ja riskifunktioita erilaisilla muotoparametrin α arvolla.

Keskitytään seuraavaksi erikoistapaukseen, jossa $\lambda = 1$. Tällöin tiheysfunktio (17) saadaan kirjoitettua yksiparametrisen gammajakauman tiheysfunktiona

$$f[t; T \sim \Gamma(\alpha)] = \frac{t^{\alpha-1} e^{-t}}{\Gamma(\alpha)}, \quad t > 0.$$

Yksiparametrisen gammajakauman kertymäfunktiota

$$F[t; T \sim \Gamma(\alpha)] = \int_0^t \frac{u^{\alpha-1} e^{-u}}{\Gamma(\alpha)} \mathrm{d}u$$



Kuva 4: Muotoparametrin α vaikutus gammajakauman välttö- ja riskifunktioihin.

kutsutaan myös epätäydelliseksi gammafunktioksi. Gamma-jakauman tapauksessa välttöfunktio ja riskifunktio määritellään epätäydellisen gammafunktion avulla:

$$\begin{aligned} S\left[t; T \sim \Gamma\left(\alpha\right)\right] &= 1 - F\left[t; T \sim \Gamma\left(\alpha\right)\right] \\ h\left[t; T \sim \Gamma\left(\alpha\right)\right] &= \frac{\lambda\left(\lambda t\right)^{\alpha-1} e^{-\lambda t}}{\Gamma\left(\alpha\right)\left\{1 - F\left[t; T \sim \Gamma\left(\alpha\right)\right]\right\}}. \end{aligned}$$

2.1.4 Log-normaalijakauma

Myös log-normaalijakauma on yksi suosituista elinaika-analyysissa käytetyistä parametrisista malleista. Log-normaalijakauman käyttökelpoisuus perustuu siihen, että sen logaritmi on normaalijakautunut parametrein μ ja σ^2. Eli jos elinaika-analyysissä vikaantumisaika T noudattaa log-normaalijakaumaa (merkitään $T \sim LN\left(\mu, \sigma^2\right)$), niin vikaantumisajan T logaritmi $\log T$ noudattaa normaalijakaumaa, eli voidaan merkitä $\log T \sim N\left(\mu, \sigma^2\right)$.

Log-normaalijakauman tiheysfunktio on muotoa

$$f\left[t; T \sim LN\left(\mu, \sigma\right)\right] = \frac{1}{t\sigma\sqrt{2\pi}} \exp\left\{-\frac{\left(\log t - \mu\right)^2}{2\sigma^2}\right\}$$

ja sen kertymäfunktio voidaan lausua normaalijakauman kertymäfunktion Φ avulla:

$$F\left[t; T \sim LN\left(\mu, \sigma\right)\right] = \Phi\left(\frac{\log t - \mu}{\sigma}\right) = \Phi\left(\frac{\log t}{\sigma}\right), \quad t > 0. \tag{18}$$

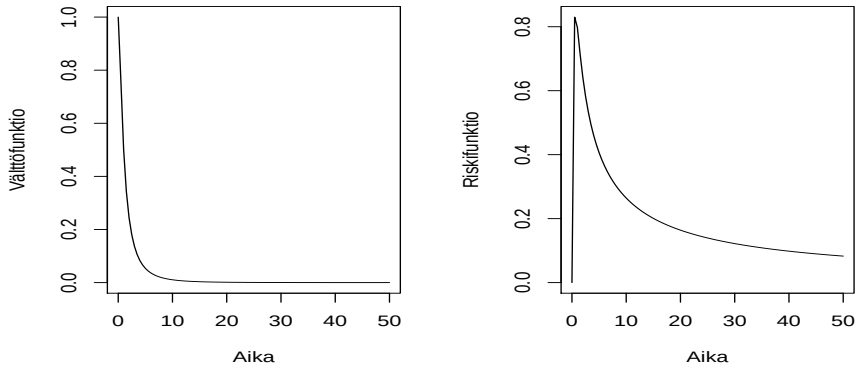
Jälleen saadaan esitettyä välttöfunktio kertymäfunktion (18) avulla:

$$S\left[t; T \sim LN\left(\mu, \sigma\right)\right] = 1 - \Phi\left(\frac{\log t}{\sigma}\right).$$

Myös riskifunktio ja kumulatiivinen riskifunktio voidaan kirjoittaa normaalijakauman tiheys- ja kertymäfunktioiden φ ja Φ avulla:

$$\begin{aligned} h\left[t; T \sim LN\left(\mu, \sigma\right)\right] &= \frac{\frac{1}{t\sigma}\varphi\left(\frac{\log t}{\sigma}\right)}{\Phi\left(\frac{-\log t}{\sigma}\right)} \\ H\left[t; T \sim LN\left(\mu, \sigma\right)\right] &= -\log\left[1 - \Phi\left(\frac{\log t}{\sigma}\right)\right]. \end{aligned}$$

Kuvassa 5 näytetään esimerkit log-normaalijakauman välttö- ja riskifunktioiden muodoista parametrien arvoilla $\mu = 0$ ja $\sigma = 1$.



Kuva 5: Havainnollistetaan log-normaalijakauman välttö- ja riskifunktioiden muotoja parametrien arvoilla $\mu = 0$ ja $\sigma = 1$.

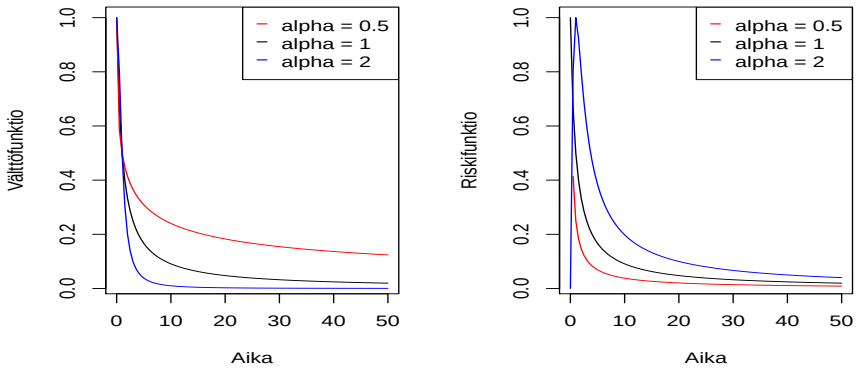
2.1.5 Log-logistinen jakauma

Seuraavana on vuorossa log-logistinen jakauma, joka on muodoltaan melko samanlainen kuin log-normaalijakauma. Log-logistisen jakauman ero log-normaalijakaumaan on kuitenkin merkittävä. Log-normaalijakauma sopii vain tilanteisiin, joissa ei ole tapahtunut sensurointia. Toinen log-logistisen jakauman tärkeä ominaisuus on se, että sen kertymä-, välttö- ja riskifunktiot pystytään esittämään suljetussa muodossa.

Log-logistista jakaumaa kuvataan kahdella parametrilla, λ ja α. Vikaantumisajan T noudattaessa log-logistista jakaumaa merkitään $T \sim LLogist\left(\lambda, \alpha\right)$.

Allaolevassa listauksessa esitellään log-logistisen jakauman tiheys-, kertymä-, välttö- ja riskifunktiot, ja kuvassa 6 on esimerkit tämän jakauman välttö- ja riskifunktioista.

$$
\begin{aligned}
f\left[t; T \sim LLogist\left(\lambda, \alpha\right)\right] &= \frac{\left(\alpha/\lambda\right)\left(t/\lambda\right)^{\alpha-1}}{\left[1+\left(t/\lambda\right)^{\alpha}\right]^{2}}, \quad t > 0 \\
F\left[t; T \sim LLogist\left(\lambda, \alpha\right)\right] &= \frac{1}{1+\left(t/\lambda\right)^{-\alpha}} \\
S\left[t; T \sim LLogist\left(\lambda, \alpha\right)\right] &= \frac{1}{1+\left(t/\lambda\right)^{\alpha}} \\
h\left[t; T \sim LLogist\left(\lambda, \alpha\right)\right] &= \frac{\left(\alpha/\lambda\right)\left(t/\lambda\right)^{\alpha-1}}{1+\left(t/\lambda\right)^{\alpha}}
\end{aligned}
$$



Kuva 6: Esimerkki log-logistisen jakauman välttö- ja riskifunktioista.

2.1.6 Gompertz-jakauma

Viimeisenä parametrisena mallina tässä tutkielmassa esitellään Gompertz-jakauma. Gompertzin laki on ehkä yleisimmin käytetty ihmisen kuolevuutta ja eloonjäämistä kuvaileva malli. Gompertz-jakaumassa riskifunktio on muotoa

$$h\left(t; h_0, r\right) = h_0 e^{rt}, \quad h_0, r > 0. \tag{19}$$

Parametria h_0 kutsutaan usein iästä riippumattomaksi riskikertoimeksi ja parametri r puolestaan on iästä riippuva riskikerroin. Yhtälön (4) mukaisesti integroimalla riskifunktio (19) saadaan välttöfunktioksi

$$S\left(t; h_0, r\right) = \exp\left[-\int_0^t h\left(u; h_0, r\right) \mathrm{d}u\right] = \exp\left[\frac{h_0}{r}\left(1-e^{rt}\right)\right]. \tag{20}$$

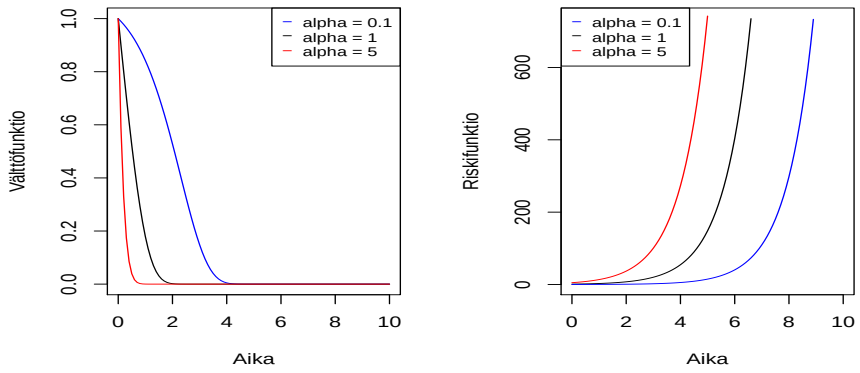
Gompertzin lain mukaisen jakauman tiheysfunktio saadaan funktioiden (19) ja (20) tulona

$$f\left(t; h_0, r\right) = h_0 e^{rt} \exp\left[\frac{h_0}{r}\left(1 - e^{rt}\right)\right]$$

ja kertymäfunktio saadaan jälleen ratkaistua kaavasta (1), jolloin

$$F\left(t; h_0, r\right) = 1 - \exp\left[\frac{h_0}{r}\left(1 - e^{rt}\right)\right].$$

Havainnollistetaan vielä tätäkin jakaumaa kuvaajilla. Kuvassa 7 esitetään Gompertz-jakauman välttö- ja riskifunktiot erilaisilla parametrin arvoilla.



Kuva 7: Esimerkit Gompertz-jakauman välttö- ja riskifunktioista.

2.2 Epäparametriset mallit

Käydään seuraavaksi lyhyesti läpi teoriaa epäparametrisista malleista. Epäparametrisissa malleissa ei nimensä mukaisesti ole jakaumaa kuvaavia parametreja, vaan estimointi perustuu suoraan havaitusta datasta laskettaviin arvoihin.

Parametrisia malleja käytettäessä täytyy pystyä määrittämään vikaantumisajan T jakauman muotoa kuvaava funktio. Epäparametristen mallien käyttö on joustavampaa, sillä niitä käytettäessä ei ole tällaisia rajoituksia. Epäparametrista mallia voidaan käyttää myös sovituksen hyvyyden graafisessa tarkastelussa. [2]

Yleisimmin käytetty epäparametrinen malli on Kaplan–Meier-malli, jota usein kutsutaan myös rajatuloestimaatiksi. Tässä osiossa esiteltävä teoria pohjautuu artikkeliin [7], mutta yhdenmukaisuuden vuoksi käytetään osion 2.1 merkintätapoja. Tärkeimpiä näistä ovat välttöfunktio $S(t)$, joka määritellään kuten yhtälössä (1), välttöfunktion rajatuloestimaatti $\hat{S}(t)$ ja funktio $n(t)$, joka kuvaa ajanhetkeä t pidempään selviävien ja havaittavien yksilöiden määrää. Funktion $n(t)$ arvoa laskettaessa ei oteta huomioon hetkellä t tapahtuvia vikaantumisia.

Rajatuloestimaatin laskeminen perustuu seuraaviin vaiheisiin:

1. Jaetaan aika sopivasti valittuihin aikaväleihin $(0, u_1), (u_1, u_2), ..., (u_{r-1}, u_r)$

2. Lasketaan jokaiselle aikavälille (u_{j-1}, u_j) osamäärä $\hat{s}_j = \dfrac{S_j}{S_{j-1}}$, joka kertoo hetkeä u_j pidempään selviävien yksilöiden osuuden hetkeä u_{j-1} pidempään selviävistä yksilöistä

3. Lasketaan aikavälin rajapistettä t pidemmälle selviävien yksilöiden välttöfunktion arvo tulona
$$S(t) = \prod_{u_j \leq t} s_j.$$

Vaiheessa 1 aikavälit (u_{j-1}, u_j) valitaan siten, että vaiheen 2 osamäärät saadaan laskettua kaavalla

$$\hat{s}_j = \frac{n_j - d_j}{n_j} = \frac{n'_j}{n_j}. \tag{21}$$

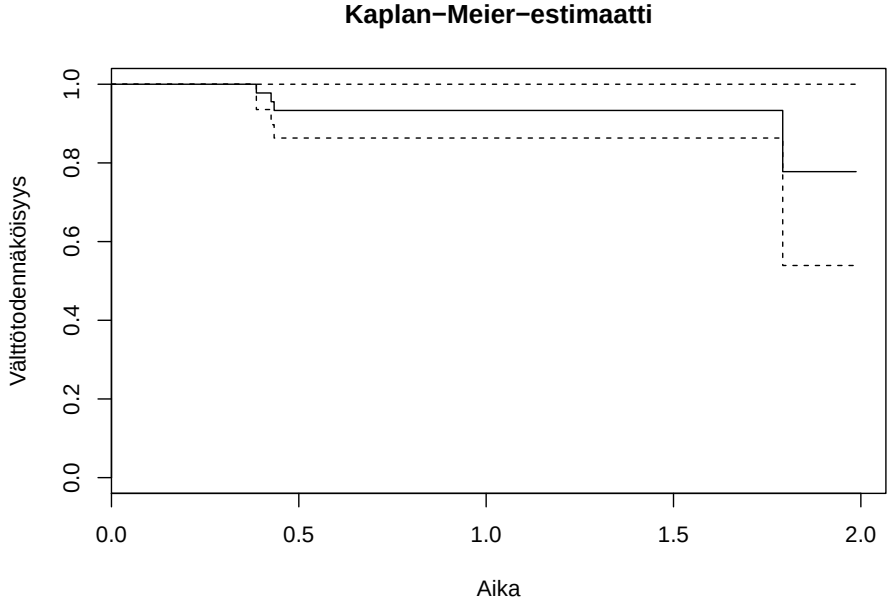
Kaavassa (21) muuttuja n_j on heti hetken u_{j-1} jälkeen havaittujen yksilöiden määrä, muuttuja d_j on aikavälillä (u_{j-1}, u_j) vikaantuvien yksilöiden määrä ja muuttuja n'_j on havaittujen yksilöiden määrä aikavälillä tapahtuvien vikaantumisten jälkeen. Vaiheessa 3 ei ole väliä sensurointien jakautumisesta

aikavälillä (u_{j-1}, u_j) kunhan kaavassa (21) esitetyt n_j ja n'_j on laskettu oikein. Vaiheessa 3 oletetaan myös, että sensurointi tapahtuu joko ennen aikavälin ensimmäistä vikaantumista tai vasta viimeisen vikaantumisen jälkeen ennen aikavälin päättymistä.

Määrä n_j on laskettu oikein, kun siihen ei ole laskettu mukaan yksilöitä, jotka vikaantuvat ajanhetkellä u_j. Siispä jokaisella ajanhetkellä u_j pätee epäyhtälöt $n_j^{'} \leq n\left(u_j\right) \leq n_{j+1}$. Tämä tarkoittaa, että ajanhetkeä u_j pidempään selviävien ja havaittavien yksilöiden määrä $n\left(u_j\right)$ on vähintään yhtäsuuri kuin aikavälillä (u_{j-1}, u_j) tapahtuvien vikaantumisten jälkeen havaittava yksilöiden määrä n'_j ja enintään yhtäsuuri kuin heti ajanhetken u_{j+1} jälkeen havaittujen yksilöiden määrä n_{j+1}. Tällöin rajatuloestimaatti on muotoa

$$\hat{S}\left(t\right) = \prod_{j=1}^{k} \frac{n'_j}{n_j},$$

missä $u_k = t$ ja n'_j on määritelty samoin kuin kaavassa (21). Kaplan–Meierin rajatuloestimaattia laskettaessa data on oikealta sensuroitua ja vasemmalta katkaistua, eli yksilöistä, jotka vikaantuvat ennen havaintoajan alkamista, ei tiedetä mitään.



Kuva 8: Esimerkki Kaplan–Meier-estimaatista kuvan 1 datalle.

2.3 Mixture cure -malli

Esitellään vielä tämän luvun lopuksi cure-malli, minkä jälkeen pystytään määrittelemään sovellusosiossa käytettävä mixture cure -malli. Cure-malli on elinaika-analyysimalli, jota käytetään tilanteissa, joissa osa datan yksilöistä on immuuneja tarkastelun kohteena olevan tapahtuman realisoitumiselle. Tällaiset yksilöt eivät siis vikaannu, vaikka niitä seurattaisiin äärettömän pitkä aika. Matemaattisesti esitettynä tilanne, jossa cure-mallia voidaan käyttää, on sellainen, että välttöfunktion (1) raja-arvo on nollasta eroava eli

$$\lim_{t\to\infty} S(t) \neq 0.$$

Datan sopivuutta cure-mallin sovelluskohteeksi voidaan arvioida tarkastelemalla esimerkiksi osiossa 2.2 esitetyn Kaplan–Meierin rajatuloestimaatin $\hat{S}(t)$ raja-arvoa. Kuvan 8 perusteella kuvan 1 data vaikuttaa olevan esimerkki tällaisesta datasta. [8]

Cure-mallin kohdalla osiossa 2 selitetty oikealta sensurointi voi tapahtua kahdesta eri syystä: yksilö on immuuni, eikä siis missään vaiheessa vikaannu, tai yksilö on altis, mutta sitä ei ole vielä seurattu riittävän pitkään. Joissain tilanteissa on mahdollista määrittää aika, johon mennessä kaikki alttiit yksilöt vikaantuvat. Aina tämä ei kuitenkaan ole mahdollista ja silloin immuunien ja alttiiden yksilöiden toisistaan erottaminen on mahdotonta. Cure-mallin kohdalla on kahdenlaisia yksilöitä, joten on luonnollista ajatella sitä kahden ryhmän sekoitemallina. Tällaisesta mallista käytetään nimitystä mixture cure -malli. [8]

Merkitään tässäkin osiossa vikaantumisaikaa muuttujalla T, ja määritellään, että muuttuja ζ kuvaa yksilön immuuniutta. Yksilön ollessa immuuni tarkastelun kohteena olevan tapahtuman realisoitumiselle, alttiusindeksi $\zeta = 0$, ja muussa tapauksessa $\zeta = 1$. Kummallekin ryhmälle voidaan määritellä oma välttöfunktio, joka on ehdollinen todennäköisyys, että yksilö vikaantuu vasta tarkasteluhetkeä myöhemmin. Nämä ovat muotoa

$$\begin{aligned} S_u(t) &= P(T > t|\zeta = 1) \quad \text{ja} \\ S_c(t) &= P(T > t|\zeta = 0) \equiv 1. \end{aligned}$$

Erottelussa käytetyt alaindeksit u ja c tulevat sanoista ”uncured” ja ”cured” vastaavassa järjestyksessä. Immuunien yksilöiden populaation välttöfunktio $S_c(t) \equiv 1$, sillä siihen kuuluvat yksilöt eivät missään vaihesssa vikaannu, joten todennäköisyys $P(T > t) = 1$ millä tahansa t. Koko populaation marginaalinen välttöfunktio on muotoa

$$\begin{aligned} S(t) &= \pi S_u(t) + (1-\pi) S_c(t) \\ &= \pi S_u(t) + 1 - \pi, \end{aligned} \tag{22}$$

kun yksilön vikaantumistodennäköisyys $P(\zeta = 1) = \pi$. [8]

Mixture cure -malli on siis parametrinen malli, jossa vikaantumisaika T noudattaa jotakin osiossa 2.1 esiteltyä jakaumaa. Tämän lisäksi mallia selittää parametri π, joka on alttiiden yksilöiden osuus koko populaatiosta. Tämän mallin suurimman uskottavuuden estimaatin ratkaiseminen on ongelmallista tavallisilla optimointimenetelmillä, sillä malli sisältää latentteja muuttujia.

3 Expectation Maximization -algoritmi

Esitellään seuraavaksi osion 2.3 mallin parametrien optimiarvojen ratkaisemiseen sopiva optimointimenetelmä. Artikkelissa [9] Dempster, Laird ja Rubin käyttivät siitä nimeä Expectation Maximization -algoritmi (EM-algoritmi). Esitellään ensin algoritmiin liittyviä käsitteitä ja sen toimintaan liittyvä teoria pääasiassa kirjaan [5] pohjautuen. Tämän jälkeen perehdytään algoritmin konvergenssin todistamiseen pääasiassa artikkelin [10] mukaisesti. Yhdenmukaisuuden vuoksi koko luvussa käytetään kirjassa [5] käytettyjä merkintöjä.

3.1 Algoritmin toiminta

EM-algoritmi on iteroiva algoritmi uskottavuusfunktion suurimman arvon selvittämiseksi. Sitä käytetään usein tilanteissa, joissa on esimerkiksi puuttuvaa dataa ja muiden optimointialgoritmien käyttö olisi tästä syystä hankalaa. EM-algoritmia voidaan myös käyttää tilanteissa, joissa datan epätäydellisyys ei ole selvää. Esimerkki tällaisesta tilanteesta on tapaus, jossa on latentteja muuttujia.

EM-algoritmia käytettäessä on havaittu data $\boldsymbol{y}$ ja tiedetään, että on dataa, jota ei syystä tai toisesta ole voitu havaita. Puuttuvaa dataa kuvataan vektorilla $\boldsymbol{z}$. Datasta käytetään nimitystä *täydellinen data*, kun puuttuvaa dataa ei ole, ja sitä merkitään muuttujalla $\boldsymbol{x}$. Osa taulukkoon 1 kerätyistä funktioista on taulukossa vain havaitulle datalle $\boldsymbol{y}$, mutta vastaavat funktiot ovat olemassa myös täydelliselle datalle $\boldsymbol{x}$. Täydellistä dataa käytettäessä funktiot tunnistaa alaindeksistä c.

Algoritmin nimi kuvaa suoraan sen toimintaa, sillä se muodostuu kahdesta vaiheesta, joita kutsutaan nimillä E-askel (expectation) ja M-askel (maximization). Näitä vaiheita toistetaan vuorotellen, kunnes saavutetaan optimiarvo. Ensimmäisessä vaiheessa (E-askel) puuttuvan datan arvoja $\boldsymbol{z}$ arvioidaan laskemalla niille ehdolliset odotusarvot tunnettujen arvojen $\boldsymbol{y}$ avulla. Toisessa vaiheessa (M-askel) selvitetään uskottavuusfunktion $L(\boldsymbol{\Psi})$ maksimoiva parametrivektori $\hat{\boldsymbol{\Psi}}$, suurimman uskottavuuden estimaatti, E-askeleessa laskettuja arvioita hyödyntämällä. Ensin siis lasketaan arviot puuttuvalle datalle ja sen jälkeen niitä hyödynnetään täydellisen datan uskottavuusfunktion maksimoinnissa. Tämän jälkeen saatua suurimman uskottavuuden estimaattia käyttämällä lasketaan uudet arviot puuttuvalle datalle.

Täydellisen datan uskottavuusfunktiota $L_c(\boldsymbol{\Psi})$ kuvaa täydellistä dataa $\boldsymbol{x}$ vastaavan satunnaismuuttujan X todennäköisyysjakauman tiheysfunktio $g_c(\boldsymbol{x};\boldsymbol{\Psi})$, jossa vektori $\boldsymbol{\Psi} = (\Psi_1, ..., \Psi_d)^T$ koostuu jakauman parametrien arvoista. Laskennallisista syistä käytetään usein logaritmista uskottavuusfunktiota $l_c(\boldsymbol{\Psi}) = \log L_c(\boldsymbol{\Psi}) = \log g_c(\boldsymbol{x};\boldsymbol{\Psi})$. Logaritmifunktio on monotoninen,

joten logaritminen uskottavuusfunktio saa suurimman arvonsa samoilla parametrien $\boldsymbol{\Psi}$ arvoilla kuin uskottavuusfunktio. Havaitun datan $\boldsymbol{y}$ tiheysfunktio $g(\boldsymbol{y};\boldsymbol{\Psi})$ on muotoa $g(\boldsymbol{y};\boldsymbol{\Psi}) = \prod_{j=1}^{n} f(\boldsymbol{w}_j;\boldsymbol{\Psi})$, kun havaittu data $\boldsymbol{y} = \left(\boldsymbol{w}_1^T, ..., \boldsymbol{w}_n^T\right)^T$ ja n-kokoisen satunnaisotannan havainnot noudattavat satunnaisvektorin W todennäköisyysjakaumaa $f(\boldsymbol{w};\boldsymbol{\Psi})$.

Taulukko 1: EM-algoritmin ominaisuuksista kerrottaessa käytetyt merkinnät.

$\boldsymbol{x}$	Täydellistä dataa kuvaava vektori
$\boldsymbol{y}$	Havaittua dataa kuvaava vektori
$\boldsymbol{z}$	Puuttuvaa dataa kuvaava vektori
$\boldsymbol{X}$	Täydellistä dataa vastaava satunnaismuuttuja
$\boldsymbol{Y}$	Havaittua dataa vastaava satunnaismuuttuja
$g(\boldsymbol{y};\boldsymbol{\Psi})$	Satunnaismuuttujan $\mathbf{Y}$ todennäköisyysjakauman tiheysfunktio
$\boldsymbol{\Psi} = (\Psi_1, ..., \Psi_d)^T$	Satunnaismuuttujan $\boldsymbol{Y}$ tiheysfunktion parametrivektori
$\boldsymbol{\Omega}$	Parametrivektorin määrittelyjoukko
$L(\boldsymbol{\Psi}) = g(\boldsymbol{y};\boldsymbol{\Psi})$	Uskottavuusfunktio
$l(\boldsymbol{\Psi})$	Logaritminen uskottavuusfunktio
$k(\boldsymbol{x}\|\boldsymbol{y};\boldsymbol{\Psi})$	Satunnaismuuttujan $\boldsymbol{X}$ ehdollinen tiheysfunktio
$\boldsymbol{\Psi}^{(k)}$	Parametrivektori k iteraatiokierroksen jälkeen
$Q\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right)$	Täydellisen datan uskottavuusfunktion ehdollinen odotusarvo
$\hat{\boldsymbol{\Psi}}$	Parametrivektorin $\boldsymbol{\Psi}$ suurimman uskottavuuden estimaatti

Varsinaisesti EM-algoritmia käytettäessä on siis kaksi otosavaruutta $\mathcal{X}$ ja $\mathcal{Y}$ sekä joukosta pisteeseen -kuvaus otosavaruudesta $\mathcal{X}$ otosavaruuteen $\mathcal{Y}$. Pystytään havaitsemaan vain epätäydellisen datan vektori $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{x})$ joukossa $\mathcal{Y}$ sen sijasta, että havaittaisiin täydellisen datan vektori $\boldsymbol{x}$ avaruudessa $\mathcal{X}$. Havaitun datan jakauman tiheysfunktio voidaan tällöin kirjoittaa muodossa

$$g(\boldsymbol{y};\boldsymbol{\Psi}) = \int_{\mathcal{X}(\boldsymbol{y})} g_c(\boldsymbol{x};\boldsymbol{\Psi})\, d\boldsymbol{x},$$

missä $\mathcal{X}(\boldsymbol{y})$ on otosavaruuden $\mathcal{X}$ osajoukko, jonka määrää yhtälö $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{x})$.

EM-algoritmi pyrkii selvittämään epätäydellisen datan logaritmisen uskottavuusfunktion $\log L(\boldsymbol{\Psi})$ maksimin $\hat{\boldsymbol{\Psi}}$ iteratiivisesti hyödyntämällä täydellisen datan logaritmista uskottavuusfunktiota $\log L_c(\boldsymbol{\Psi})$. Sitä ei kuitenkaan tiedetä, joten käytetään sen ehdollista odotusarvoa, kun tiedetään havaittu data $\boldsymbol{y}$ ja parametrien senhetkiset arvot $\boldsymbol{\Psi}^{(k)}$. Tällöin EM-algoritmin vaiheet ovat seuraavaa muotoa:

E-askel: Lasketaan täydellisen datan logaritmisen uskottavuusfunktion ehdollinen odotusarvo

$$Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}\right) = E_{\boldsymbol{\Psi}^{(k)}}\left\{\log L_c(\boldsymbol{\Psi}) \,|\boldsymbol{y}\right\}. \tag{23}$$

M-askel: Etsitään $\boldsymbol{\Psi}^{(k+1)} \in \boldsymbol{\Omega}$, joka maksimoi funktion $Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}\right)$, eli jolle

$$Q\left(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}\right) \geq Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}\right) \tag{24}$$

kaikilla $\boldsymbol{\Psi} \in \boldsymbol{\Omega}$.

Yleistetyssä EM-algoritmissa (GEM-algoritmi) maksimointivaiheessa valitaan parametrivektori $\boldsymbol{\Psi}^{(k+1)}$, jolla funktion Q arvo on aiemman parametrivektorin $\boldsymbol{\Psi}^{(k)}$ tuottamaa arvoa suurempi, mutta ei edellytetä sen maksimoivan funktiota Q. Eli M-askel on muotoa: etsitään $\boldsymbol{\Psi}^{(k+1)} \in \boldsymbol{\Omega}$, joka toteuttaa yhtälön

$$Q\left(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}\right) \geq Q\left(\boldsymbol{\Psi}^{(k)}; \boldsymbol{\Psi}^{(k)}\right).$$

EM-algoritmilla on monia hyviä ominaisuuksia verrattuna muihin algoritmeihin. Osa näistä ominaisuuksista osoitetaan osiossa 3.2. Mainittakoon ensimmäisenä ominaisuutena, että EM-algoritmi on parantava algoritmi, sillä uskottavuusfunktion arvo kasvaa jokaisella iteraatiokierroksella (ks. lause 3). Toinen tärkeä ominaisuus on, että algoritmin konvergenssi pystytään osoittamaan melko yleisten ehtojen ollessa voimassa. Osiossa 3.2 osoitetaan algoritmin konvergenssi lokaaliin optimiin.

EM-algoritmi on usein helppo implementoida, sillä siinä käytetään täydellistä dataa ja laskuissa ei tarvitse laskea uskottavuusfunktion arvoa eikä sen derivaattoja. EM-algoritmin suorittamiseen ei tarvita paljoa muistia, mikä on hyvä asia. EM-algoritmia käytettäessä iteraatiokierrosten määrä voi olla suuri, mutta yhden kierroksen suorittaminen on melko nopeaa, joten algoritmi ei suuresta iteraatiokierrosten määrästä huolimatta ole muita optimointimenetelmiä hitaampi.

EM-algoritmia käytettäessä tarvittavien funktioiden johtaminen on usein helpompi kuin muita menetelmiä käytettäessä, sillä EM-algoritmissa mak-

simoidaan vain täydellisen datan logaritmisen uskottavuusfunktion ehdollinen odotusarvo. E-askeleessakaan se ei yleensä ole kovinkaan monimutkaista. EM-algoritmin konvergenssia on helppo tarkkailla seuraamalla uskottavuusfunktion arvon kasvua. Tällä menetelmällä pystytään myös laskemaan jonkinlaiset uskottavat arviot puuttuvalle datalle.

Esitellään osion lopuksi vielä määritelmässä 1 toinen tapa merkitä EM-algoritmin M-askeleessa olevaa epäyhtälöä (24) ja käytetään sitä suljetun kuvauksen määritelmässä 2.

Määritelmä 1. Funktion $Q\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right)$ maksimoivien pisteiden joukko on

$$\boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right) = \arg\max_{\boldsymbol{\Psi}\in\boldsymbol{\Omega}} Q\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right).$$

Määritelmä 2. Pisteeltä joukolle -kuvaus $\boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right)$ on *suljettu* pisteessä $\boldsymbol{\Psi} = \boldsymbol{\Psi}_0$, jos ehdoista

$$\begin{aligned} \boldsymbol{\Psi}^{(k)} &\to \boldsymbol{\Psi}_0, & \boldsymbol{\Psi}^{(k)} &\in \boldsymbol{\Omega} \\ \boldsymbol{\phi}^{(k)} &\to \boldsymbol{\phi}_0, & \boldsymbol{\phi}^{(k)} &\in \boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right) \end{aligned}$$

seuraa, että $\boldsymbol{\phi}_0 \in \boldsymbol{M}\left(\boldsymbol{\Psi}_0\right)$. Kuvaus on *suljettu joukossa* $\boldsymbol{\Omega}$, jos se on suljettu jokaisessa pisteessä $\boldsymbol{\Psi} \in \boldsymbol{\Omega}$. [11]

3.2 Algoritmin konvergenssi

Nyt kun EM-algoritmin toimintaperiaate ja siihen liittyvät käsitteet on käyty läpi, voidaan keskittyä algoritmin konvergenssin osoittamiseen. Tässä osiossa esitettävät lauseet ja niiden todistukset ovat pääasiassa lähteestä [10]. Osoitetaan, että sekä uskottavuusfunktion $L\left(\boldsymbol{\Psi}\right)$ että parametrivektorin $\boldsymbol{\Psi}$ arvot konvergoivat. Aloitetaan uskottavuusfunktion konvergenssista, jonka avulla sitten osoitetaan parametrien konvergenssi. Osion lopussa käydään vielä läpi EM-algoritmin konvergenssiasteen laskeminen.

3.2.1 Uskottavuusfunktion arvo suppenee

Aloitetaan uskottavuusfunktion konvergenssin osoittaminen todistamalla, että millä tahansa EM-algoritmin iteraatiokierroksella saatava uskottavuusfunktion arvo $L\left(\boldsymbol{\Psi}^{(k)}\right)$ on vähintään yhtäsuuri kuin edellisellä iteraatiokierroksella saatu uskottavuusfunktion arvo $L\left(\boldsymbol{\Psi}^{(k-1)}\right)$. EM-algoritmin monotonisuus on alun perin esitetty artikkelissa [9], mutta lauseen 3 todistuksessa käytetään kirjassa [5] esitetyn todistuksen mukaisia merkintöjä.

Lause 3. (EM-algoritmin monotonisuus) Uskottavuusfunktion arvo ei pienene, kun käytetään EM-algoritmia uuden parametrivektorin $\boldsymbol{\Psi}$ laskemiseen. Eli jokaisella $k = 0, 1, 2, ...$ pätee

$$L\left(\boldsymbol{\Psi}^{(k+1)}\right) \geq L\left(\boldsymbol{\Psi}^{(k)}\right). \tag{25}$$

Todistus. Määritellään ehdollisen tiheysfunktion $k\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}\right)$ olevan muotoa

$$k\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}\right) = \frac{g_c\left(\boldsymbol{x};\boldsymbol{\Psi}\right)}{g\left(\boldsymbol{y};\boldsymbol{\Psi}\right)}, \tag{26}$$

kun tiedetään havaittu data $\boldsymbol{y}$ ja tämänhetkiset parametrien arvot $\boldsymbol{\Psi}$. Havaitun datan uskottavuusfunktio saadaan muotoon

$$\begin{aligned} \log L\left(\boldsymbol{\Psi}\right) &= \log g\left(\boldsymbol{y};\boldsymbol{\Psi}\right) \\ &= \log g_c\left(\boldsymbol{x};\boldsymbol{\Psi}\right) - \log k\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}\right) \\ &= \log L_c\left(\boldsymbol{\Psi}\right) - \log k\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}\right) \end{aligned} \tag{27}$$

ottamalla yhtälöstä (26) puolittain logaritmi ja sijoittamalla saatu lauseke logaritmisen uskottavuusfunktion taulukossa 1 esitettyyn määritelmään. Ottamalla odotusarvot yhtälön (27) kummaltakin puolelta, saadaan uskottavuusfunktio muotoon

$$\begin{aligned} \log L\left(\boldsymbol{\Psi}\right) &= E_{\boldsymbol{\Psi}^{(k)}}\left\{\log L_c\left(\boldsymbol{\Psi}\right)|\boldsymbol{y}\right\} - E_{\boldsymbol{\Psi}^{(k)}}\left\{\log k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}\right)|\boldsymbol{y}\right\} \\ &= Q\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right) - \mathcal{H}\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right), \end{aligned} \tag{28}$$

missä funktiolla $\mathcal{H}$ on merkitty täydellisen datan ehdollisen tiheysfunktion logaritmin odotusarvoa ehdoilla $\boldsymbol{y}$ ja $\boldsymbol{\Psi}^{(k)}$. Nyt yhtälön (28) perusteella

$$\begin{aligned} \log L\left(\boldsymbol{\Psi}^{(k+1)}\right) - \log L\left(\boldsymbol{\Psi}^{(k)}\right) = &\left\{Q\left(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}\right) - Q\left(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)}\right)\right\} \\ &- \left\{\mathcal{H}\left(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}\right) - \mathcal{H}\left(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)}\right)\right\}. \end{aligned} \tag{29}$$

Yhtälön (29) oikeanpuolen ensimmäisen termin tiedetään olevan ei-negatiivinen kaavan (24) perusteella, joten epäyhtälö (25) toteutuu, kun

$$\mathcal{H}\left(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}\right) - \mathcal{H}\left(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)}\right) \leq 0. \tag{30}$$

Epäyhtälö (30) on tosi, sillä jokaisella $\boldsymbol{\Psi}$

$$
\begin{aligned}
&\mathcal{H}\left(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}\right)-\mathcal{H}\left(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)}\right)\\
&\quad=E_{\boldsymbol{\Psi}^{(k)}}\left\{\log k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}\right)|\boldsymbol{y}\right\}-E_{\boldsymbol{\Psi}^{(k)}}\left\{\log k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}^{(k)}\right)|\boldsymbol{y}\right\}\\
&\quad=E_{\boldsymbol{\Psi}^{(k)}}\left\{\log k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}\right)-\log k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}^{(k)}\right)|\boldsymbol{y}\right\}\\
&\quad=E_{\boldsymbol{\Psi}^{(k)}}\left\{\log\left[\frac{k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}\right)}{k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}^{(k)}\right)}\right]\Big|\boldsymbol{y}\right\}\\
&\quad\leq\log\left[E_{\boldsymbol{\Psi}^{(k)}}\left\{\frac{k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}\right)}{k\left(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi}^{(k)}\right)}\right\}\Big|\boldsymbol{y}\right]\\
&\quad=\log\int_{\mathcal{X}(\boldsymbol{y})}k\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}^{(k)}\right)d\boldsymbol{x}\\
&\quad=0.
\end{aligned}
$$

Siispä uskottavuusfunktion arvo uusilla parametrien arvoilla $\boldsymbol{\Psi}^{(k+1)}$ on vähintään yhtäsuuri kuin edellisellä iteraatiokierroksella lasketuilla parametrien arvoilla $\boldsymbol{\Psi}^{(k)}$ ja täten uskottavuusfunktion arvojen jono $\left\{L\left(\boldsymbol{\Psi}^{(k)}\right)\right\}$ on kasvava. □

EM-algoritmin monotonisuuden lisäksi vaaditaan, että uskottavuusfunktion arvojen jono $\left\{L\left(\boldsymbol{\Psi}^{(k)}\right)\right\}$ on rajoitettu, jotta sen tiedetään suppenevan arvoon L^*. Arvo L^* on uskottavuusfunktion ääriarvokohta, mikäli $L^*=L\left(\boldsymbol{\Psi}^*\right)$ kriittisessä pisteessä $\boldsymbol{\Psi}^*$. Piste $\boldsymbol{\Psi}^*$ on *kriittinen piste*, jos se on yhtälön $\frac{\partial L(\boldsymbol{\Psi})}{\partial\boldsymbol{\Psi}}=\mathbf{0}$ ratkaisu.

Oletuksista **OL1**, **OL2** ja **OL3** seuraa suoraan, että uskottavuusfunktion arvojen jono $\left\{L\left(\boldsymbol{\Psi}^{(k)}\right)\right\}$ on ylhäältä rajoitettu millä tahansa parametrivektorin alkuarvauksella $\boldsymbol{\Psi}_0\in\boldsymbol{\Omega}$.

OL1 Parametrien määrittelyjoukko $\boldsymbol{\Omega}$ on r-ulotteisen euklidisen avaruuden $\mathbb{R}^r$ osajoukko.

OL2 Parametrivektorin alkuarvauksen määrittelemä avaruuden $\boldsymbol{\Omega}$ osajoukko $\boldsymbol{\Omega}_{\boldsymbol{\Psi}_0}=\left\{\boldsymbol{\Psi}\in\boldsymbol{\Omega}:L\left(\boldsymbol{\Psi}\right)\geq L\left(\boldsymbol{\Psi}_0\right)\right\}$ on kompakti millä tahansa $L\left(\boldsymbol{\Psi}_0\right)>-\infty$.

OL3 Funktio L on jatkuva määrittelyjoukossa $\boldsymbol{\Omega}$ ja differentioituva määrittelyjoukon $\boldsymbol{\Omega}$ sisäosassa.

Nyt tiedetään oletukset, joiden tulee olla voimassa, jotta uskottavuusfunktion arvojen jono suppenee. Lemmassa 4 esitetään lisää oletuksia, joiden avulla uskottavuusfunktion arvojen jonon $\left\{L\left(\boldsymbol{\Psi}^{(k)}\right)\right\}$ kriittiseen pisteeseen

suppeneminen EM-algoritmin tuottamissa pisteissä $\boldsymbol{\Psi}^{(k)}$ todistetaan lauseessa 6. Merkitään lokaalien maksimien joukkoa kirjaimella $\mathcal{M}$ ja kriittisten pisteiden joukkoa kirjaimella $\mathcal{S}$.

Lemma 4. Olkoon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ jono GEM-algoritmilla laskettuja arvoja $\boldsymbol{\Psi}^{(k+1)} \in \boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right)$. Oletetaan, että kuvaus $\boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right)$ on suljettu joukon $\boldsymbol{\Omega}$ kriittisten pisteiden joukon $\mathcal{S}$ komplementissa $\overline{\mathcal{S}}$, ja että

$$L\left(\boldsymbol{\Psi}^{(k+1)}\right) > L\left(\boldsymbol{\Psi}^{(k)}\right), \text{ kaikilla } \boldsymbol{\Psi}^{(k)} \notin \mathcal{S}.$$

Tällöin jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ suppenee aina kriittiseen pisteeseen $\boldsymbol{\Psi}^* \in \mathcal{S}$ ja $L\left(\boldsymbol{\Psi}^{(k)}\right)$ suppenee monotonisesti pisteeseen $L^* = L\left(\boldsymbol{\Psi}^*\right)$.

Lemman 5 tulos on riittävä ehto sille, että EM-algoritmissa käytetty kuvaus $\boldsymbol{M}$ on suljettu. Lauseen 6 todistuksessa käytetään tätä ehtoa ja lemmaa 4, kun osoitetaan, että uskottavuusfunktion arvo konvergoi ääriarvokohtaan $L\left(\boldsymbol{\Psi}^*\right)$ jollakin kriittisellä pisteellä $\boldsymbol{\Psi}^*$. Lauseen 6 todistuksessa käytetty merkintä $D^{1,0}$ tarkoittaa ensimmäistä derivaattaa ensimmäisen parametrivektorin suhteen.

Lemma 5. (Jatkuvuusehto) EM-algoritmin kuvaus $\boldsymbol{M}$ on suljettu, jos funktio $Q\left(\boldsymbol{\Psi}^{'}; \boldsymbol{\Psi}\right)$ on jatkuva sekä pisteessä $\boldsymbol{\Psi}^{'}$ että pisteessä $\boldsymbol{\Psi}$.

Lause 6. Oletetaan, että funktio Q täyttää lemmassa 5 esitetyn jatkuvuusehdon. Tällöin kaikki EM-algoritmilla lasketun jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteet ovat uskottavuusfunktion $L\left(\boldsymbol{\Psi}\right)$ kriittisiä pisteitä ja uskottavuusfunktion arvot $L\left(\boldsymbol{\Psi}^{(k)}\right)$ suppenevat monotonisesti arvoon $L^* = L\left(\boldsymbol{\Psi}^*\right)$ jollakin kriittisellä pisteellä $\boldsymbol{\Psi}^*$.

Todistus. Lemman 5 jatkuvuusehto on riittävä osoittamaan lemman 4 ensimmäisen oletuksen voimassaolo, joten riittää osoittaa lemman 4 toinen oletus kaikille pisteille $\boldsymbol{\Psi}^{(k)} \notin \mathcal{S}$. Valitaan parametrivektori $\boldsymbol{\Psi}^{(k)}$ siten, että se on joukon $\boldsymbol{\Omega}$ sisäpiste. Piste $\boldsymbol{\Psi}^{(k)}$ maksimoi funktion $\mathcal{H}\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}\right)$ joukossa $\boldsymbol{\Omega}$ yhtälön (30) mukaan, joten sen derivaatta häviää kyseissä pisteessä, eli $D^{1,0}\mathcal{H}\left(\boldsymbol{\Psi}^{(k)}; \boldsymbol{\Psi}^{(k)}\right) = 0$. Siispä uskottavuusfunktion derivaatta $DL\left(\boldsymbol{\Psi}^{(k)}\right) = D^{1,0}Q\left(\boldsymbol{\Psi}^{(k)}; \boldsymbol{\Psi}^{(k)}\right)$. Uskottavuusfunktion derivaatta $DL\left(\boldsymbol{\Psi}^{(k)}\right) \neq 0$ missä tahansa kriittisten pisteiden joukkoon kuulumattomassa pisteessä $\boldsymbol{\Psi}^{(k)} \notin \mathcal{S}$. Tästä seuraa, että piste $\boldsymbol{\Psi}^{(k)}$ ei ole funktion $Q\left(\boldsymbol{\Psi}^{(k)}; \boldsymbol{\Psi}^{(k)}\right)$ lokaali maksimi joukossa $\boldsymbol{\Omega}$. M-askeleen määritelmässä olevasta yhtälöstä (24) saadaan, että

$$Q\left(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}\right) > Q\left(\boldsymbol{\Psi}^{(k)}; \boldsymbol{\Psi}^{(k)}\right).$$

Tämä yhdessä yhtälön (30) kanssa osoittaa, että $L\left(\boldsymbol{\Psi}^{(k+1)}\right) > L\left(\boldsymbol{\Psi}^{(k)}\right)$ kaikilla pisteillä $\boldsymbol{\Psi}^{(k)} \notin \mathcal{S}$. Uskottavuusfunktion konvergenssi seuraa tästä. □

Nyt on osoitettu, että uskottavuusfunktion arvo suppenee jossakin kriittisessä pisteessä, mutta kriittisen pisteen laatua (minimi, maksimi vai satulapiste) ei vielä ole osoitettu. Lemmassa 7 esitetään kirjan [5] mukaisesti, että EM-algoritmi voi myös konvergoida uskottavuusfunktion satulapisteeseen eikä täten aina saavuta välttämättä edes lokaalia maksimia.

Lemma 7. Epäyhtälöstä (30) saadaan, että

$$\frac{\partial \mathcal{H}\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}\right)}{\partial \boldsymbol{\Psi}}\Big|_{\boldsymbol{\Psi}=\boldsymbol{\Psi}^{(k)}} = 0, \tag{31}$$

eli että funktion $\mathcal{H}$ derivaatta pisteessä $\boldsymbol{\Psi}^{(k)}$ häviää. Olkoon $\boldsymbol{\Psi}_0$ mielivaltainen parametrivektori $\boldsymbol{\Psi}$. Valitaan $\boldsymbol{\Psi}^{(k)} = \boldsymbol{\Psi}_0$ ja käytetään yhtälöä (31) yhtälön (28) derivaatan lausekkeeseen. Tällöin saadaan, että

$$\frac{\partial \log L\left(\boldsymbol{\Psi}\right)}{\partial \boldsymbol{\Psi}}\Big|_{\boldsymbol{\Psi}=\boldsymbol{\Psi}_0} = \frac{\partial Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}_0\right)}{\partial \boldsymbol{\Psi}}\Big|_{\boldsymbol{\Psi}=\boldsymbol{\Psi}_0}. \tag{32}$$

Uskottavuusfunktion L ja täydellisen datan uskottavuusfunktion ehdollisen odotusarvon Q derivaatat ovat siis yhtäsuuret pisteessä $\boldsymbol{\Psi}_0$. Oletetaan seuraavaksi, että $\boldsymbol{\Psi} = \boldsymbol{\Psi}^*$ ja että $\boldsymbol{\Psi}^*$ on funktion $L\left(\boldsymbol{\Psi}\right)$ kriittinen piste. Tällöin saadaan kriittisen pisteen määritelmästä ja yhtälöstä (32), että

$$\frac{\partial \log L\left(\boldsymbol{\Psi}\right)}{\partial \boldsymbol{\Psi}}\Big|_{\boldsymbol{\Psi}=\boldsymbol{\Psi}^*} = \frac{\partial Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}_0\right)}{\partial \boldsymbol{\Psi}}\Big|_{\boldsymbol{\Psi}=\boldsymbol{\Psi}^*} = 0.$$

Saadaan siis, että EM-algoritmissa maksimoitavan funktion Q kriittiset pisteet ovat myös logaritmisen uskottavuusfunktion kriittisiä pisteitä. Jos piste $\boldsymbol{\Psi}^*$ on funktion $Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}^*\right)$ globaali maksimi joukossa $\boldsymbol{\Omega}$, niin EM-algoritmi voi konvergoida uskottavuusfunktion $L\left(\boldsymbol{\Psi}\right)$ satulapisteeseen, sillä kriittisen pisteen $\boldsymbol{\Psi}^*$ laadusta uskottavuusfunktiolle L ei ole takeita.

Esitetään vielä lauseessa 8, että uskottavuusfunktio konvergoi lokaaliin maksimiin. Konvergenssia globaaliin maksimiin ei pystytä takaamaan EM-algoritmille kuten ei muillekaan optimointialgoritmeille.

Lause 8. Oletetaan, että funktio Q täyttää lemman 5 jatkuvuusehdon ja että millä tahansa $\boldsymbol{\Psi} \in \mathcal{S} \setminus \mathcal{M}$ pätee

$$\sup_{\boldsymbol{\Psi}' \in \boldsymbol{\Omega}} Q\left(\boldsymbol{\Psi}'; \boldsymbol{\Psi}\right) > Q\left(\boldsymbol{\Psi}; \boldsymbol{\Psi}\right).$$

Tällöin kaikki jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteet ovat funktion L lokaaleja maksimeja ja $L\left(\boldsymbol{\Psi}^{(k)}\right)$ konvergoi monotonisesti arvoon $L^* = L\left(\boldsymbol{\Psi}^*\right)$ jollakin lokaalilla maksimilla $\boldsymbol{\Psi}^*$.

3.2.2 Parametrien jono suppenee

Osoitetaan seuraavaksi, että parametrivektoreiden jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ suppenee arvoon $\boldsymbol{\Psi}^*$, mikäli uskottavuusfunktion $L(\boldsymbol{\Psi})$ arvo suppenee arvoon $L^* = L(\boldsymbol{\Psi}^*)$. Merkitään

$$\mathcal{S}(a) = \{\boldsymbol{\Psi} \in \mathcal{S} : L(\boldsymbol{\Psi}) = a\}$$

ja

$$\mathcal{M}(a) = \{\boldsymbol{\Psi} \in \mathcal{M} : L(\boldsymbol{\Psi}) = a\},$$

jotka kuvaavat niitä kriittisten pisteiden joukkoon $\mathcal{S}$ tai lokaalien maksimien joukkoon $\mathcal{M}$ kuuluvia pisteitä $\boldsymbol{\Psi}$, joissa uskottavuusfunktio saa arvon a. Lemman 4 oletuksien nojalla $L(\boldsymbol{\Psi}) \to L^*$ ja kaikki jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteet kuuluvat joukkoon $\mathcal{S}(L^*)$.

Aloitetaan parametrien jonon suppenemisen todistaminen lauseen 9 mukaisella tilanteella, jossa uskottavuusfunktion $L(\boldsymbol{\Psi})$ optimi L^* saavutetaan vain yhdessä kriittisessä pisteessä $\boldsymbol{\Psi}^*$.

Lause 9. Olkoon jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ lemman 4 oletuksien mukaisella GEM-algoritmilla laskettuja parametrivektorin arvoja. Jos $\mathcal{S}(L^*) = \{\boldsymbol{\Psi}^*\}$, missä L^* on lemmassa 4 esiintyvä funktion $L\left(\boldsymbol{\Psi}^{(k)}\right)$ rajapiste, niin silloin $\boldsymbol{\Psi}^{(k)} \to \boldsymbol{\Psi}^*$.

Lauseen 9 ehtoa $\mathcal{S}(L^*) = \{\boldsymbol{\Psi}^*\}$ voidaan relaksoida olettamalla, että $\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\| \to 0$, kun $k \to \infty$. Näin voidaan olettaa, sillä se on välttämätön ehto parametrien jonon suppenemiselle. Lauseessa 10 tätä ehtoa hyödynnetään, kun osoitetaan parametrien jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ suppenevan johonkin kriittisten pisteiden joukon $\mathcal{S}(L^*)$ pisteeseen.

Lause 10. Olkoon jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ lemman 4 oletuksien mukaisella GEM-algoritmilla laskettuja parametrivektorin arvoja. Jos $\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\| \to 0$, kun $k \to \infty$, niin kaikki jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteet kuuluvat yhdistettyyn ja kompaktiin joukkoon $\mathcal{S}(L^*)$, jossa L^* on lemmassa 4 esitetty ääriarvokohta funktiolle $L\left(\boldsymbol{\Psi}^{(k)}\right)$. Erityisesti jos $\mathcal{S}(L^*)$ on diskreetti, niin parametrivektori $\boldsymbol{\Psi}^{(k)}$ konvergoi johonkin pisteeseen $\boldsymbol{\Psi}^* \in \mathcal{S}(L^*)$.

Todistus. Oletuksen **OL2** mukaan parametrien arvojen jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ on rajoitettu. Rajoitetun jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteiden joukko on yhtenäinen ja kompakti, jos $\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\| \to 0$, kun $k \to \infty$. Lemman 4 nojalla jonon $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ rajapisteet kuuluvat joukkoon $\mathcal{S}(L^*)$. Tästä seuraa haluttu tulos. □

Yleistetään seuraavaksi lauseessa 11 lause 10 tilanteeseen, jossa kriittisten pisteiden joukkoon $\mathcal{S}$ kuuluvien parametrivektoreiden sijasta tarkastellaan koko määrittelyjoukkoa $\boldsymbol{\Omega}$.

Lause 11. Olkoon jono $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ lemman 4 oletuksien mukaisella GEM-algoritmilla laskettuja parametrivektorin arvoja. Määritellään, että

$$\mathcal{L}(L) = \{\boldsymbol{\Psi} \in \boldsymbol{\Omega} : L(\boldsymbol{\Psi}) = L\}.$$

Tehdään myös lisäoletus, että $D^{1,0}Q\left(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}\right) = 0$ ja oletetaan sen olevan jatkuva sekä pisteessä $\boldsymbol{\Psi}^{(k+1)}$ että pisteessä $\boldsymbol{\Psi}^{(k)}$. Tällöin $\boldsymbol{\Psi}^{(k)}$ suppenee kriittiseen pisteeseen $\boldsymbol{\Psi}^*$, jossa $L(\boldsymbol{\Psi}^*) = L^*$. Arvo L^* on uskottavuusfunktion $L\left(\boldsymbol{\Psi}^{(k)}\right)$ ääriarvo, jos joko $\mathcal{L}(L^*) = \{\boldsymbol{\Psi}^*\}$ tai $\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\| \to 0$, kun $k \to \infty$, ja $\mathcal{L}(L^*)$ on diskreetti.

Esitellään seuraavaksi unimodaalin funktion määritelmä, jotta osiossa 4 voidaan hyödyntää seurausta 13, joka osoittaa parametrivektorin konvergoivan unimodaalin uskottavuusfunktion globaaliin maksimiin.

Määritelmä 12. (Unimodaali funktio) Uskottavuusfunktio $L(\boldsymbol{\Psi})$ on unimodaalinen maksimoinnin suhteen, jos ehdosta $L(\boldsymbol{\Psi}') > L(\boldsymbol{\Psi})$ seuraa, että $\Delta\boldsymbol{\Psi} = \boldsymbol{\Psi}' - \boldsymbol{\Psi}$ on parantava suunta pisteestä $\boldsymbol{\Psi}$ kaikille pistepareille $\boldsymbol{\Psi}$ ja $\boldsymbol{\Psi}'$. Maksimointitehtävässä suunta $\Delta\boldsymbol{\Psi}$ on parantava, jos $\nabla L(\boldsymbol{\Psi})^T \Delta\boldsymbol{\Psi} > 0$. [12]

Seuraus 13. Oletetaan, että uskottavuusfunktio $L(\boldsymbol{\Psi})$ on unimodaalinen joukossa $\boldsymbol{\Omega}$, $\boldsymbol{\Psi}^*$ on sen ainut kriittinen piste ja $D^{1,0}Q(\boldsymbol{\Psi}'; \boldsymbol{\Psi})$ on jatkuva pisteissä $\boldsymbol{\Psi}'$ ja $\boldsymbol{\Psi}$. Tällöin jokaiselle EM-algoritmilla lasketulle jonolle $\left\{\boldsymbol{\Psi}^{(k)}\right\}$ pätee, että parametrivektori $\boldsymbol{\Psi}^{(k)}$ suppenee uskottavuusfunktion $L(\boldsymbol{\Psi})$ maksimoivaan pisteeseen $\boldsymbol{\Psi}^*$.

3.2.3 Konvergenssiaste

Käydään vielä läpi kirjan [5] mukainen tapa määrittää ja laskea käytännössä EM-algoritmin konvergenssiaste.

Määritelmässä 1 esitelty kuvaus $\boldsymbol{M}$ on kuvaus joukosta $\boldsymbol{\Omega}$ itseensä siten, että

$$\boldsymbol{\Psi}^{(k+1)} = \boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right)$$

kaikilla $k = 0, 1, 2, ...$ Jos parametrivektori $\boldsymbol{\Psi}^{(k)}$ konvergoi johonkin pisteeseen $\boldsymbol{\Psi}^*$ ja kuvaus $\boldsymbol{M}(\boldsymbol{\Psi})$ on jatkuva, niin piste $\boldsymbol{\Psi}^*$ on kuvauksen $\boldsymbol{M}$ kiintopiste eli $\boldsymbol{\Psi}^* = \boldsymbol{M}(\boldsymbol{\Psi}^*)$.

Kuvaukselle $\boldsymbol{\Psi}^{(k+1)} = \boldsymbol{M}\left(\boldsymbol{\Psi}^{(k)}\right)$ pisteessä $\boldsymbol{\Psi}^*$ lasketusta Taylorin sarjasta saadaan pisteen $\boldsymbol{\Psi}^*$ ympäristössä, että

$$\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^* \approx J(\boldsymbol{\Psi}^*)\left(\boldsymbol{\Psi}^{(k)} - \boldsymbol{\Psi}^*\right). \tag{33}$$

Yhtälössä (33) $J(\boldsymbol{\Psi})$ on $d \times d$ Jacobin matriisi kuvaukselle $\boldsymbol{M}(\boldsymbol{\Psi}) = (M_1(\boldsymbol{\Psi}), ..., M_d(\boldsymbol{\Psi}))^T$, jossa alkio

$$J_{ij}(\boldsymbol{\Psi}) = \frac{\partial M_i(\boldsymbol{\Psi})}{\partial \Psi_j}.$$

Jacobin matriisista $J(\boldsymbol{\Psi}^*)$ puhutaan usein konvergenssiasteen matriisina.

Parametrivektorin $\boldsymbol{\Psi}$ todellisen havaitun konvergenssiasteen mitta on globaali konvergenssiaste, joka määritellään raja-arvona

$$r = \lim_{k \to \infty} \frac{\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^*\|}{\|\boldsymbol{\Psi}^{(k)} - \boldsymbol{\Psi}^*\|}, \tag{34}$$

missä $\|\cdot\|$ on mikä tahansa normi d-dimensioisessa euklidisessa avaruudessa $\mathbb{R}^d$. Merkitään, että Jacobin matriisin $J(\boldsymbol{\Psi}^*)$ suurin ominaisarvo on $\lambda_{\max}$. Tiettyjen säännöllisyysehtojen ollessa voimassa, tiedetään konvergenssiasteen olevan $r = \lambda_{\max}$.

Konvergenssiaste voidaan laskea myös komponenteittain. Tällöin komponentin i konvergenssiaste on

$$r_i = \lim_{k \to \infty} \frac{\|\Psi_i^{(k+1)} - \Psi_i^*\|}{\|\Psi_i^{(k)} - \Psi_i^*\|}. \tag{35}$$

Tiettyjen ehtojen ollessa voimassa konvergenssiaste r on komponenttien konvergenssiasteiden r_i maksimi:

$$r = \max_{1 \le i \le d} r_i.$$

Algoritmi tietysti konvergoi jos ja vain jos jokainen komponentti konvergoi. Siispä on luonnollista, että algoritmin konvergenssiaste määräytyy hitaimmin konvergoivan komponentin mukaan, sillä konvergenssia ei saavuteta ennen kuin jokainen komponentti on saavuttanut optimiarvonsa.

Käytännössä yleensä kuitenkin käytetään yhtälöiden (34) ja (35) tilalla muotoja

$$\begin{aligned} r &= \lim_{k \to \infty} \frac{\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\|}{\|\boldsymbol{\Psi}^{(k)} - \boldsymbol{\Psi}^{(k-1)}\|} \quad \text{ja} \\ r_i &= \lim_{k \to \infty} \frac{\|\Psi_i^{(k+1)} - \Psi_i^{(k)}\|}{\|\Psi_i^{(k)} - \Psi_i^{(k-1)}\|}. \end{aligned}$$

4 EM-algoritmin käyttö mixture cure -mallin parametrien laskemiseen

Kuten osiossa 3 todettiin, EM-algoritmi on mixture cure -mallin parametrien laskemiseen sopiva menetelmä, sillä mixture cure -mallissa on latentteja muuttujia. Johdetaan nyt EM-algoritmin käyttämiseen tarvittavat funktiot mixture cure -mallin mukaiselle datalle, implementoidaan saatu algoritmi R-ohjelmointikielellä ja havainnollistetaan sekä dataa että algoritmin konvergenssia kuvilla. Mallin mukaisessa tilanteessa data on sensuroitua, joten vikaantumisaika T tiedetään tarkasti vain osalle yksilöistä. Mallissa on oletuksena myös, että osa yksilöistä on immuuneja tarkastelun kohteena olevalle tapahtumalle, mutta immuunien ja alttiiden yksilöiden osuuksia ei tiedetä.

4.1 Malli

Sovelluksen kohteena olevassa mallissa havaittuina muuttujina ovat aika T, joka on joko vikaantumisaika T^* tai sensurointiaika C yhtälön (6) mukaisesti, ja yhtälön (7) mukaisesti määritetty sensurointi-indeksi δ. Voidaan siis osiossa 3 käytettyjen merkintöjen mukaisesti kirjoittaa, että $\boldsymbol{Y} = (T, \delta)$. Osiossa 2.3 esitelty alttiusindeksi ζ on tässä tapauksessa vaillinaista dataa.

Muuttuja ζ_i kuvaa siis yksilön alttiutta tehdä tarkastelun kohteena oleva tapahtuma ja se on näyte satunnaismuuttujasta $\zeta \sim \text{Bern}(\pi)$. Täydelliselle datalle muuttujan ζ_i arvot tiedetään, mutta tässä tapauksessa se on latentti muuttuja, joten sen arvoja ei tiedetä. Yksilön vikaantumisaika t_i^* on realisaatio satunnaismuuttujasta $T^* \sim \text{Exp}(\lambda)$ ja näitä merkintöjä käyttäen, havaittu aika $t_i = \min(t_i^*, c_i)$ ja sensurointi-indeksi $\delta_i = \mathbb{I}(t_i^* \leq c_i)$, kun yksilön i sensurointiaika c_i on joko erikseen määritetty tai se on näyte jotakin satunnaisjakaumaa noudattavasta satunnaismuuttujasta C. Sensurointiaika on näyte satunnaismuuttujasta C, kun kyseessä on osiossa 2 mainittu satunnainen sensurointi.

Oletetaan, että vikaantumisaika T noudattaa eksponentijakaumaa, joten alttiin populaation välttöfunktio on kaavan (10) mukaan muotoa

$$S(t|\zeta = 1) = P(T > t|\zeta = 1) = e^{-\lambda t}$$

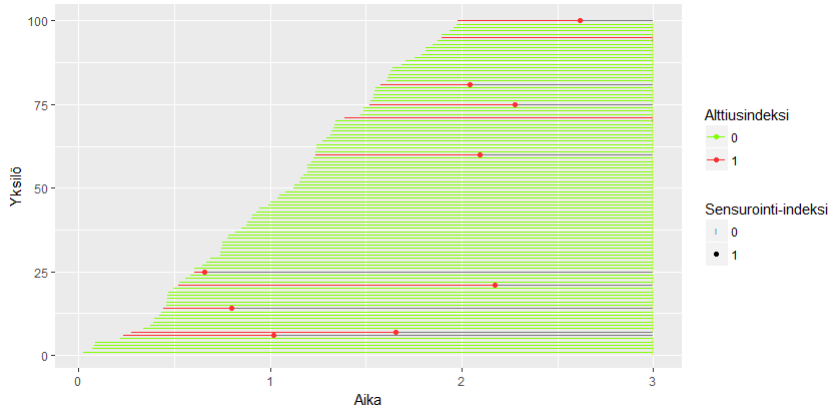
ja tiheysfunktio on kaavan (12) mukaisesti

$$f(t|\zeta = 1) = P(T = t|\zeta = 1) = \lambda e^{-\lambda t}.$$

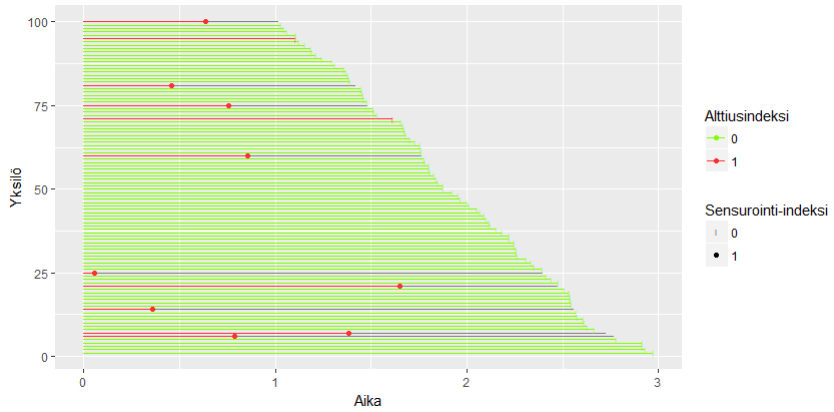
Oletetaan, että alttiiden yksilöiden osuus koko populaatiosta on $P(\zeta = 1) = \pi$. Tällöin immuunien yksilöiden osuus $P(\zeta = 0) = 1 - \pi$, sillä kyseessä on kahden ryhmän sekoitemalli. Saadaan, että mallissa on kaksi

EM-algoritmilla estimoitavaa parametria $\boldsymbol{\theta} = (\pi, \lambda)$, jotka ovat tapahtuman tekevien yksilöiden osuus ja vakiovikaantumistahti.

Kuvassa 9 havainnollistetaan, miltä tällainen data näyttää, kun aloitusajat on piirretty kalenteriajan mukaisesti ja sensurointiaika on kaikille sama tyypin I sensuroinnin mukaisesti. Samaa dataa voidaan havainnollistaa myös kuvan 10 mukaisesti. Siinä havainnollistetaan seuranta-aikaa. Kuvasta 10 on helpompi vertailla yksilöiden vikaantumisaikoja. Data on generoitu siten, että aloittaminen tapahtuu yhden aikayksikön aikana ja maksimiseuranta-aika on kolme aikayksikköä. Generoidussa datassa on 100 yksilöä ja generoinnissa käytetty alttiiden yksilöiden osuus on 10 %.



Kuva 9: Yksilöiden seuranta-ajat kalenterin mukaisesta aloitusajankohdasta tyypin I mukaiseen sensurointiajankohtaan.



Kuva 10: Yksilöiden seuranta-aikojen pituudet tyypin I sensuroinnin tapauksessa.

Mixture cure -mallin uskottavuusfunktio ja muut EM-algoritmin käyttöä varten tarvittavat funktiot on johdettu artikkelissa [13] tilanteelle, jossa vikaantumisaika T noudattaa Weibull-jakaumaa. Tämän tutkielman sovelluksen tilanne on Weibull-jakauman erikoistapaus, joten esitetään nyt juuri tämän tapauksen funktiot, jotka on itse johdettu.

Taulukossa 2 esitetään todennäköisyydet, että yksilö vikaantuu tai selviää ajanhetkellä t ehdolla, että se on immuuni tai altis. Näiden ehdollisten todennäköisyyksien avulla saadaan marginaalinen välttöfunktio (22) kirjoitettua eksplisiittisesti. Se on tällöin muotoa

$$\begin{aligned} S(t) = P(T > t) &= P(T > t|\zeta = 0)\, P(\zeta = 0) + P(T > t|\zeta = 1)\, P(\zeta = 1) \\ &= 1 \cdot (1 - \pi) + e^{-\lambda t} \cdot \pi \\ &= 1 - \pi + \pi e^{-\lambda t}. \end{aligned} \tag{36}$$

Vastaavalla tavalla saadaan laskettua marginaalinen tiheysfunktio

$$\begin{aligned} f(t) = P(T = t) &= P(T = t|\zeta = 0)\, P(\zeta = 0) + P(T = t|\zeta = 1)\, P(\zeta = 1) \\ &= 0 \cdot (1 - \pi) + \lambda e^{-\lambda t} \cdot \pi \\ &= \pi \lambda e^{-\lambda t}. \end{aligned}$$

Taulukko 2: Välttö- ja tiheysfunktioiden laskemiseen tarvittavat ehdolliset todennäköisyydet.

$P(T\|\zeta)$	$T = t$	$T > t$
$\zeta = 0$	0	1
$\zeta = 1$	$\lambda e^{-\lambda t}$	$e^{-\lambda t}$

Epätäydellisen data uskottavuusfunktio on kaavan (8) mukaisesti muotoa

$$L(\boldsymbol{\theta}|\ \boldsymbol{t}, \boldsymbol{\delta}) = \prod_{i=1}^{n} \left(\pi \lambda e^{-\lambda t_i}\right)^{\delta_i} \left(1 - \pi + \pi e^{-\lambda t_i}\right)^{1-\delta_i} \tag{37}$$

ja sen logaritminen uskottavuusfunktio joko ottamalla logaritmi yhtälöstä (37) tai kaavan (9) perusteella on

$$l(\boldsymbol{\theta}|\boldsymbol{t}, \boldsymbol{\delta}) = \sum_{i=1}^{n} \left\{ \delta_i \left(\log \pi + \log \lambda - \lambda t_i\right) + (1 - \delta_i) \log\left(1 - \pi + \pi e^{-\lambda t_i}\right) \right\}.$$

EM-algoritmia käytettäessä tarvitaan kuitenkin epätäydellisen datan uskottavuusfunktion sijasta täydellisen datan uskottavuusfunktio. Se on mixture cure -mallin tapauksessa muotoa

$$L_c\left(\boldsymbol{\theta}|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = \prod_{i=1}^{n}\left(\pi\lambda e^{-\lambda t_i}\right)^{\delta_i}\left[(1-\pi)^{\mathbb{I}(\zeta_i=0)}\left(\pi e^{-\lambda t_i}\right)^{\mathbb{I}(\zeta_i=1)}\right]^{1-\delta_i}. \tag{38}$$

Otetaan yhtälöstä (38) logaritmi, jolloin saadaan täydellisen datan logaritminen uskottavuusfunktio lopulta muotoon

$$\begin{aligned} &l_c\left(\boldsymbol{\theta}|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = \\ &\sum_{i=1}^{n}\{\delta_i\left[\log\pi + \log\lambda - \lambda t_i\right] + \\ &(1-\delta_i)\left[\mathbb{I}\left(\zeta_i=0\right)\log\left(1-\pi\right) + \mathbb{I}\left(\zeta_i=1\right)\left(\log\pi - \lambda t_i\right)\right]\}. \end{aligned} \tag{39}$$

EM-algoritmin E-askeleessa olevan funktion (23) muodostamisessa tarvitaan taulukossa 3 esitetyt ehdolliset todennäköisyydet. Taulukon 3 arvot on laskettu seuraavilla kaavoilla

$$P\left(\zeta_i=j|T_i>t_i,\boldsymbol{\theta}^{(k-1)}\right) = \frac{P\left(T_i>t_i|\zeta_i=j,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=j)}{P\left(T_i>t_i|\zeta_i=0,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=0)+P\left(T_i>t_i|\zeta_i=1,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=1)} \text{ ja} \tag{40}$$

$$P\left(\zeta_i=j|T_i=t_i,\boldsymbol{\theta}^{(k-1)}\right) = \frac{P\left(T_i=t_i|\zeta_i=j,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=j)}{P\left(T_i=t_i|\zeta_i=0,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=0)+P\left(T_i=t_i|\zeta_i=1,\boldsymbol{\theta}^{(k-1)}\right)P(\zeta_i=1)}, \tag{41}$$

joissa $j \in \{0,1\}$. Kaavoilla (40) ja (41) lasketaan ehdolliset todennäköisyydet, että yksilö on immuuni tai altis, jos se on selvinnyt tai vikaantunut ajanhetkellä t.

Taulukko 3: Funktion Q muodostamiseen tarvittavat ehdolliset todennäköisyydet.

$P\left(\zeta\|T\right)$	$\zeta=0$	$\zeta=1$
$T>t$	$\frac{1-\pi}{1-\pi+\pi e^{-\lambda t}}$	$\frac{\pi e^{-\lambda t}}{1-\pi+\pi e^{-\lambda t}}$
$T=t$	0	1

Nyt yhtälöt (39), (40) ja (41) yhdistämällä saadaan EM-algoritmilla mak-

simoitava funktio (23) muotoon

$$
\begin{aligned}
&Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k-1)}\right) = E_{\zeta|T,\boldsymbol{\theta}^{(k-1)}}\left[l\left(\boldsymbol{\theta}|t,\delta,\zeta\right)\right] \\
&= \sum_{i=1}^{n}\sum_{j=0}^{1} P\left(\zeta_i = j|T_i > t_i, \boldsymbol{\theta}^{(k-1)}\right)^{1-\delta_i} P\left(\zeta_i = j|T_i = t_i, \boldsymbol{\theta}^{(k-1)}\right)^{\delta_i} l_c\left(\boldsymbol{\theta}|t_i,\delta_i,\zeta_i\right) \\
&= \sum_{i:\delta_i=0}\left[\frac{1-\pi^{(k-1)}}{1-\pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}\log\left(1-\pi\right) + \frac{\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}{1-pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}\left(\log\pi - \lambda t_i\right)\right] \\
&+ \sum_{i:\delta_i=1}\left[\log\pi + \log\lambda - \lambda t_i\right],
\end{aligned}
\tag{42}
$$

kun viimeisintä muotoa edeltävät välivaiheet on jätetty kirjoittamatta. Derivoimalla yhtälö (42) erikseen kummankin parametrin, π ja λ, suhteen ja ratkaisemalla saatujen derivaattafunktioiden nollakohdat, saadaan parametreille laskettua estimaatit

$$
\pi^{(k)} = \frac{1}{n}\left[\sum_{i:\delta_i=0}\frac{\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}{1-\pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}} + \sum_{i:\delta_i=1} 1\right] \quad \text{ja} \tag{43}
$$

$$
\lambda^{(k)} = \frac{\sum_{i:\delta_i=1} 1}{\sum_{i:\delta_i=0}\frac{\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}{1-\pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}} + \sum_{i:\delta_i=1} t_i}. \tag{44}
$$

Osiossa 3 osoitettiin lemmassa 7, että EM-algoritmilla lasketut funktion Q maksimoivat parametrien arvot ovat myös täydellisen datan uskottavuusfunktion kriittisiä pisteitä, joten kaavoilla (43) ja (44) lasketut arvot maksimoivat myös uskottavuusfunktion (39).

Täydellisen datan logaritminen uskottavuusfunktio (39) saadaan jaettua kahteen osaan siten, että toinen osa sisältää kaikki parametria π sisältävät termit ja toinen sisältää kaikki parametria λ sisältävät termit. Voidaan siis merkitä, että

$$
l_c\left(\pi,\lambda|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = l_\pi\left(\pi|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) + l_\lambda\left(\lambda|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right). \tag{45}
$$

Yhtälössä (45) oleva funktio l_π on muotoa

$$
\begin{aligned}
&l_\pi\left(\pi|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = \\
&\sum_{i=1}^{n}\left\{\delta_i\log\pi + \left(1-\delta_i\right)\left[\mathbb{I}\left(\zeta_i=0\right)\log\left(1-\pi\right) + \mathbb{I}\left(\zeta_i=1\right)\log\pi\right]\right\}
\end{aligned}
\tag{46}
$$

ja funktio l_λ on muotoa

$$l_\lambda\left(\lambda|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right)=\sum_{i=1}^{n}\left\{\delta_i\left[\log\lambda-\lambda t_i\right]-\left(1-\delta_i\right)\mathbb{I}\left(\zeta_i=1\right)\lambda t_i\right\}. \tag{47}$$

Sekä funktio (46) että funktio (47) koostuu konkaavien termien summasta, sillä logaritmifunktio on konkaavi samoin kuin lineaariset termit ovat. Konkaavien funktioiden summana myös funktio (45) on konkaavi. Toisaalta, konkaavit funktion ovat unimodaalisia maksimoinnin suhteen [12], joten seurauksen 13 mukaan tiedetään siis, että EM-algoritmi konvergoi täydellisen datan logaritmisen uskottavuusfunktion globaaliin maksimiin.

Täydelliselle datalle pystytään suoraan laskemaan paramertien π ja λ optimaaliset arvot. Derivoidaan yhtälö (46) muuttujan π suhteen ja yhtälö (47) muuttujan λ suhteen. Täydellisen datan logaritmiselle uskottavuusfunkiolle optimaaliset arvot saadaan selvittämällä laskettujen derivaattojen nollakohdat. Datan ollessa täydellistä, tiedetään, mitkä yksilöistä ovat immuuneja ja mitkä alttiita. Siitä syystä parametrien π ja λ optimiarvot saadaan ratkaistua sijoittamalla havaitut muuttujien t, δ ja ζ arvot yhtälöihin

$$\pi = \frac{1}{n}\sum_{i=1}^{n}\left[\delta_i+\left(1-\delta_i\right)\mathbb{I}\left(\zeta_i=1\right)\right]=\frac{1}{n}\sum_{i=1}^{n}\zeta_i \quad \text{ja} \tag{48}$$

$$\lambda = \frac{\sum_{i=1}^{n}\delta_i}{\sum_{i=1}^{n}t_i\left[\delta_i+\left(1-\delta_i\right)\mathbb{I}\left(\zeta_i=1\right)\right]}=\frac{\sum_{i=1}^{n}\delta_i}{\sum_{i=1}^{n}t_i\zeta_i}. \tag{49}$$

Yhtälöitä (48) ja (49) ei siis voida käyttää puuttuvalle datalle parametrien π ja λ laskemiseen, sillä puuttuvalle datalle ei tiedetä muuttujan ζ arvoa. Tästä syystä käytetään EM-algoritmia parametrien optimaalisten arvojen selvittämiseen.

4.2 Toteutus R-ohjelmointikielellä

Funktioiden johtamisen jälkeen toteutettiin saman asian R-ohjelmointikielellä RStudio-ohjelmointiympäristöä käyttämällä. Luetteloinnissa 1 esitetään funktio `cureEM`, jolla saadaan laskettua mixture cure -mallin parametrien optimiarvot EM-algoritmilla. Funktio `cureEM` ottaa kaksi parametria: ensimmäinen näistä, *params*, on parametrien π ja λ alkuarvot sisältävä vektori ja toinen, D, on havaittu data. Havaittu data koostuu muuttujista *time*, *censored* ja *weight*. Muuttuja *time* on havaittu aika t_i ja muuttuja *censored* on sensurointi-indeksi δ_i. Muuttuja *weight* kuvaa todennäköisyyttä, että yksilö on altis tarkastelun kohteena olevalle tapahtumalle, ja se lasketaan jokaisella iteraatiokierroksella uudelleen yhtälöitä (40) ja (41) käyttäen arvolla $j=1$.

```r
#EM-algoritmi mixture cure -mallille

cureEM <- function(params, D){

  lambda_prev <- params[1]
  pi_prev <- params[2]
  n <- length(D$time)
  time_censored <- D$time[D$censored==0]
  time_not_censored <- D$time[D$censored==1]

  #Lopetuskriteeri
  eps <- 10^(-5)
  continue <- TRUE

  while(continue){

    #Kertoimien laskeminen
    D$weight[D$censored == 0] <- pi_prev*exp(-lambda_prev*time_censored) /
      (1-pi_prev + pi_prev*exp(-lambda_prev*time_censored))
    D$weight[D$censored == 1] <- 1

    #Parametrien maksimointi
    pi <- sum(D$weight)/n

    lambda <- sum(D$weight[D$censored == 1])/
      (sum(D$weight[D$censored == 0]*time_censored) +
         sum(D$weight[D$censored == 1]*time_not_censored))

    #Lopetuskriteerin toteutumisen tarkistus ja parametrien arvojen paivitys
    if(sqrt((pi_prev - pi)^2 + (lambda_prev - lambda)^2) < eps){
      continue <- FALSE
    }else{
      lambda_prev <- lambda
      pi_prev <- pi
    }

  }

  result <- c(lambda, pi)
  return(result)
}
```

Luettelointi 1: Mixture cure -mallin parametrien laskeminen EM-algoritmilla.

Funktion `cureEM` toiminnan oikeellisuus varmistettiin generoimalla testidataa luetteloinnin 2 mukaisella tavalla. Data on generoitu parametrien arvoilla $\pi = 0.3$ ja $\lambda = 1$, joten tiedetään funktion `cureEM` toimivan, kun sen laskemat arvot vastasivat datan generoinnissa käytettyjä arvoja.

```
#Luodaan testidataa
library(LaplacesDemon)
tmax <- 2
n <- 1000
cure_fraction <- 0.7
cured <- rbern(n, cure_fraction)
testidata <- data.frame(time = rexp(n)*(1-cured) + tmax*cured,
                        cured = 1-cured)
testidata$censored <- as.integer(testidata$time < tmax)
testidata$time[which(testidata$censored == FALSE)] <- tmax
testidata$weight <- 1
```

Luettelointi 2: Testidatan generoiminen.

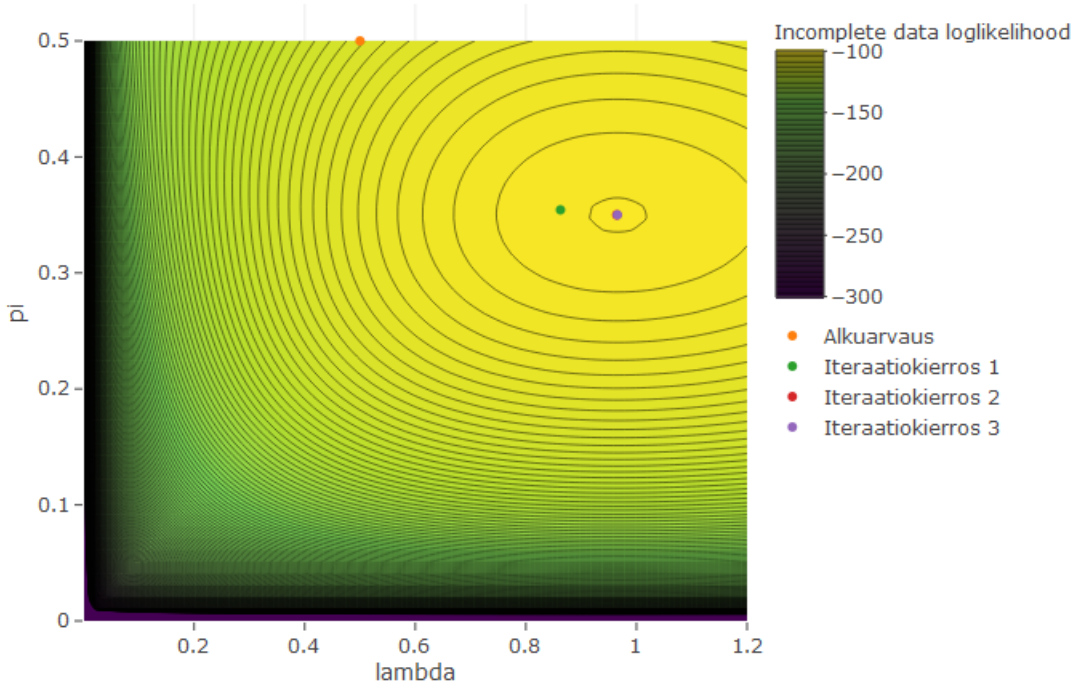
Tämän lisäksi laskettiin R:n valmiilla optimointifunktiolla `optim` logaritmisen uskottavuusfunktion (39) maksimoivat parametrien arvot luetteloinnissa 3 esitetyllä koodilla. Tällä tavoin varmistettiin, että funktio `cureEM` tuottaa saman ratkaisun kuin jokin muukin optimointimenetelmä. Mixture cure -mallille halutaan löytää uskottavuusfunktion maksimi ja funktio `optim` ratkaisee annetun funktion minimin, joten täytyy muistaa, että

$$\arg\max f(x) = \arg\min -f(x).$$

```
#Logaritminen uskottavuusfunktio
loglikelihood <- function(params){

  lambda <- params[1]
  pi <- params[2]
  delta <- testidata$censored
  t <- testidata$time
  cured <- testidata$cured

  -sum(delta*(log(pi) + log(lambda) - lambda*t) +
           (1-delta)*((cured == 0)*log(1-pi) + (cured == 1)*(log(pi) -
               lambda*t)))
}

#Optimointi funktiolla optim()
#Alkuarvaus parametreille lambda ja pi
init <- c(0.5, 0.5)
optim(init, fn = loglikelihood)
```

Luettelointi 3: Täydellisen datan logaritminen uskottavuusfunktio ja valmiilla optimointifunktiolla optimaalisten parametrien ratkaiseminen

Havainnollistetaan algoritmin konvergenssia kuvalla 11. Siitä nähdään, että epätäydellisen datan logaritmisen uskottavuusfunktion suurin arvo saavutetaan suunnilleen arvoilla $\lambda = 1$ ja $\pi = 0.35$. Lisäksi nähdään, että tällä datalla luetteloinnin 1 funktio konvergoi jo muutaman iteraatiokierroksen jälkeen uskottavuusfunktion optimiin.



Kuva 11: Generoidun datan konvergenssi epätäydellisen datan logaritmisen uskottavuusfunktion maksimiin.

5 Pelidataan soveltaminen

Käytetään osiossa 2 esiteltyä teoriaa peleistä kerättyyn dataan. Monissa peleissä on mahdollisuus ostotapahtumien tekemiseen peliä pelatessa ja niiden tekeminen on pelin tekijän kannalta jopa toivottavaa. Kaikki pelaajat eivät kuitenkaan ostotapahtumaa tee ja osa pelaajista tekee useita ostoja. Pelaajat voidaan siis jaotella kahteen osapopulaatioon, jotka osion 2 käsitteiden mukaisesti ovat alttiiden pelaajien joukko ja immuunien pelaajien joukko. Ensin mainittuun siis kuuluvat ne, jotka tekevät ostotapahtuman jossain vaiheessa, ja jälkimmäiseen kuuluvat ne, jotka eivät ostotapahtumaa tee, vaikka niitä seurattaisiin äärettömän pitkään. Tilanne on siis osiossa 4.1 esitellyn mallin mukainen. Ostotapahtuman tekevillä pelaajilla alttiusindeksi $\zeta = 1$ ja ostotapahtuman tekemiselle immuuneille pelaajille $\zeta = 0$.

Pelistä on kerätty dataa tietyllä aikavälillä, joten dataa on sensuroitu tyypin I sensuroinnin mukaisesti. Kaikilla pelaajilla on siis kalenterin mukaisessa ajassa sama sensurointiajankohta – havainnoinnin päättymisen ajankohta. Aiemmin esitelty sensurointi-indeksi δ toimii tässäkin tapauksessa samalla tavalla: $\delta = 0$, jos pelaaja on sensuroitu, ja $\delta = 1$, jos pelaaja on tehnyt ostotapahtuman.

Joissain peleissä pelaajat aloittavat hyvin lyhyen aikavälin aikana, joten seuranta-aika jokaiselle pelaajalle on likimain sama. Toisissa peleissä aloitukset tapahtuvat pidemmän aikavälin aikana. Aloitusaikojen jakautumiseen vaikuttaa se, että peliä testatessa sitä mainostetaan. Pelaajia saattaa silloin tulla hyvinkin lyhyen ajan sisällä tai sitten niitä tulee koko mainoskampanjan ajan tasaisesti. Aikaisemmin julkaistuille peleille uusia pelaajia tulee yleensä melko tasaista tahtia. Pelidatan tapauksessa havaittu aika t on aika aloituksesta joko ostotapahtuman tekemiseen tai datan keräämisen päättymiseen.

Osion 4.1 mallissa käytetyt parametrit, λ ja π, ovat tässä tapauksessa konversiotahti ja monetisaatioprosentti, ja ne lasketaan osion 4 mukaisella tavalla. Monetisaatiolla tarkoitetaan ostotapahtuman tekemistä. Pelaaja siis monetisoituu tehdessään ensimmäisen ostotapahtuman. Konversiotahdilla tarkoitetaan monetisoitumisnopeutta. Vikaantumisaikojen oletetaan noudattavan eksponenttijakaumaa, joten konversiotahti λ on vakio. Yksittäiselle pelille malli siis ennustaa, kuinka iso osa pelaajista tekee ostotapahtuman ja kuinka nopeasti ensimmäinen ostotapahtuma tehdään. Tässä tutkielmassa tavoitteena on tutkia, missä vaiheessa ja kuinka tarkasti malli pystyy lopullisen monetisaatioprosentin ennustamaan. Osan testeistä tein Hipster sheep -nimisestä pelistä [1] kerätylle datalle ja osan tein mallin mukaiselle generoidulle datalle.

5.1 Käytetyn datan kuvailu ja muokkaus

Käyttämäni data on kerätty Hipster sheep -nimisestä pelistä 11.12.2014 ja 4.1.2017 välisenä aikana. Data koostuu kolmesta taulukosta, joiden nimet ovat *sheepsters_device, sheepsters_end_session* ja *sheepsters_store*.

Taulukossa *sheepsters_device* on seitsemän muuttujaa – *id, nr, version, timestamp, platform, country* ja *language* – ja 20 295 havaintoa. Taulukossa 4 näytetään viisi ensimmäistä riviä tästä datasta. Tämä data kertoo pelaajan ensimmäisen pelin alkamisajankohdan sekä mitä versiota, millä alustalla (android tai ios), missä maassa ja millä kielellä pelaaja on peliä pelannut. Taulukko koostuu siis ensimmäisellä pelikerralla kerättävistä tiedoista.

Taulukko 4: Sheepsters_device-datan ensimmäiset viisi riviä.

id	nr	version	timestamp	platform	country	langugae
gp_0010e805-3a50-47ef-a932-4752945eceec	1	1.18	2015-10-23 10:19:22	android	GB	en
gp_0014cc9a-2b20-49ba-a1a1-48d0679dfee2	1	1.18	2015-10-24 09:32:23	android	GB	en
gp_001598a0-4988-4431-a465-b42af85bec93	1	1.21	2015-12-14 14:42:33	android	CN	zh
gp_0029862d-3236-4b5d-bf19-be399b98d4fb	1	1.22b	2015-12-09 10:14:41	android	US	en
gp_0031cfeb-ee08-479e-a07e-382c94c0b764	1	1.18	2015-10-23 09:15:14	android	US	en

Taulukossa *sheepsters_end_session* on kuusi saraketta – *id, nr, version, timestamp, number* ja *duration* – ja 299 408 riviä. Tällä kertaa muuttuja *timestamp* kuvaa pelin lopettamisajankohtaa. Muuttuja *duration* nimensä mukaisesti kuvaa pelin kestoa. Taulukossa 5 esitellään tämän datan viisi ensimmäistä riviä.

Taulukko 5: Sheepsters_end_session-datan ensimmäiset viisi riviä.

id	nr	version	timestamp	number	duration
gp_0029862d-3236-4b5d-bf19-be399b98d4fb	6	1.22b	2015-12-09 10:14:50	1	9
gp_00d19508-90fe-431e-a117-c69fc382748d	6	1.32	2016-07-23 06:32:43	1	0
gp_010702a0-ca52-4f0c-8443-f8ffba4fbdbf	6	1.35	2016-09-23 12:58:15	1	2
gp_013f2ec1-33d6-4e1d-b51c-17ad208db600	6	1.31	2016-07-12 02:49:49	1	0
gp_0161e2d0-4804-4363-bf5c-8bdaf08f6dec	6	1.33	2016-10-22 13:58:00	1	2

Taulukossa *sheepsters_store* puolestaan on ostotapahtuman tehneet pelaajat, ostotapahtuman ajankohta ja ostetun tavaran nimi. Muuttujia on siis viisi: *id, nr, version, timestamp* ja *purchase*. Havaintoja tässä taulukossa on yhteensä vain 1 018 ja niistä viisi ensimmäistä esitetään taulukossa 6.

Dataa pitää muokata ennen kuin se pystytään syöttämään listauksessa 1 esitetylle funktiolle ja pystytään laskemaan arvot osiossa 4.1 esitellyn mallin parametreille λ ja π. Ensin poistetaan yksilöt, joilla jokin havaituista ajoista on havainnointivälin ulkopuolella. Tämän jälkeen otetaan *sheepsters_end_session* taulukosta vain kunkin pelaajan viimeinen havaittu lopetusajankohta. Vastaavasti ostotapahtumien taulukosta otetaan vain

Taulukko 6: Sheepsters_store-datan viisi ensimmäistä riviä.

id	nr	version	timestamp	purhcase
gp_ef98c216a189eb83	6	1.9	2015-05-07 09:24:08	jarful_of_diamonds
macosx_eaed619e-f0e7-468b-9993-8ad944fd0ecc	12	1.22b	2015-12-07 14:15:38	jarful_of_diamonds
macosx_eaed619e-f0e7-468b-9993-8ad944fd0ecc	12	1.22b	2015-12-07 14:22:37	jarful_of_diamonds
macosx_855cb5e4-01ca-4081-91f8-30cfad0b8e9d	14	1.35b	2016-08-29 10:55:20	backbag_of_diamonds
gp_3bfb710c2bf9bd5a	15	1.13b	2015-07-13 07:51:24	jarful_of_diamonds

ensimmäisten ostotapahtumien ajankohdat. Uniikit pelaajat on tunnistettu muuttujien *id* ja *version* parien avulla. Osa pelaajista on pelannut useaa eri versiota, joten tästä syystä pelkästään muuttujan *id* käyttäminen tuottaisi ongelmia.

Aloitus- ja lopetusaikojen taulukot yhdistetään siten, että otetaan vain ne muuttujien *id* ja *version* yhdistelmät, jotka esiintyvät kummassakin taulukossa. Saatuun taulukkoon lisätään vielä tieto ostotapahtuman ajankohdasta niille pelaajille, jotka sellaisen tekivät, ja lopuilla pelaajilla muuttujan *timestamp_store* arvo on puuttuva. Tämän jälkeen poistetaan datasta vielä ne pelaajat, joilla aloitus- ja lopetusajankohdat eivät ole oikeassa järjestyksessä tai ostotapahtuman ajankohta on ennen aloittamista tai vasta lopettamisen jälkeen. Oikea järjestys on siis $timestamp\_start < timestamp\_store < timestamp\_end$, kun ajankohtien vertailuun käytetään kirjastossa `anytime` olevaa funktiota `anytime`, joka muuttaa ajankohdat sekuntimääräksi vuoden 1970 alusta laskettuna.

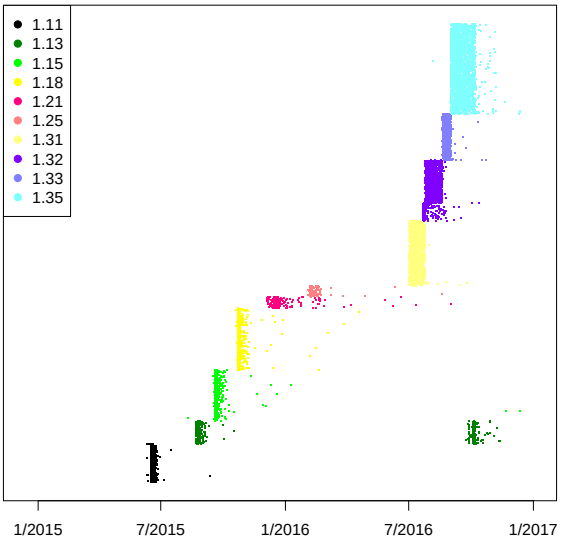
Pelaajat ovat pelanneet yhteensä 68:aa eri versiota. Data rajataan siten, että tarkastellaan vain pelaajia, jotka pelasivat jotakin kymmenestä suosituimmasta versiosta. Taulukossa 7 esitetään näiden versioiden pelaajamäärät ja ostotapahtuman tehneiden pelaajien lukumäärät. Huomataan, että kovinkaan moni pelaaja ei ole tehnyt ostotapahtumaa.

Taulukko 7: Suosituimpien versioiden pelaajien ja ostotapahtumien määrät.

Versio	1.11	1.13	1.15	1.18	1.21	1.25	1.31	1.32	1.33	1.35
Pelaajia	1006	601	1338	1604	309	287	1691	1582	1211	2366
Ostajia	1	1	2	6	6	5	24	21	18	35

Myös pelaajien aloitustahdeissa on eroja, kuten nähdään kuvasta 12. Versioissa 1.31, 1.32, 1.33 ja 1.35 aloitusajat ovat jakautuneet laajemmalle aikavälille kuin muissa versioissa. Versiossa 1.13 puolestaan on mielenkiintoista, että moni on alkanut pelaamaan sitä reilu vuotta myöhemmin kuin ensimmäiset sitä versiota pelanneet. Tämän tutkielman myöhemmissä vaiheissa keskitytään vain versioon 1.18 ja sitä myöhempiin versioihin, sillä käytettä-

vä malli olettaa, että ostotapahtumia on tehty, ja versiota 1.18 aiemmissa versioissa niitä ei juurikaan ole.



Kuva 12: Suosituimpia versioita pelaajaavien pelaajien aloitusajat.

Osiossa 4.1 kuvaillussa tilanteessa havaittu data $\boldsymbol{y} = (t, \delta)$, joten selvitetään havaitut ajat t_i ja sensurointi-indeksit δ_i Hipster sheep -pelin datalle. Pelaajan havaittu aika on ensimmäisen pelin aloitusajankohdasta joko ostotapahtuman ajankohtaan tai sensurointiajankohtaan mennessä kulunut aika. Havaitun ajan laskemiseen käytetään R:n valmista funktiota `difftime` ja yksikön määritin olevan päiviä. Funktio `difftime` laskee kahden merkkijonona annetun ajankohdan välisen ajan. Sensurointi-indeksit ovat kuten aiemminkin ja ne määritetään havaitun ajan laskemisen yhteydessä.

5.2 Mallin oletuksien toteutuminen

Nyt käytettävä data on saatu mallin vaatimaan muotoon, joten tarkastellaan seuraavaksi osiossa 4.1 esitetyn mallin oletuksien toteutumista. Ensimmäinen oletus on, että osa pelaajista on immuuneja ja osa alttiita tarkastelun kohteena olevalle tapahtumalle, eli tässä tapauksessa ostotapahtuman tekemiselle. Toinen oletus on, että havaittujen ostotapahtumien ajankohdat noudattavat eksponenttijakaumaa.

Käytetään artikkelissa [14] esiteltyä Akaike-informaatiokriteeriä immuunien yksilöiden olemassaolon selvittämiseen. Sen avulla voidaan selvittää dataa parhaiten kuvaava malli useiden vaihtoehtojen joukosta. Lasketaan vaihtoehtoina oleville malleille AIC-arvot kaavalla

$$\mathrm{AIC} = 2k - 2\log\left(\hat{L}\right),$$

jossa vakio k on mallin parametrien lukumäärä ja muuttuja $\hat{L}$ on suurimman uskottavuuden estimaatti, ja valitaan malleista se, jolla saadaan pienin AIC-arvo.

Tässä tapauksessa vertailtavia malleja on kaksi. Ensimmäisessä näistä oletetaan, että $\pi = 1$, mikä tarkoittaa, että ostotapahtumalle immuuneja pelaajia ei ole eli kaikki pelaajat tekevät ostotapahtuman jossain vaiheessa. Tässä tapauksessa mixture cure -mallin käyttäminen ei olisi järkevää. Toinen vertailtava malli on mixture cure -malli, jolloin datan tulee olla sen oletuksien mukaista eli osan pelaajista tulee olla immuuneja monetisoitumiselle. Tällöin tapahtuman tekevien pelaajien osuus $\pi < 1$.

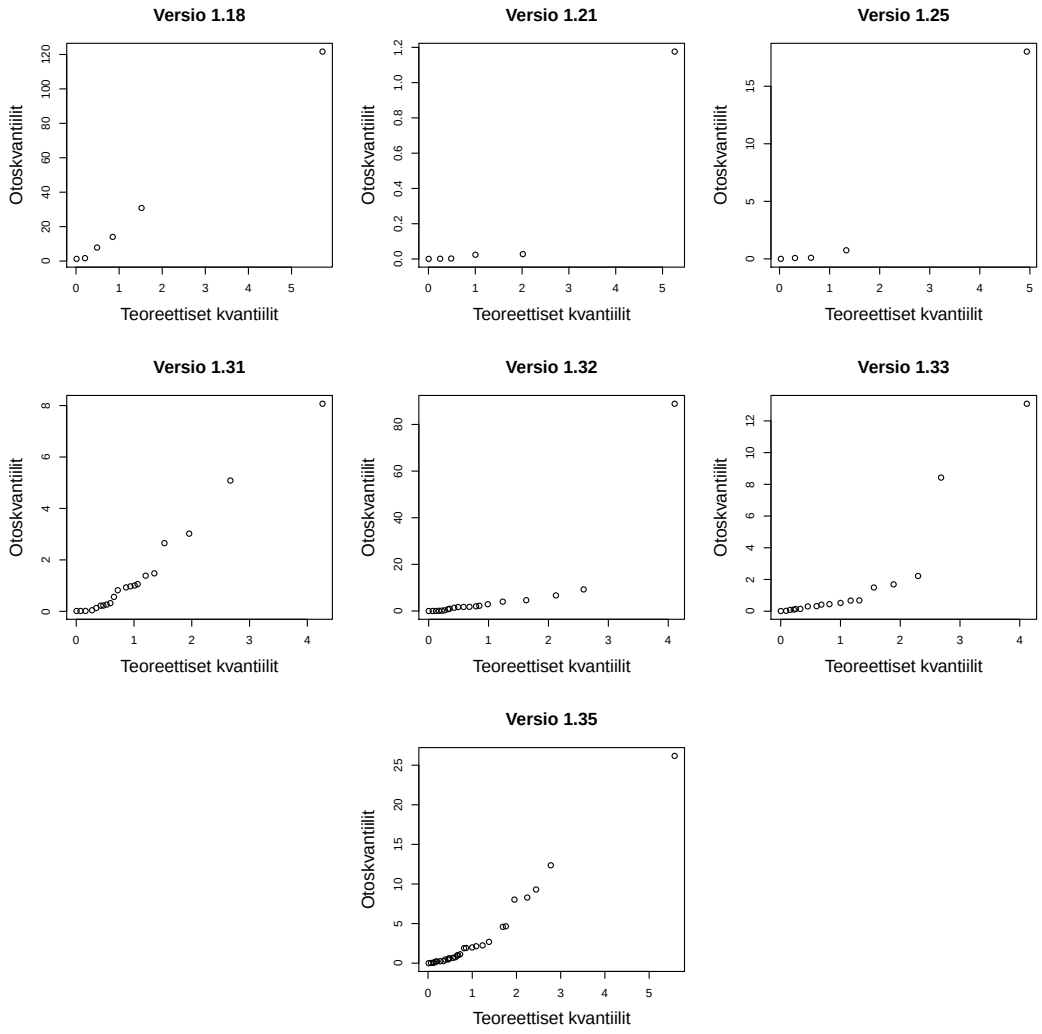
Dataan sovitettava funktio on nyt epätäydellisen datan uskottavuusfunktio (37), sillä täydellisen datan uskottavuusfunktion käyttämiseen tarvittaisiin tieto pelaajien alttiusindeksin ζ arvoista. Lasketaan AIC-arvot jokaiselle tarkasteltelun kohteena olevalle versiolle erikseen, jotta tiedetään jokaisen osadatan noudattavan haluttua mallia. Taulukossa 8 esitetään lasketut arvot ja niistä nähdään, että jokaiselle versiolle pienemmän AIC-arvon tuottaa immuuneja pelaajia sisältävä malli; tämän kriteerin pohjalta mixture cure -mallin käyttäminen on siis järkevää.

Taulukko 8: Tarkasteltaville versioille lasketut AIC-arvot.

Versio	$\pi < 1$	$\pi = 1$
1.18	135.6777	154.0082
1.21	114.0396	132.1508
1.25	97.14472	110.3026
1.31	395.2878	444.7831
1.32	383.3518	417.9001
1.33	337.9899	364.7231
1.35	565.7867	604.2286

Tarkastellaan vielä mallin toista oletusta: havaittujen vikaantumisaikojen pitäisi noudattaa eksponenttijakaumaa. Verrataan kunkin version vikaantu-

misaikojen jakaumaa eksponenttijakauman mukaiseen dataan kuvan 13 kvantiilikuvaajien avulla. Kuvaajissa pitäisi näkyä pisteiden muodostama suora, jotta otos noudattaisi eksponenttijakaumaa. Kullakin versiolla on suoralta poikkeavia arvoja enintään kaksi, joten voidaan olettaa, että mallin toinenkin oletus toteutuu jokaiselle tarkasteltavalle versiolle. Eksponenttijakauma vaikuttaakin olevan vähintäänkin hyvä arvio vikaantumisaikojen jakaumasta, vaikka joillekin pelaajille konversio kestää kauan.



Kuva 13: Vikaantumisaikojen kvantiilikuvaajat.

5.3 Mallin käyttäminen

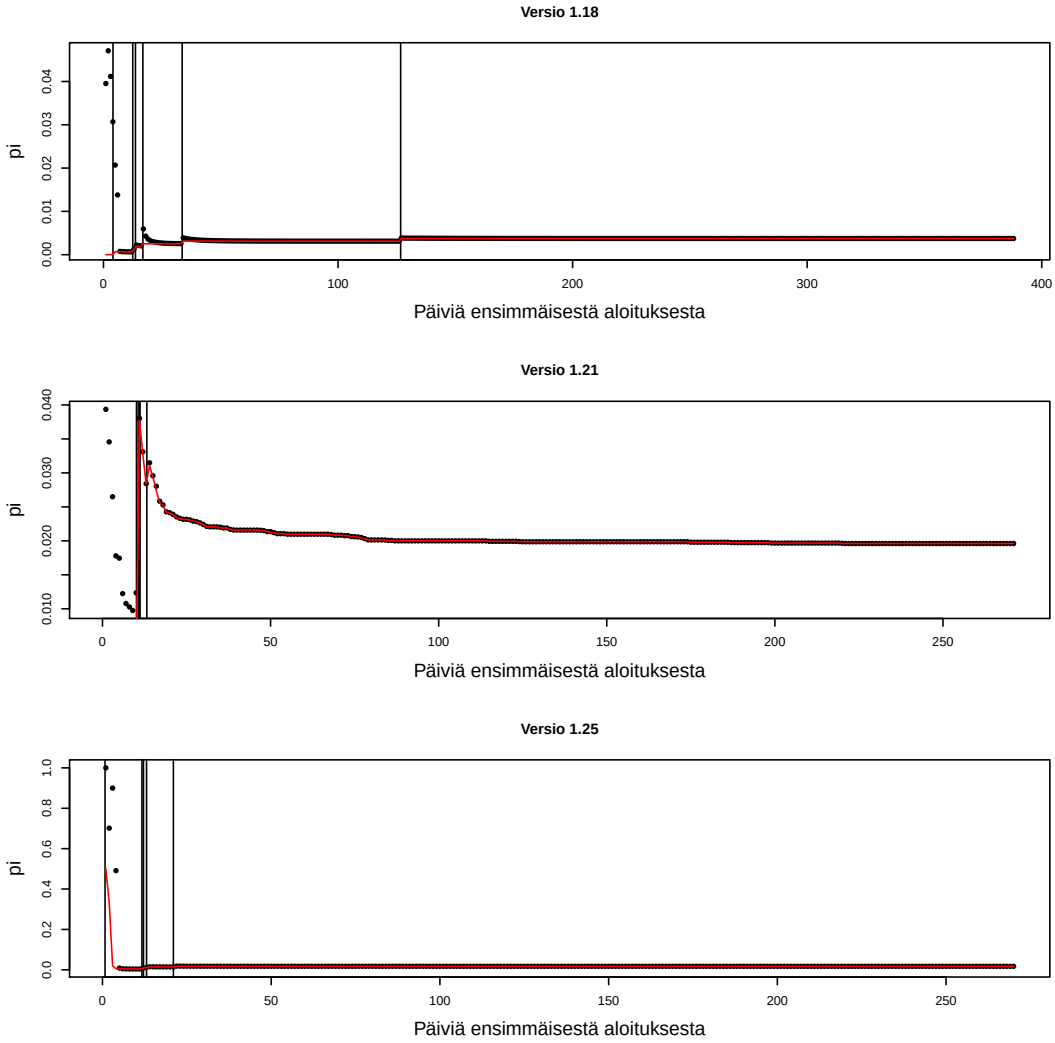
Mixture cure -mallin oletuksien toteutuminen on varmistettu, joten seuraavaksi voidaan varsinaisesti käyttää tätä mallia Hipster sheep -pelin datan aiemmin määriteltyihin osadatoihin. Selvitetään siis kyseisen pelin suosituimpien versioiden datoille ostotapahtuman tekevien pelaajien osuutta kuvaavan parametrin π ja ostotahtia kuvaavan parametrin λ arvot. Sensurointiajankohtaa muuttamalla voidaan selvittää, missä vaiheessa EM-algoritmilla lasketut parametrien arvot ovat samat kuin mitkä ne olisivat myöhemmillä sensurointiajankohdilla.

Määritetään kullekin tarkasteltavalle versiolle sensurointiajat päivän välein ensimmäisen pelaajan aloitusajasta viimeisen pelaajan lopettamisaikaan. Kullakin sensurointiajalla datasta piirrettävät kuvien 9 ja 10 mukaiset kuvat näyttäisivät hieman erilaiselta, mutta perusmuodoltaan ne kuitenkin olisivat kuvien 9 ja 10 kaltaiset. Sensurointiaikana voidaan käyttää todellista sensurointihetkeä eli havainnoinnin päättymishetkeä, jos halutaan verrata lyhemmillä tarkasteluväleillä laskettuja parametrien arvoja mahdollisimman pitkällä tarkasteluvälillä laskettuihin arvoihin. Tällä datalla se kuitenkin on turhaa, sillä kuvista 14 ja 15 nähdään, että jokaisella versiolla on jo käytettyjen sensurointiaikojen joukossa sellaisia ajankohtia, että monetisaatioprosentti π ei enää muutu.

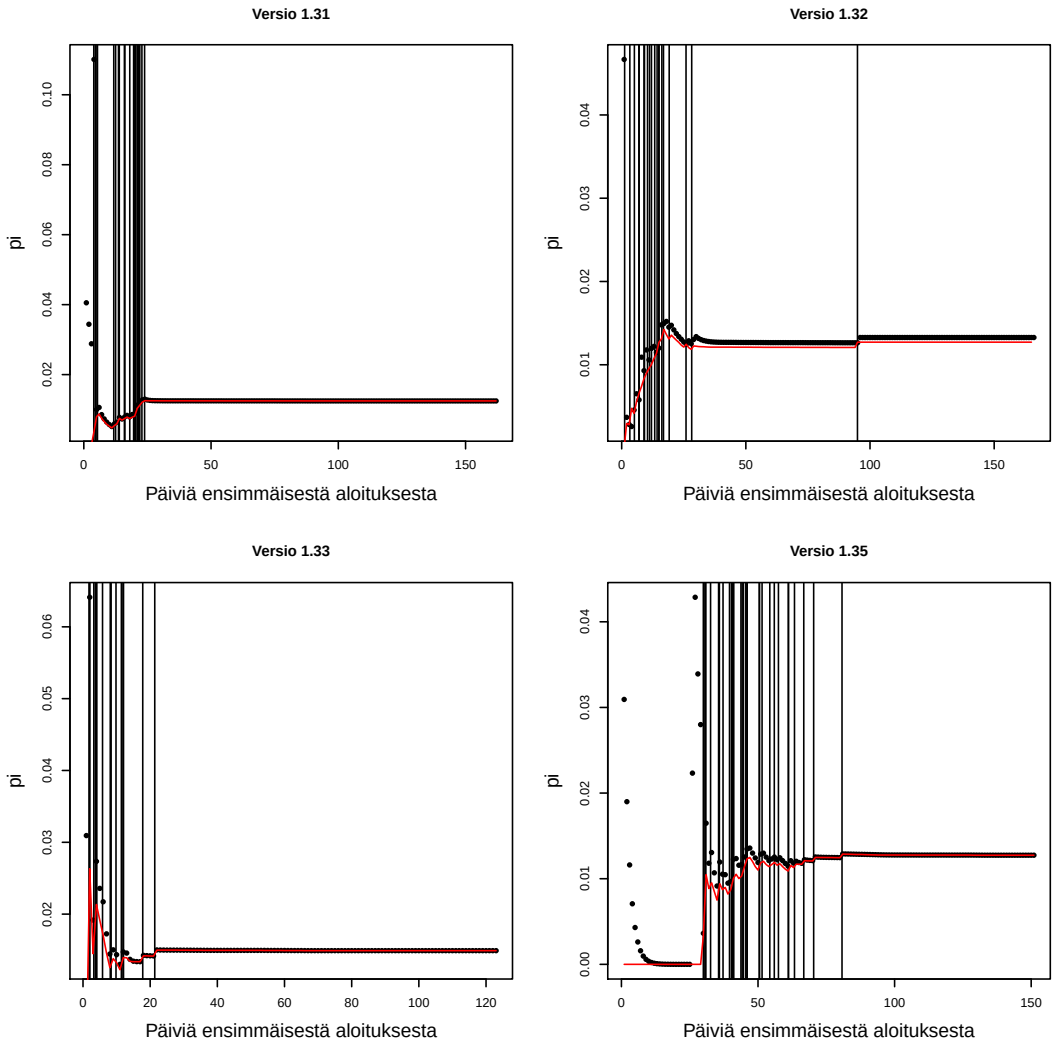
Kuvista 14 ja 15 nähdään, että suurin osa ostotapahtuman tekevistä pelaajista tekee sen ensimmäisen kuukauden kuluessa siitä, kun ensimmäinen pelaaja on kyseistä versiota alkanut pelata. Versio 1.35 poikkeaa muista versioista tässä asiassa, sillä kukaan sen pelaajista ei tee ostotapahtumaa ensimmäinen 30 päivän aikana, mutta sen jälkeen ostotapahtumia tehdään enemmän kuin missään muussa versiossa. Kuvasta 12 nähdään, että yksi aloitusaika on selvästi muita ennen, joten tämä poikkeus johtunee siitä. Kyseinen aloitusaika saattaa olla virheellinen, mutta asiaa ei saada varmistettua, joten annetaan kyseisen pelaajan olla datassa mukana.

Kuvista 14 ja 15 nähdään myös, että `cureEM`-funktiolla laskettu ostotapahtuman tekevien pelaajien osuus on melko sama kuin todellinen alttiiden yksilöiden osuus kullakin sensurointiajalla. Todellinen alttiiden yksilöiden osuus on kuitenkin jokaisella ajanhetkellä arvioitua osuutta pienempi. Mixture cure -malli huomioi myös mahdollisesti myöhemmin ostotapahtuman tekevät pelaajat, joten monetisaatioprosentin kuuluukin olla isompi kuin senhetkisestä datasta laskettu ostotapahtuman tehneiden pelaajien osuus.

Kuvasta 16 nähdään, että absoluuttisella sensurointiajalla laskettujen parametrien arvoilla piirretty välttöfunktion (36) kuvaaja vastaa erittäin hyvin osiossa 2.2 esitellyn epäparametrisen Kaplan–Meier-mallin kuvaajaa. Riittävän suurelle otokselle parametrisen mallin kuuluukin tuottaa samanlainen

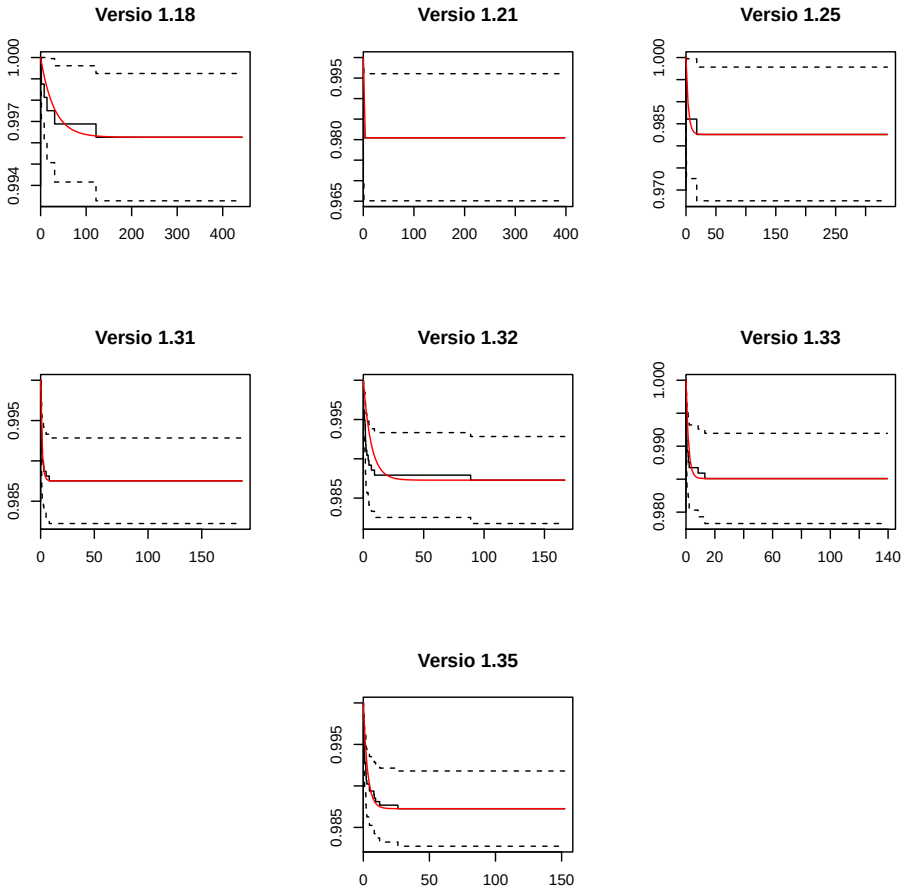


Kuva 14: Versioille 1.18, 1.21 ja 1.25 arvioitu monetisaatioprosentti päivän välein olevilla sensurointiajoilla. Pystyviivat kuvaavat ajanhetkiä, jolloin ostotapahtuman tehneiden pelaajien määrä on muuttunut. Punainen viiva kuvaa ostotapahtuman tehneiden pelaajien osuutta sen hetken kaikista pelaajista.



Kuva 15: Versioille 1.31, 1.32, 1.33 ja 1.35 arvioitu monetisaatioprosentti päivän välein olevilla sensurointiajoilla. Pystyviivat kuvaavat ajanhetkiä, jolloin ostotapahtuman tehneiden pelaajien määrä on muuttunut. Punainen viiva kuvaa ostotapahtuman tehneiden pelaajien osuutta sen hetken kaikista pelaajista.

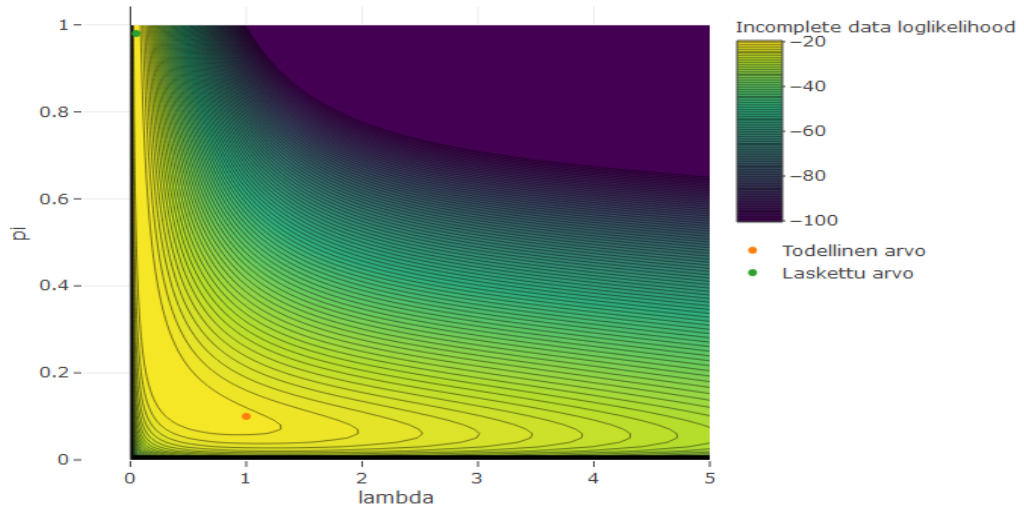
vastaus kuin epäparametrisen mallin, kunhan parametrisen mallin oletukset toteutuvat. Huomataan myös, että välttöfunktio laskee osalla versioista jyrkemmin kuin toisilla. Mitä nopeammin aloituksen jälkeen pelaajat tekevät ensimmäisen ostotapahtuman, sitä jyrkemmin välttöfunktion arvo laskee. Version 1.32 kuvaajasta nähdään, että punainen käyrä ei täysin mukaile epäparametrisen mallin käyrää. Tämä johtuu siitä, että version 1.32 datassa on yksi pelaaja, joka monetisoituu vasta pidemmän ajan kuluttua. EM-algoritmilla laskettu parametrin λ arvo lasketaan eräänlaisena keskiarvona, joten tämä myöhemmin monetisoituva pelaaja kasvattaa sen arvoa selvästi eikä saatu käyrä tästä syystä täysin vastaa epäparametrisen mallin käyrää.



Kuva 16: Kullekin versiolle piirretty välttöfunktion (36) kuvaaja absoluuttisella sensurointiajalla lasketuilla parametrien π ja λ arvoilla sekä Kaplan–Meier-mallin tuottamat välttöfunktion estimaatit 95 % luottamusväleineen.

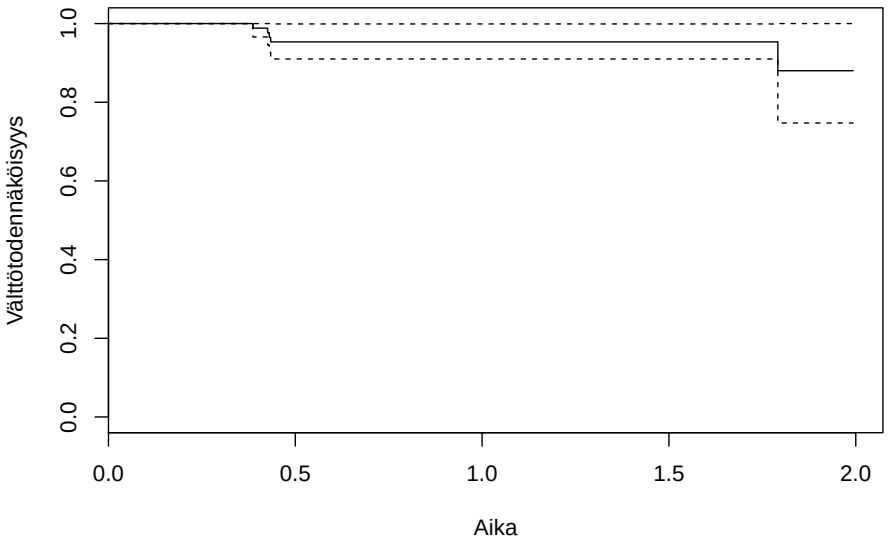
Joillain datajoukoilla tulee ongelmia funktion `cureEM` konvergenssin kanssa, jos käyttää hyvin pientä lopetuskriteeriä (esim. 10^{-9}) estimaattien laskemisessa. Malli ennustaa ensimmäisen ostotapahtuman jälkeen, että lähes kaikki pelaajat tekevät ostotapahtuman hitaalla konversiotahdilla. Tällaisessa tapauksessa ainoastaan muuttuja λ on mallia selittävä muuttuja eikä ole mahdollista erottaa elinaika-analyysin eksponenttijakauman mukaista mallia mixture cure -mallista.

Tämä ongelma ilmeni esimerkiksi version 1.15 datalla sensurointiajan ollessa 53 päivää ensimmäisen pelaajan aloituksesta. Ilmiö on kuitenkin selvemmin nähtävissä generoidulla datalla piirretyssä kuvaajassa 17. Kuvaaja on piirretty pienelle datalle ja ostotapahtumia on ollut vain muutamia. Kuvan 17 piirtämiseen käytetyn datan Kaplan–Meier-estimaatin kuvaaja esitetään kuvassa 18.



Kuva 17: Generoidulle testidatalle piirretty epätäydellisen datan logaritminen uskottavuusfunktion arvot muuttujien λ ja π funktiona. Uskottavuusfunktion arvoa vastaavat värit esitetään kuvan vieressä. Yhden alueen sisällä olevat uskottavuusfunktion arvot eroavat enintään yhdellä yksiköllä alueen suurimmasta arvosta.

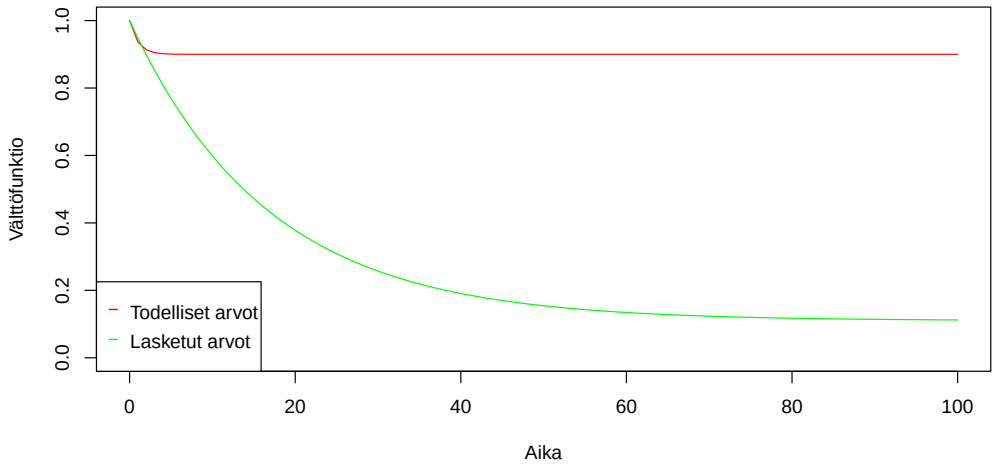
Huomataan, että kuvassa 17 esitetyt todellisten parametrien arvojen ja laskettujen parametrien arvojen pisteet ovat epätäydellisen datan logaritmisen uskottavuusfunktion isoimpien arvojen kaistaleella. Kuvan 18 perusteella vaikuttaa siltä, että vain pieni osa yksilöistä monetisoituu kohtuullisella tahdilla λ tai sitten kaikki yksilöt monetisoituvat erittäin hitaalla tahdilla.



Kuva 18: Kuvaajan 17 piirtämiseen käytetyn generoidun datan Kaplan–Meier-estimaatin kuvaaja.

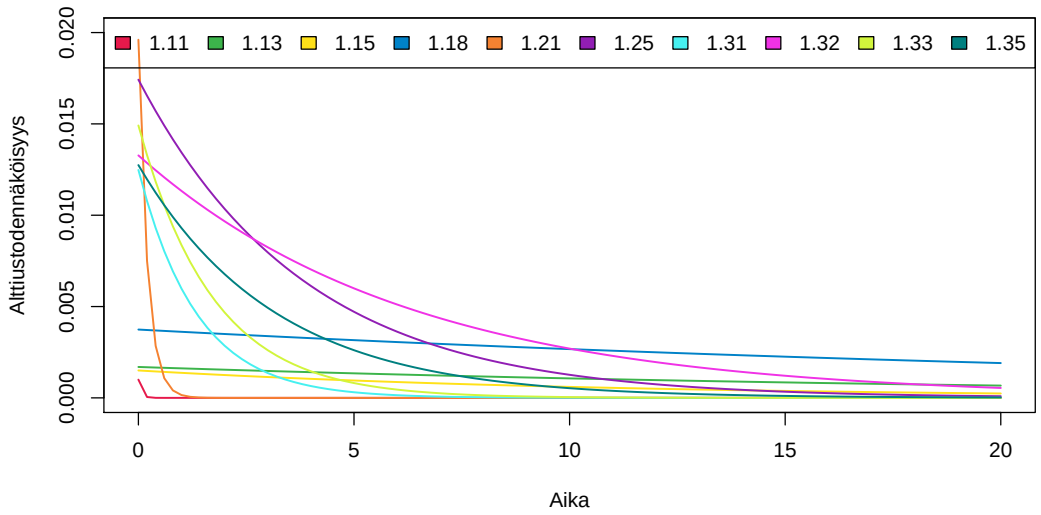
EM-algoritmin konvergenssin hitaus tälle datalle johtuu juuri uskottavien parametrien arvojen alueen laajudesta. Datasta ei näin lyhyellä seuranta-ajalla pysty erottamaan mixture cure -mallia tavallisesta eksponenttijakaumasta. Kuvasta 19 nähdään, että lyhyellä seuranta-ajalla välttöfunktiot lasketuilla ja todellisilla parametrien arvoilla saavat samanlaisia arvoja, mutta mitä pidempi seuranta-aika on, sitä isompi ero välttöfunktioiden arvoissa on.

Myös ensimmäisen ostotapahtuman tekevän pelaajan peliaika ennen ostotapahtuman tekemistä ilmeisesti vaikuttaa siten, että malli ennustaa lähes kaikkien pelaajien tekevän ostotapahtuman, jos ensimmäinen ostotapahtuma tehdään vain hieman ennen sensurointiajankohtaa. Mitä enemmän aikaa kuluu ostotapahtuman tekemisestä sensurointiajankohtaan, sitä paremmin malli ennustaa mallissa olevan immuunien pelaajien osuuden. Kaikilla versioittaisilla datoilla ei ole ongelmaa konvergenssin hitauden kanssa. Riippuu myös versiosta, että paljonko sensurointiajankohtaa pitää siirtää myöhemmäksi, jotta malli tunnistaa immuunien pelaajien osuuden olemassaolon. Suurimman uskottavuuden estimaattien konvergenssin perusteella kuitenkin on olemassa sellainen ajanhetki, että malli tunnistaa immuunien yksilöiden olemassaolon, vaikka se tapahtuisikin hitaasti.



Kuva 19: Välttöfunktion (36) arvot todellisilla parametrien arvoilla $\pi_g = 0.1$ ja $\lambda_g = 1$ ja funktiolla `cureEM` lasketuilla parametrien arvoilla $\lambda \approx 0.06$ ja $\pi \approx 0.89$.

Osiossa 3 mainittiin, että EM-algoritmilla saadaan laskettua jonkinlaiset uskottavat arvot puuttuvalle datalle. Tässä mallissa puuttuvaa dataa on alttiusindeksi ζ. Osiossa 4.1 esiteltiin kaava (40), jolla lasketaan todennäköisyys, että yksilö on altis tarkastelun kohteena olevalle tapahtumalle, vaikka se ei vielä ole sitä tehnyt. Nämä todennäköisyydet ovat luetteloinnissa 1 riveillä 18 laskettavat kertoimet. Sensuroiduille pelaajille ne muuttuvat jokaisella laskukierroksella ja sensuroimattomille yksilöille se on aina yksi, sillä sensuroimaton pelaaja on jo tehnyt ostotapahtuman. Kuvassa 20 on kullekin versiolle piirretty alttiustodennäköisyys ajanfunktiona. Huomataan, että aluksi todennäköisyydet vaihtelevat melko paljon versioittain. Ajan edetessä se kuitenkin lähetsyy nollaa jokaisella versiolla. Tästäkin kuvasta nähdään, että versioiden 1.11, 1.13 ja 1.15 tarkastelu ei edes olisi ollut järkevää, sillä jo heti aluksi alttiustodennäköisyys on hyvin pieni.



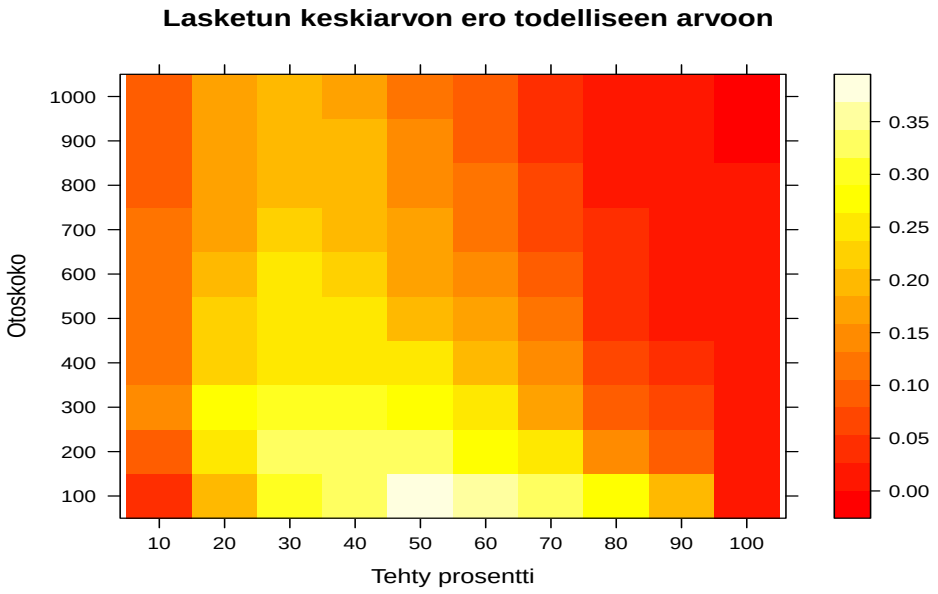
Kuva 20: Funktio (40) piirrettynä kullekin versiolle ajan suhteen arvolla $j = 1$ ja absoluuttisella sensurointiajalla lasketuilla parametrien λ ja π optimaalisilla arvoilla.

5.4 Mallin muuttujien vaikutus laskettuihin estimaatteihin

Keskitytään lopuksi vielä mallin toimintaan vaikuttavien muuttujien vaikutuksen voimakuuden tarkasteluun simuloimalla mallinmukaisia tilanteita. Tutkitaan otoskoon n, sensurointiajan määrittävän ostotapahtumien prosentin p ja datan generoinnissa käytettävän alttiustodennäköisyyden π_g vaikutusta EM-algoritmilla laskettavaan arvoon π.

Tarkastellaan ensin otoskoon n ja seuranta-ajan vaikutusta monetisaatioprosentin π estimaattiin. Seuranta-aika määritetään siten, että sensurointi tapahtuu siinä vaiheessa, kun p prosenttia alttiista yksilöistä on monetisoitunut. Generoidaan 1000 mallinmukaista otosta arvolla $\pi_g = 0.05$ kullekin (n, p)-parille. Lasketaan kullekin otokselle parametrien π ja λ arvot funktiolla `cureEM`. Otetaan keskiarvo saaduista parametrin π arvoista ja lasketaan sille 95 %:n luottamusväli. Datan generoinnin yhteydessä määritetään, että $\pi = 0$ ja $\lambda = 0$ niille otoksille, joissa sensuroimattomia yksilöitä ei ole ollenkaan. Lopulliset parametrien arvot riippuvat alkuarvauksesta, jos kukaan ei ole tapahtumaa tehnyt, joten määrittämällä niille arvo 0, pyritään välttämään parametrien jakauman vinoutuminen. Saatuja tuloksia havain-

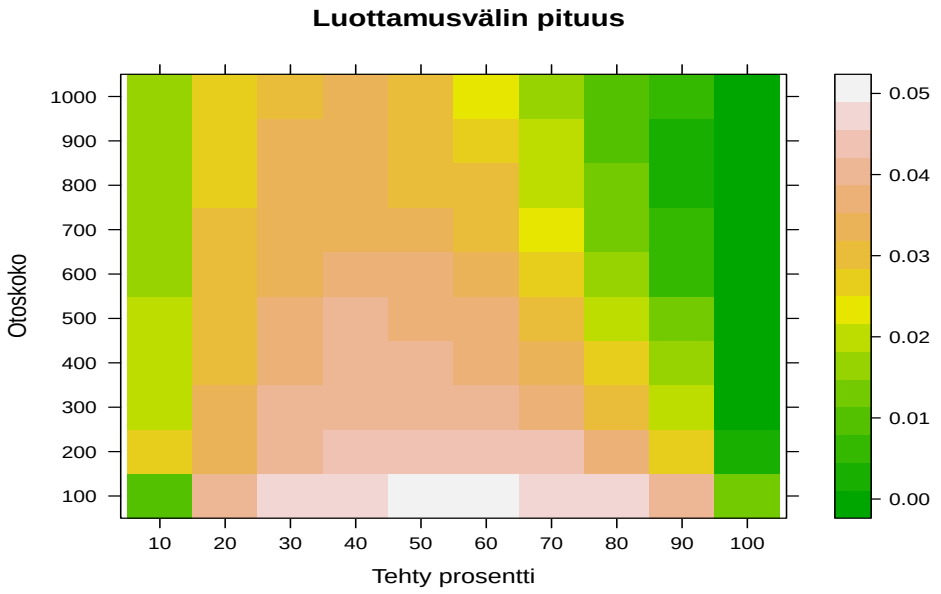
nollistetaan kuvilla 21 ja 22.



Kuva 21: Funktiolla `cureEM` laskettujen parametrin π estimaattien keskiarvojen ero todelliseen arvoon $\pi_g = 0.05$.

Kuvasssa 21 havainnollistetaan laskettujen estimaattien π keskiarvon eroa datan generoinnissa käytettyyn alttiiden yksilöisen osuuteen $\pi_g = 0.05$. Huomataan, että isoin ero on arvoilla $n = 100$ ja $p = 50$ %. Pienemmillä prosenteilla p ero on pienempi todennäköisesti siitä johtuen, että niissä tapauksissa on useita otoksia, joissa ei ollenkaan ole sensuroituja yksilöitä. Näin pienellä otoskoolla todennäköisesti suurimmalle osalle otoksia tehtyjen tapahtumien määrä näillä sensurointiajoilla on nolla, joten se tietysti pienentää keskiarvoa huomattavasti. Kuvasta 21 nähdään, että mitä isompi osa alttiista yksilöistä on jo monetisoitunut, sitä lähempänä lasketut arvot ovat todellista arvoa.

Näin pienellä alttiiden yksilöiden määrällä π_g ei kovinkaan monella 1000 otoksen simulaatiolla saatu sellaista arviota parametrille π, että ero todelliseen arvoon π_g olisi pienempi kuin todellisen arvon suuruus 0.05. Otoskokoa n kasvattamalla tällaisia arvoja saadaan useammin. Tällaisia arvoja ei saada kuin arvolla $p = 100$ %, jos otoskoko $n \leq 400$. Kun $n \geq 600$, tällaisia arvoja saadaan jo siinä vaiheessa, kun $p = 80$ %. Otoskokoa täytyy vielä kasvattaa, jotta saadaan lähes oikeansuuruinen estimaatti, pienemmällä tehtyjen tapahtumien prosentilla p. Kuitenkaan seuranta-aikaa ei voida määrittää arvoilla $p \leq 70$ %, vaikka otoskoko olisi 1000 yksilöä, jos saatu keskiarvo halutaan melko lähelle todellista arvoa. On kuitenkin mahdollista, että vielä isommal-

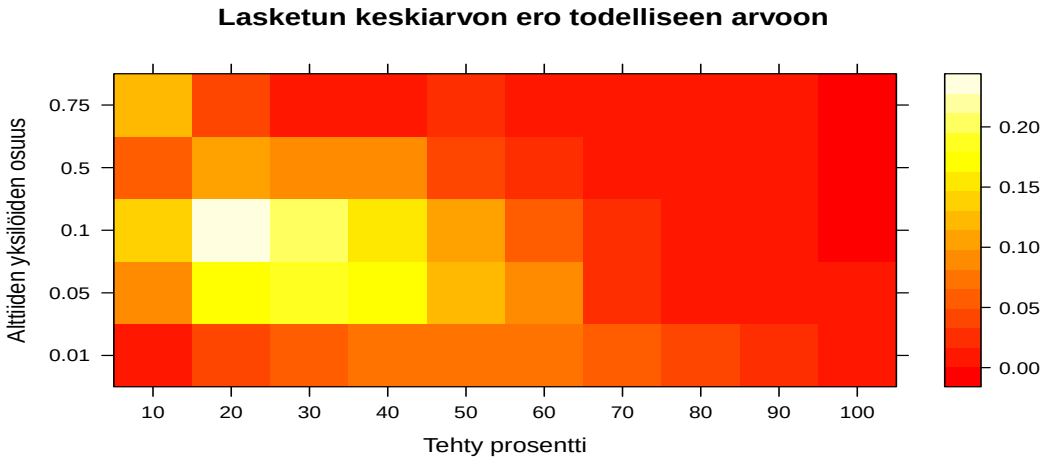


Kuva 22: Funktiolla `cureEM` laskettujen parametrin π estimaattien 95 %:n luottamusvälin pituus.

la otoskoolla ennustamiseen tarvittavaa tehtyjen tapahtumien prosenttia p saadaan vielä pienemmäksi.

Kuvasta 22 nähdään, että simuloinnilla laskettujen parametrin π estimaattien 95 %:n luottamusvälin pituus vaihtelee jonkinverran. Pienellä otoskoolla n se on isompi kuin isolla otoskoolla. Jälleen huomataan, että myös sensurointiajan määrittämiseen vaikuttava monetisoituneiden alttiiden yksilöiden osuus p vaikuttaa luottamusvälin pituuteen. Nyt kuitenkin sen vaikutus nähdään selkeämmin vasta isommilla muuttujien n ja p arvoilla.

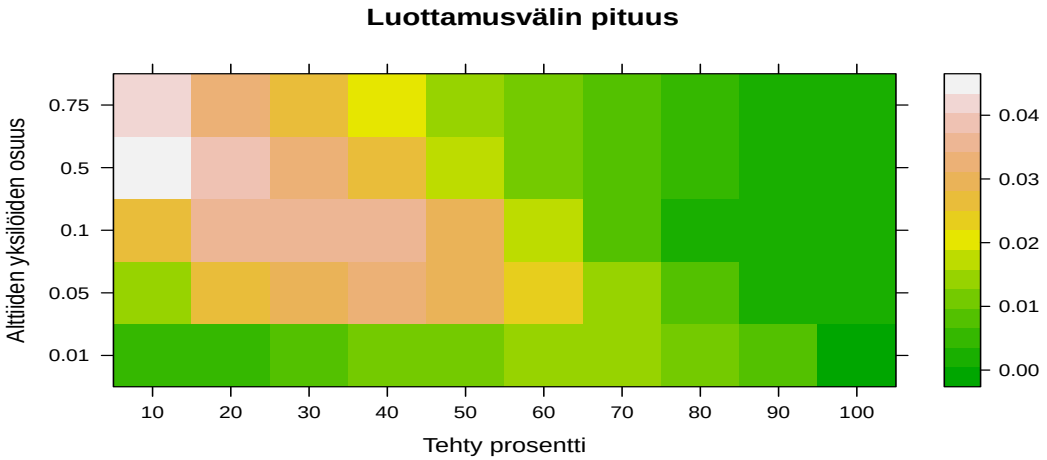
Tarkastellaan seuraavaksi parametrin π_g arvon vaikutusta lasketun monetisaatioprosentin π arvoon. Käytetään datan generoinnissa otoskokoa $n = 1000$. Generoidaan tuhat otosta kullekin (π_g, p)-parille ja lasketaan `cureEM`-funktiolla laskettujen parametrin π estimaattien keskiarvo ja 95 %:n luottamusväli. Tällä kertaa simuloinnissa muuttujina ovat datan generoinnissa käytettävä alttiusprosentti π_g ja sensurointiajan määrittävä tapahtumien osuus p kaikkien alttiiden yksilöiden määrästä. Generoidaan dataa arvoilla $\pi_g = 0.01, 0.05, 0.1, 0.5$ ja 0.75. Sensurointiajan määrittävä tehtyjen tapahtumien prosentti p saa samat arvot kuin otoskoon vaikutusta selvitettäessä. Kuvissa 23 ja 24 havainnollistetaan simuloinnin tuloksia esittämällä jälleen laskettujen parametrin π estimaattien keskiarvon ero todelliseen arvoon π_g ja 95 %:n luottamusvälit.



Kuva 23: Parametrin π estimaattien keskiarvon ero todellisesta alttiiden yksilöiden osuudesta π_g.

Kuvasta 23 nähdään, että mitä suurempi osa yksilöistä on alttiita, sitä pienempi osa tapahtumista täytyy tapahtua, jotta pystytään ennustamaan lopullinen muuttujan π arvo lähes oikein. Tästä kuvasta nähdään sama kuin kuvasta 21; alttiusprosentin arvolla $\pi_g = 0.05$ sensurointiaika täytyy määrittää siten, että 70 % alttiista yksilöistä on monetisoitunut. Arvolla $\pi_g = 0.01$ saadut tulokset ovat hyviä, mutta se todennäköisesti on seurausta sensuroimattomien yksilöiden puutteesta. Arvolla $\pi_g = 0.75$ ei tarvitse seurata yksilöitä kuin siihen asti, että 30 % alttiista yksilöistä tekee tapahtuman. Jo siinä vaiheessa otoksilla laskettujen arvojen π keskiarvo on lähellä datan generoinnissa käytettyä arvoa π_g. Huomataan, että yli puolet otoksesta täytyy olla alttiita yksilöitä, jotta sensurointiajankohta voidaan määrittää osuutta 50 % pienemmällä arvolla p ja tulos saadaan melko tarkasti ennustettua. Kuva 24 on melko samanlainen kuvioltaan kuin kuva 23, mikä tarkoittaa, että isommilla muuttujien π_g ja p arvoilla 1 000 otoksen estimaatti monetisaatioprosentista π ei vaihtele niin paljoa kuin pienemmillä muuttujien π_g ja p arvoilla.

Huomataan siis, että alle tuhannen yksilön otoksilla parametrin π suurimman uskottavuuden estimaatti $\hat{\pi}$ ei konvergoi todelliseen arvoon π_g ennen kuin lähes kaikki alttiit yksilöt ovat tehneet tarkastelun kohteena olevan tapahtuman. Alttiiden yksilöiden osuutta kasvattamalla, suurimman uskottavuuden estimaatti $\hat{\pi}$ konvergoi paremmin todelliseen arvoon π_g myös ly-



Kuva 24: Parametrin π estimaattien 95 % luottamusväli.

hyemmällä seuranta-ajalla.

6 Yhteenveto

Tässä tutkielmassa johdettiin EM-algoritmin vaatimat funktiot mixture cure -mallille. Mallissa osa yksilöistä on immuuneja tarkastelun kohteena olevalle tapahtumalle ja siinä käytetään oikealta sensurointia. Pääasiassa tässä tutkielmassa käytettiin tyypin I sensurointia, mutta vastaavalla tavalla mallin pitäisi toimia tyypin II tai satunnaiselle sensuroinnille. Mallin konvergenssi muodostuu kahdesta osasta: EM-algoritmi konvergoi suurimman uskottavuuden estimaattiin ja suurimman uskottavuuden estimaatti konvergoi parametrien todellisiin arvoihin. Mallin johtamisen jälkeen tarkasteltiin sen soveltuvuutta pelistä kerättyyn dataan. Soveltamisosion lopuksi tutkittiin datan tiettyjen piirteiden vaikutusta siihen, että suurimman uskottavuuden estimaatti $\hat{\pi}$ konvergoi todelliseen parametrin π arvoon.

Johdettua mallia sovellettiin vain yhdestä pelistä kerättyyn dataan. Ainakin kyseinen peli toteutti mallin oletukset, joten mahdollisesti tätä mallia voidaan soveltaa muistakin peleistä kerättyyn dataan. Tässä tutkielmassa keskityttiin enimmäkseen vain monetisaatioprosentin π ennustamiseen, mutta vastaavalla tavalla voitaisiin tutkia konversiotahtia λ.

Huomattiin, että malli ennustaa Hipster sheep -pelin datalle melko oikean arvon muuttujalle π jo siinä vaiheessa, kun vasta osa ostotapahtumista on tehty. Vaikuttaa siis siltä, että malli toimii tässä tutkielmassa käytetyn datan mukaiselle datalle. Ennustettu parametrin π arvo on vain hieman jo havaittua ostotapahtuman tekevien pelaajien osuutta suurempi, mikä tarkoittaa, että sensuroitujen pelaajien todennäköisyys ostotapahtuman tekemiseen on melko pieni. Tämä todennäköisyys pienenee hyvin nopeasti seuranta-ajan pidentyessä.

Mallin konvergenssiin vaikuttavia tekijöitä ovat otoskoko, sensurointiaikaan mennessä tehtyjen tapahtumien määrä sekä alttiiden yksilöiden osuus koko populaatiosta. EM-algoritmi laskee suuren määrän iteraatiokierroksia ennen konvergenssia, jos ostotapahtumia on vähän. Soveltamisosion lopussa tehdyistä testeistä huomattiin, että mitä isompi otos, sitä paremmin EM-algoritmilla laskettu suurimman uskottavuuden estimaatti vastaa todellisia arvoja. Huomattiin myös, että sitä paremmin suurimman uskottavuuden estimaatti vastaa todellisia arvoja, mitä isompi osa populaation alttiista yksilöistä on tehnyt tapahtuman ennen sensurointia. Myös alttiiden yksilöiden osuus kaikista yksilöistä vaikuttaa siten, että mitä suurempi todellinen osuus on, sitä lyhyemmällä seuranta-ajalla tulos pystytään ennustamaan. Malli siis ennustaa sitä paremmin, mitä enemmän monetisoituneita pelaajia on.

Mallin huonona puolena on, että liian pienellä datalla ja liian vähäisellä tapahtumien määrällä menetelmä ei tunnista eroa tämän mallin ja tavallisen elinaika-analyysin mallin välillä. Tällaisessa tilanteessa parametrien esti-

maatteihin liityy suurta tilastollista vaihtelua ja malli saattaa ennustaa jopa kaikkien pelaajien tekevän ostotapahtuman. Siispä tässä tutkielmassa käytetyn mallin käyttäminen ei ole järkevää, jos otoskoko on liian pieni, vain harva alttiista pelaajista on monetisoitunut sensurointiajankohtaan mennessä tai data on sellaista, että alttiita yksilöitä ei ylipäätään ole kovin monta.

Mallia pystytään kehittämään pienemmälle datalle sopivammaksi lisäämällä siihen arvio parametrien arvoista ennen datan näkemistä. Ennen datan näkemistä muun tiedon perusteella arvioidaan parametrien arvojen prioritiheysfunktioit, jotka myöhemmin päivitetään posterioritiheysfunktioiksi datan perusteella.

Kiitokset

Haluan kiittää ohjaajiani Markus Viljasta ja Marko Mäkelää. Erityisesti kiitos Markus Viljaselle mielenkiintoisesta aiheesta tälle tutkielmalle sekä ajatuksista mallin kehittämiselle. Kiitos myös ohjeista todellisen datan saamiseen sekä R-koodista, jolla sain havainnollistettua dataa kuvilla 9 ja 10.

Kirjallisuutta

[1] Tribeflame hipster sheep. http://www.tribeflame.com/games.html. Viitattu: 29.4.2018.

[2] D. R. Cox and D. Oakes. *Analysis of survival data*. Chapman and Hall Ltd, 1984.

[3] R. G. Miller Jr. *Survival analysis*, volume 66. John Wiley & Sons, 1981.

[4] D. F. Moore. *Applied Survival Analysis Using R*. Springer International Publisher Switzerland, 2016.

[5] G. McLachlan and T. Krishnan. *The EM algorithm and extensions*. John Wiley & Sons, New York, 1997.

[6] X. Liu. *Survival Analysis: Models and Applications*. Higher Education Press, 2012.

[7] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282): 457–481, 1958. URL http://www.jstor.org/stable/2281868.

[8] J.P. Klein, H.C. van Houwelingen, J.G. Ibrahim, and T.H. Scheike. *Handbook of Survival Analysis*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, 2016. URL https://books.google.fi/books?id=t1vOBQAAQBAJ.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. URL http://www.jstor.org/stable/2984875.

[10] C. F. J. Wu. On the convergence properties of the EM algorithm. *The Annals of statistics*, pages 95–103, 1983.

[11] M. Mäkelä. *Optimointialgoritmit*. Turun yliopisto, luentomoniste, Turku, 2016, . URL https://www.utu.fi/fi/yksikot/sci/yksikot/mattil/opiskelu/kurssit/Documents/moniste_2016.pdf. Viitattu 27.4.2018.

[12] M. Mäkelä. *Matemaattinen optimointi I*. Turun yliopisto, luentomoniste, Turku, 2015, . URL https://www.utu.fi/fi/yksikot/sci/yksikot/mattil/opiskelu/2013-14-kurssit/Documents/MonisteMatOpt1.pdf. Viitattu 27.4.2018.

[13] A. Wang, Y. Zhang, and Y. Shao. On the likelihood of mixture cure models. *Statistics & Probability Letters*, 131:51–55, 2017.

[14] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974. doi: 10.1109/TAC.1974.1100705.