

ARMA-mallin takaisinotantaluottamusväleistä

Jere Niemi

Perustieteiden korkeakoulu

Kandidaatintyö
Espoo 20.7.2023

Vastuopettaja

Prof. Pauliina Ilmonen

Työn ohjaaja

Prof. Pauliina Ilmonen

Copyright © 2023 Jere Niemi

Työn saa tallentaa ja julkistaa Aalto-yliopiston avoimilla verkkosivuilla. Muilta osin kaikki oikeudet pidätetään.

Tekijä Jere Niemi

Työn nimi ARMA-mallin takaisinotantaluottamusväleistä

Koulutusohjelma Teknistieteellinen kandidaattiohjelma

Pääaine Matematiikka ja systeemitieteet

Pääaineen koodi SCI3029

Vastuupettaja Prof. Pauliina Ilmonen

Työn ohjaaja Prof. Pauliina Ilmonen

Päivämäärä 20.7.2023

Sivumäärä 35+13

Kieli Suomi

Tiivistelmä

Takaisinotanta-algoritmi on tunnetusti hyvä tapa ARMA-mallin parametrien luottamusvälien estimointiin. On kiinnostavaa tietää, vaikuttaako ARMA-mallin virheiden jakauma algoritmin hyvyyteen, koska oikeassa elämässä aikasarjojen virheet voivat olla jakautuneet monin tavoin.

Työssä tutkittiin takaisinotantaluottamusvälien hyvyyttä varioimalla kolmea asiaa: virhejakaumaa, luottamustasoa ja alkuperäisen aikasarjan pituutta. Tutkittavat virhejakaumat olivat normaalijakauma, gammajakauma ja tasajakauma. Takaisinotanta-algoritmin hyvyyttä mitattiin tarkastelemalla, kuinka monta kertaa satunnaisesti luotu luottamusväli kattoi oikeat entuudestaan tiedetyt ARMA-mallin parametrit. Luottamusväli on hyvä, jos luottamusväli kattaa parametrit luottamustason osoittaman todennäköisyyden mukaisesti, esimerkiksi 80 prosentin luottamustasolle luottamusvälin kuuluisi kattaa parametrin 80 kertaa sadasta. Tutkimukset suoritettiin ARMA(2,3)-mallille.

Tutkimuksessa osoitettiin, että takaisinotanta-algoritmi suoriutui kohtuullisen hyvin jokaisella ARMA-mallin virhejakaumalla, jota tarkasteltiin ja luottamusväli kattoi oikean parametrin luottamustason mukaisesti tietyille luottamustasoille. Havaittiin, että alkuperäisen aikasarjan lyhentäminen johti pidempiin luottamusväleihin, mutta luottamusvälin oikean parametrin kattamiseen aikasarjan pituuden muuttaminen ei vaikuttanut. Luottamustason kasvattaminen johti välin muuttumiseen sitä konservatiivisemmaksi, mitä korkeampi luottamustaso oli. Viimeisin havainto johti lisätutkimuksiin, joista selvisi, että yli 80 prosentin luottamustasoilla luottamusväli tosiaan oli konservatiivinen ja luottamustason vaikutus oikean parametrin osumiseen ei ollut lineaarista, mutta kaikille virhejakaumatyypeille samanmuotoista. Matalilla luottamustasoilla parametrin osumistodennäköisyys luottamusvälille oli pienempää kuin sen olisi pitänyt, noin 75 prosentin kohdalla yhtä suurta ja yli 75 prosentin luottamustasoilla todennäköisempää kuin sen olisi pitänyt olla.

Tulevaisuudessa lisätutkimuksia voisi suorittaa useammalle todennäköisyysjakaumalle, jolloin voitaisiin varmistua luottamusvälille osumistodennäköisyyden kasvusta luottamusvälin funktiona ja tämän riippuvuuden kuvaajan muodosta. Koska takaisinotanta-algoritmi oli raskas suorittaa tavallisella kannettavalla tietokoneella, olisi pilvessä tehtävä laskenta tai tehokkaampien tietokoneiden hyödyntäminen suositeltavaa lisätutkimuksia tehdessä.

Avainsanat ARMA-malli, takaisinotanta, luottamusväli, luottamustaso

Author	Jere Niemi	
Title	About ARMA model bootstrap confidence intervals	
Degree programme	Bachelor's Programme in Science and Technology	
Major	Mathematics and Systems Sciences	Code of major SCI3029
Teacher in charge	Prof. Pauliina Ilmonen	
Advisor	Prof. Pauliina Ilmonen	
Date	Number of pages	Language
20.7.2023	35+13	Finnish

Abstract

The bootstrap algorithm is famously a good method for estimating the confidence intervals of ARMA model parameters. It is interesting to know whether the distribution of errors in the ARMA model affects the algorithm's effectiveness, because in real life, time series errors can be distributed in many ways.

The study examined the quality of bootstrap confidence intervals by varying error distribution, confidence level, and the length of the original time series. The error distributions under investigation were normal distribution, gamma distribution, and uniform distribution. The quality of the bootstrap algorithm was measured by examining how many times a randomly created confidence interval covered previously known ARMA model parameters. The confidence interval is good if it covers the parameters with a probability indicated by the confidence level, for example, for an 80% confidence level, the confidence interval should cover the parameter 80 times out of 100. The studies were conducted for the ARMA(2,3) model.

The study demonstrated that the bootstrap algorithm performed reasonably well for each error distribution examined, and the confidence interval covered the correct parameter in accordance with the confidence level for certain confidence levels. It was found that shortening the original time series led to longer confidence intervals, but changing the length of the time series did not affect the coverage of the correct parameter by the confidence interval. Increasing the confidence level led to the interval becoming more conservative. The latter observation led to further research, which revealed that at confidence levels above 80%, the confidence interval was indeed conservative, and the effect of the confidence level on hitting the correct parameter was not linear but had the same shape for all error distribution types. At low confidence levels, the probability of the parameter hitting the confidence interval was lower than it should have been. With a level of around 75% the hitting probability was equal, and above 75% confidence level was greater than expected.

Further studies could be conducted for several different types of probability distributions, which would confirm the increase in the probability of hitting the confidence interval as a function of the confidence interval and the shape of this dependence graph. Since the bootstrap algorithm was heavy to run on a regular laptop, it would be advisable to use cloud computing or utilize more powerful computers in further studies.

Keywords ARMA model, bootstrap, confidence interval, confidence level

Sisällys

Tiivistelmä	3
Tiivistelmä (englanniksi)	4
Sisällys	5
1 Johdanto	6
2 Aikaisempi tutkimus	7
3 Tutkimusaineisto ja -menetelmät	8
3.1 ARMA-malli	8
3.2 Takaisinotanta	8
4 Tulokset	11
4.1 Virhejakaumatyyppin varioinnin vaikutus luottamusvälin hyvyyteen . .	15
4.2 Luottamustason varioinnin vaikutus luottamusvälin hyvyyteen	18
4.2.1 50 % luottamustaso	18
4.2.2 90 % luottamustaso	21
4.3 Alkuperäisen aikasarjan pituuden varioinnin vaikutus luottamusvälin hyvyyteen	24
4.3.1 500 elementin aikasarja	24
4.3.2 100 elementin aikasarja	27
4.4 Luottamusvälille osumisen todennäköisyyden kehittyminen luottamus- tason funktiona	30
5 Yhteenveto	33
A Käytetty lähdekoodi	36

1 Johdanto

Aikasarjojen parametrien ja luottamusvälien estimoinneilla on monia käytännön sovelluksia kaikkialla, missä käsitellään aikasarjadataa. Esimerkki tällaisesta käytännön osa-alueesta on talousmatematiikka. Erilaisten aikasarjojen parametrien estimointi on helpottunut tietokoneiden laskentatehon kasvaessa. Ajan myötä onkin kehitetty tehokkaita algoritmeja parametrien ja parametrien luottamusvälien estimointiin, ja näistä jackknife-algoritmiin perustuva takaisinotanta-algoritmi [7] lienee yksi tunnetuimmista. Monet muut tunnetut algoritmit ovat hyviä estimoinnissa silloin, kun aikasarjan virheet ovat normaalijakautuneita, mutta takaisinotanta ei vaadi tätä oletusta. Tosielämässä kuitenkin kohtaamme usein tilanteita, joissa virheet ovat jakautuneet monin tavoin. Kuinka hyvin takaisinotanta onnistuu tällöin parametrien estimoinnissa?

Työn tavoitteena oli selvittää, kuinka hyvä takaisinotanta-algoritmi todellisuudessa on. Tätä tarkasteltiin tutkimalla ARMA-mallin takaisinotantaluottamusvälien hyvyttä. ARMA-malli on yksi yleisimmistä aikasarja-analyttisistä malleista ja siksi kiinnostava tutkimuskohde. Luottamusvälien hyvyyden tutkimista varten työssä varioitiin kolmea asiaa: aikasarjan virheiden jakaumaa, luottamustasoa sekä alkuperäisen aikasarjan pituutta. Luottamusväli on hyvä, jos se kattaa tunnetun parametrin luottamustason osoittaman todennäköisyyden mukaisesti, eli esimerkiksi 80 prosentin luottamustasolle luottamusvälin kuuluisi kattaa parametrin 80 kertaa sadasta. Sen lisäksi, että tarkasteltiin, kuinka monta kertaa satunnainen luottamusväli kattoi tunnetun parametrin, kerättiin tietoa myös luottamusvälien pituuksista.

Työssä tutkitut virhejakaumat olivat normaalijakauma, vino gammajakauma sekä tasajakauma. Kaikkia näitä jakaumia yhdisti se, että virheiden varianssi oli äärellinen ja niiden odotusarvo oli 0. Lisäksi työssä tutkittiin, kuinka luottamustason muuttaminen vaikutti luottamusvälien hyvyyteen. Lopuksi vielä tarkasteltiin, kuinka alkuperäisen aikasarjan pituuden lyhentäminen vaikutti luottamusväleihin.

Tutkimuksessa keskityttiin nimenomaan ARMA(2,3)-malliin. Mallin parametrit valittiin etukäteen. Työssä käytettyjen aikasarjojen alkuarvot luotiin keinotekoisesti hyödyntäen R:n satunnaislukuja. Lisäksi ARMA-aikasarjan luontiin tarvittavat virheet luotiin samasta jakaumasta kuin mistä aikasarjan alkuarvot. Itse aikasarja luotiin edellisiä arvoja, ja edellisiä sekä nykyhetken virheitä hyödyntäen. Tutkittava aineisto ja aikasarjat luotiin satunnaisesti uudelleen jokaiseen tutkittavaan skenaarioon.

Takaisinotannassa tutkimus rajattiin koskemaan vain satunnaisesti valittuja otosten pituuksia. Monia tutkimuksia on tehty liittyen optimaalisiin otoksen kokoihin. Näistä esimerkkinä on Bertailin ja Dudekin tutkimus [2]. Lisäksi rajattiin ARMA-mallin parametrien estimointi tehtäväksi automaattisesti R:n sisäänrakennetulla `arima()`-funktioilla.

2 Aikaisempi tutkimus

Takaisinotanta-algoritmin esitteli Bradley Efron vuonna 1979. Takaisinotanta on mielenkiintoinen algoritmi sen monien sovelluksien vuoksi. Tosielämässä aikasarjojen virheet voivat olla jakautuneet monin tavoin ja oletus siitä, että virheet olisivat riippumattomia ja samoin jakautuneita, ei useinkaan päde. Siksi takaisinotanta-algoritmi on yliverlainen, koska se ei vaadi oletusta virheiden riippumattomuudesta ja samoinjakautuneisuudesta toimiakseen eikä alkuperäisestä jakaumasta tarvitse tehdä oletuksia. Näin ollen sitä voidaan käyttää laajalti monimutkaisiin malleihin ja sillä voidaan saada kelvollisia tuloksia [6].

Vaikka takaisinotantaa voidaan käyttää laajalti, on luottamusvälien hyvydessä havaittu vaihtelua tutkimuksissa silloin, kun mallin virheiden riippumattomuus ja samoinjakautuneisuus eivät toteudu. Esimerkiksi Liu ja Singh tutkimuksessaan [13] toteavat, että takaisinotantaluottamusvälit ovat konservatiivisia silloin, kun oletetaan mallin olevan riippumaton ja samoinjakautunut, vaikka malli ei täyttäisikään oikeasti tätä oletusta. Konservatiivisilla luottamusväleillä tarkoitetaan, että luottamusvälit ovat pidempiä ja kattavat mallin parametrin useammin kuin sen kyseisellä luottamustasolla pitäisi [5].

Tässä tutkimuksessa varioitiin virhejakaumatyyppin lisäksi luottamustasoa ja alkuperäisen aikasarjan pituutta. Simulaatioiden määrä pidettiin vakiona kaikissa tutkimuksen vaiheissa. Hall tutkimuksessaan [8] toteaa, että simulaatioiden määrän lyhentäminen johtaa luottamusvälin pituuden kasvuun, eli luottamusvälit muuttuvat tällöin konservatiivisemmaksi.

Tässä tutkimuksessa rajattiin takaisinotanta käyttämään satunnaisen kokoista otosta. Muita otoksiin perustuvia takaisinotantamenetelmiä on tutkittu useasti. Esimerkkejä tästä ovat liikkuvan lohkon takaisinotanta [12], ei-päällekkäisen lohkon takaisinotanta [3], pyöreän lohkon takaisinotanta [16][14] sekä stationaarinen takaisinotanta [15], joka on lähinnä tässä tutkimuksessa käytettyä takaisinotantamenetelmää.

ARMA-mallien parametrien osalta takaisinotantaa on hyödynnetty menestyksekkäästi parametristimaattien standardivirheiden arvioimiseen [4]. Toimivaksi metodiksi samaan tarkoitukseen takaisinotanta todetaan myös Souzan ja Neton tutkimuksessa [17], jossa takaisinotanta-aineiston pituus oli tosin melko lyhyt, ollen vain 50 elementtiä pitkä.

Takaisinotannalla on tutkittu myös monia käytännön sovelluksia. Esimerkiksi Horowitz esittelee tutkimuksessaan, kuinka takaisinotantaa voidaan käyttää ekonometrian tueksi [10]. Lisäksi Zoubir ja Boashash ovat käyttäneet takaisinotantaa menestyksekkäästi signaalinkäsittelyssä [20] ja tarjoavat tämän tueksi monia käytännön esimerkkejä. Tosielämän ongelmissa ARMA-mallien takaisinotantaa, jota tässä työssä tutkittiin, on hyödynnetty esimerkiksi sähkömekaanisten tilojen ja alueiden vaimennussuhteiden selvittämiseksi oikealle Pohjois-Amerikan sähköverkkodatalle [1].

3 Tutkimusaineisto ja -menetelmät

3.1 ARMA-malli

ARMA-malli on tunnettu aikasarja-analyttinen malli, jolla voidaan analysoida ja ennustaa dataa, jolla on ajallista riippuvuutta. ARMA-, eli Autoregressive Moving Average -malli voidaan jakaa kahteen erilliseen yksinkertaisempaan malliin - autoregressiiviseen AR-malliin sekä liikkuvan keskiarvon MA-malliin.

Autoregressiivinen (AR) osa kuvaa sitä, miten aineiston edelliset tapahtumat vaikuttavat nykyhetkeen. Nimitys autoregressiivisyys tulee siitä, että malli on regressio itsestään [11]. Autoregressiivinen malli voidaan kuvata seuraavasti:

$$x(t) = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + \epsilon_t,$$

jossa $\phi_1, \phi_2, \dots, \phi_p$ ovat mallin parametrejä ja ϵ_t on virhetermi, jonka odotusarvo on 0 ja varianssi äärellinen. Termi p kuvastaa mallin kertalukua. Kertaluvulla tarkoitetaan sitä, kuinka monta ajanhetkeä taaksepäin malli ottaa huomioon.

Liikkuvan keskiarvon (MA) osa kuvaa sitä, kuinka paljon aineiston edellisten tapahtumien virhe vaikuttaa nykyhetkeen. Liikkuvan keskiarvon malli voidaan kuvata seuraavasti:

$$x(t) = \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q} + \epsilon_t,$$

jossa $\theta_1, \theta_2, \dots, \theta_q$ ovat mallin parametrejä ja $\epsilon_t, \epsilon_{t-1}, \dots, \epsilon_{t-q}$ ovat virhetermejä, joiden odotusarvo on 0 ja varianssi äärellinen. Liikkuvan keskiarvon malli käyttää menneitä ennustevirheitä regression kaltaisena mallina [11]. Termi q kuvastaa mallin kertalukua. Kertaluvulla tarkoitetaan sitä, kuinka monta virheen ajanhetkeä taaksepäin malli ottaa huomioon.

Yhdessä nämä muodostavat ARMA-mallin

$$x(t) = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q} + \epsilon_t.$$

Tässä tutkimuksessa keskitytään nimenomaan ARMA(2,3)-malliin, jossa AR-osiolla kertaluku $p = 2$ ja MA-osiolla kertaluku on $q = 3$. ARMA(2,3)-malli voidaan esittää seuraavasti:

$$x(t) = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \theta_3 \epsilon_{t-3} + \epsilon_t.$$

3.2 Takaisinotanta

Takaisinotanta on tietokonepohjainen menetelmä, joka perustuu otosten tekemiseen alkuperäisestä aineistosta. Menetelmän haittapuoli on se, että takaisinotanta kuluttaa paljon laskentaresursseja. Menetelmä pohjaa jackknife-menetelmään [7] ja sitä voidaan hyödyntää lähes minkä tahansa tilastollisen suureen luottamusvälin estimaatin luomiseen aineistosta.

Riippumattomien ja samoin jakautuneiden aineistojen tapauksessa takaisinotanta toimii siten, että tehdään satunnaisia pisteotoksia alkuperäisestä aineistosta niin kauan, kunnes käsissämme on alkuperäisen aineiston verran datapisteitä. Takaisinotantaotos ei kuitenkaan ole kopio alkuperäisestä datasta, koska satunnainen otos

tehdään joka kerta uudelleen alkuperäisestä aineistosta poistamatta jo käytettyä arvoa. Näin ollen osa arvoista voi toistua takaisinotantaotoksessa useasti ja osa ei ollenkaan. Joskus kuitenkin alkuperäinen aineisto on kooltaan liian suuri tehokkaan laskennan varmistamiseksi ja tällöin otoskoossa on tehtävä kompromisseja. Tämän jälkeen takaisinotanta suoritetaan useita kertoja uudelleen. Lopulta otosjoukoille voidaan laskea tutkittava tilastollinen suure.

Stokastisten prosessien tapauksessa on tärkeää säilyttää prosessin rakenne. Tällöin satunnaispistevalinta ei tule kysymykseen, vaan satunnaisotannalla valitaan pätkiä, joiden on oltava riittävän pitkiä säilyttääkseen stokastisen prosessin rakenteen. Itse takaisinotanta-algoritmi toimii samalla tavalla stokastisille prosesseille kuin riippumattomille samoin jakautuneille aineistoille - ainoa ero on siinä, että stokastisille prosesseille tutkittavat suureet analysoidaan takaisinotantaotospätkille sen sijaan, että analyysi suoritettaisiin pisteotoksista koostetulle uudelle aineistolle.

ARMA-malli on esimerkki stokastisesta prosessista. Näin ollen ARMA-aikasarjan tapauksessa takaisinotannassa on säilytettävä sarjan rakenne. ARMA-mallin takaisinotanta tehdään myös otoksina alkuperäisestä aineistosta. Tässä tutkimuksessa käytimme takaisinotantaa ARMA-mallin parametrien luottamusvälien selvittämiseen. ARMA-mallin parametrejä estimoitiin takaisinotannassa hyödyntämällä R:n sisäänrakennettua `arima()`-funktioita. Tämä funktio pyrkii minimoimaan mallissa esiintyvää virhetermiä ML-estimaattorin (maximum likelihood) avulla. Minimointi suoritetaan Kalmanin filtteriä hyödyntäen [9].

ARMA-mallin takaisinotanta-algoritmi voidaan esitellä seuraavalla tavalla. Oetaan ARMA-aikasarja $x_t, t = 1, 2, \dots, n$. Olkoon w lyhin aikasarjan pituus mallin parametrien estimoimiseksi. ARMA(2,3)-mallille w :n pienin mahdollinen arvo aikasarjan rakenteen säilyttämiseksi on 3, koska kyseinen malli huomioi kolme edellistä arvoa nykyhetken arvon laskemisessa. Parametrien estimointi näin lyhyelle aikasarjalle ei kuitenkaan ole mielekäästä ja tähän tutkimukseen otettiin hieman lyhintä mahdollisinta arvoa suurempi arvo. Olkoon täten $w = 10$ ja aikasarjan pituus $n = 1000$. Takaisinotanta toistetaan 1000 kertaa ja se koostuu seuraavista vaiheista:

1. Valitaan satunnainen otoksen pituus väliltä 10-1000.
2. Valitaan satunnainen aloituspiste siten, että koko otos saadaan irrotettua alkuperäisestä aineistosta.
3. Tehdään otos.
4. Estimoidaan otokselle ARMA-parametrit käyttäen R:n sisäänrakennettua `arima()`-funktioita ja tallennetaan tulos.

Alla takaisinotanta on esiteltynä pseudoalgoritimuodossa. ARMA-mallin parametrien takaisinotantaestimaatteja luova funktio ottaa parametreikseen alkuperäisen aikasarjan ja ARMA-parametrien lukumäärän ja palauttaa listan ARMA-parametrien estimaatteja.

```
function LUO_TAKAISINOTANTAESTIMAATIT(sarja, arma)
  toistot ← 1000                                ▷ Toistetaan takaisinotanta 1000 kertaa
```

```

minimipituus ← 10                                ▷ Aikasarjan minimipituus 10
for i ≤ toistot do
  satunnaispituus ← rand(minimipituus, length(sarja))
  aloituspiste ← rand(1, length(sarja) - satunnaispituus)
  to_sarja ← sarja[aloituspiste : (aloituspiste + satunnaispituus)]
  ARMA_estimaatit ← arima(to_sarja, arma)
  i ← i + 1
end for
return ARMA_estimaatit
end function

```

Näin saadaan 1000 estimaattia jokaiselle ARMA(2,3)-mallin parametrille. Takaisinotannan luottamusväli α , jossa α kuvaa tutkittavaa luottamustasoa, voidaan laskea ARMA-mallin parametreille seuraavalla funktiolla. Esitelty funktio ottaa parametreikseen luodut ARMA-parametriestimaatit sekä tutkittavan luottamustason ja palauttaa luottamusvälin ala- ja ylärajan.

```

function LUO_LUOTTAMUSVÄLI(estimaatit, luottamustaso)
  sortattu ← sort(estimaatit)                    ▷ Estimaatit suuruusjärjestykseen
  alaidx ← (1 - luottamustaso)/2 * length(estimaatit) + 1
  yläidx ← 1 - ((1 - luottamustaso)/2) * length(estimaatit)
  alाराजा ← estimaatit[alaidx]
  yläraja ← estimaatit[yläidx]
  return alाराजा, yläraja                        ▷ Palautetaan ala- ja yläraja
end function

```

Lopuksi ajetaan takaisinotanta-algoritmi 100 kertaa ja tutkitaan, kuinka monta kertaa valitun luottamustason mukainen luottamusväli kattaa oikean parametrin. Funktio ottaa parametreikseen alkuperäisen aikasarjan ja ARMA-parametrien lukumäärän sekä tutkittavan tunnetun parametrin ja palauttaa, kuinka monta kertaa luottamusväli kattaa parametrin.

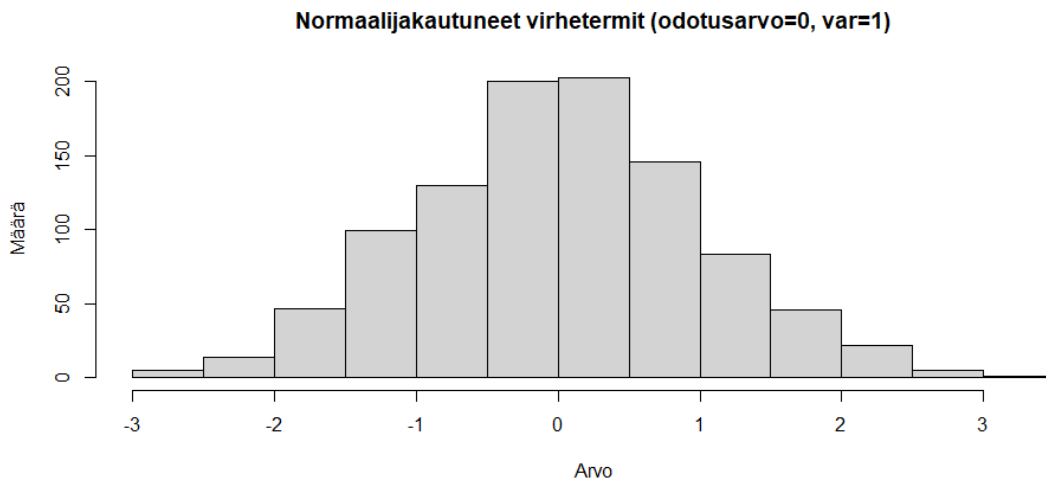
```

function LASKE_KATTAVUUSKERRAT(sarja, arma, parametri)
  toistot ← 100                                    ▷ Toistetaan algoritmi 100 kertaa
  väli_kattaa_parametrin ← 0
  estimaatit ← luo_takaisinotantaestimaatit(sarja, arma)
  luottamusväli ← luo_luottamusväli(estimaatit)
  for i ≤ toistot do
    if parametri ∈ luottamusväli then
      väli_kattaa_parametrin ← väli_kattaa_parametrin + 1
    end if
    i ← i + 1
  end for
  return väli_kattaa_parametrin
end function

```

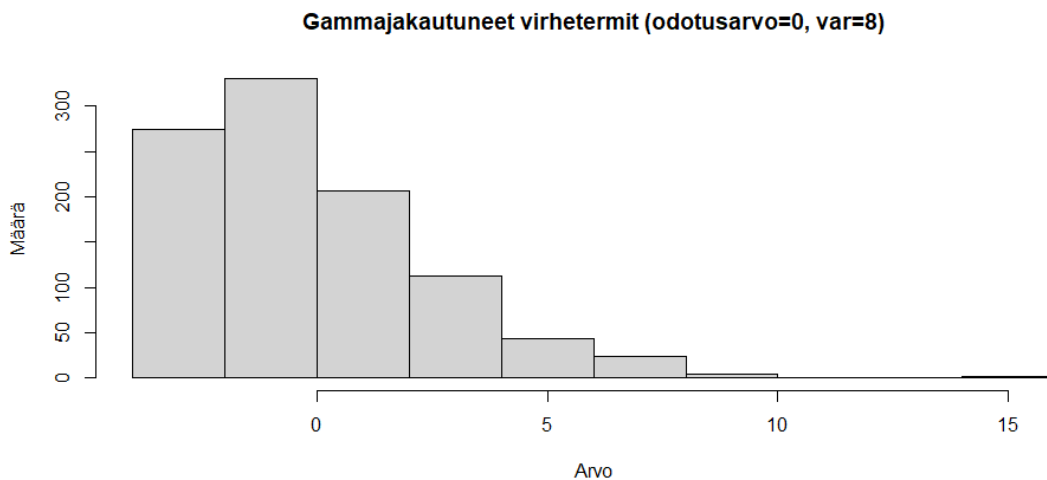
4 Tulokset

Kuvataan aluksi, miten testidata luotiin ja miten analysointi eteni vaihe vaiheelta. Aluksi luotiin vertailuaikasarjat. Tutkimuksen ARMA(2,3)-aikasarjan parametreiksi valittiin $\phi_1 = 0.5, \phi_2 = -0.3, \theta_1 = 0.02, \theta_2 = -0.15, \theta_3 = 0.5$. Ensimmäisessä tarkastelussa aikasarjan pituus oli $n = 1000$. Ensimmäisen aikasarjan virhetermit olivat normaalijakautuneita, joilla $\mu = 0, \sigma^2 = 1$, jossa μ on virheen odotusarvo ja σ^2 on virheen varianssi. Normaalijakautuneiden virhetermien histogrammi on esitelty Kuvassa 1. Virhetermit luotiin R:n komennolla `rnorm()`. Toisen aikasarjan virhetermit luotiin gammajakaumasta, joilla $\alpha = 2, \beta = 0.5$, jossa α kuvaa jakauman muotoa ja β on jakauman aste parametri, joka kuvaa, kuinka laajalle jakauma on levittäytynyt. Pienet aste parametrin arvot johtavat kapeaan, piikkimäiseen jakaumaan, kun taas suuret vastaavasti matalampaan, kukkulamaiseen muotoon [19]. Tätä jakaumaa siirrettiin x-akselin suuntaisesti siten, että odotusarvoksi saatiin $\mu = 0$. Jakauman varianssi oli $\sigma^2 = \frac{\alpha}{\beta^2} = \frac{2}{0.5^2} = 8$. Gammajakauman virhetermit luotiin R:n komennolla `rgamma()`. Näiden virhetermien histogrammi on esitelty Kuvassa 2. Kolmannen aikasarjan virhetermit luotiin tasajakaumasta, jonka alaraja oli $a = -5$ ja yläraja $b = 5$. Jakaumalle $\mu = 0, \sigma^2 = \frac{1}{12}(b - a)^2 = \frac{100}{12} \approx 8.3$. Nämä virhetermit on esitelty Kuvassa 3. Tasajakauman virhetermit luotiin R:n komennolla `runif()`.



Kuva 1: Histogrammi normaalijakautuneista virhetermeistä

Itse aikasarja generoitiin luomalla kolme ensimmäistä termiä hyödyntäen virhetermejä luovia funktioita ja loput termit riippuivat aina edellisistä termeistä tai edellisten termien virheistä sekä nykyhetken virheestä. Normaalijakautuneiden virhetermien tapauksessa aikasarjan kolme ensimmäistä termiä otettiin normaalijakaumasta. Esimerkki ARMA(2,3)-aikasarjasta, jonka virhetermit olivat normaalijakautuneita, on esitelty Kuvassa 4. Sama toistettiin myös muille jakaumille - gammajakautuneiden virhetermien aikasarjan kolme ensimmäistä termiä luotiin satunnaisesti gammajakaumasta samoilla parametreilla kuin jakauman virhetermit. Esimerkki ARMA(2,3)-aikasarjasta, jonka virhetermit olivat gammajakautuneita, on esitelty Kuvassa 5.

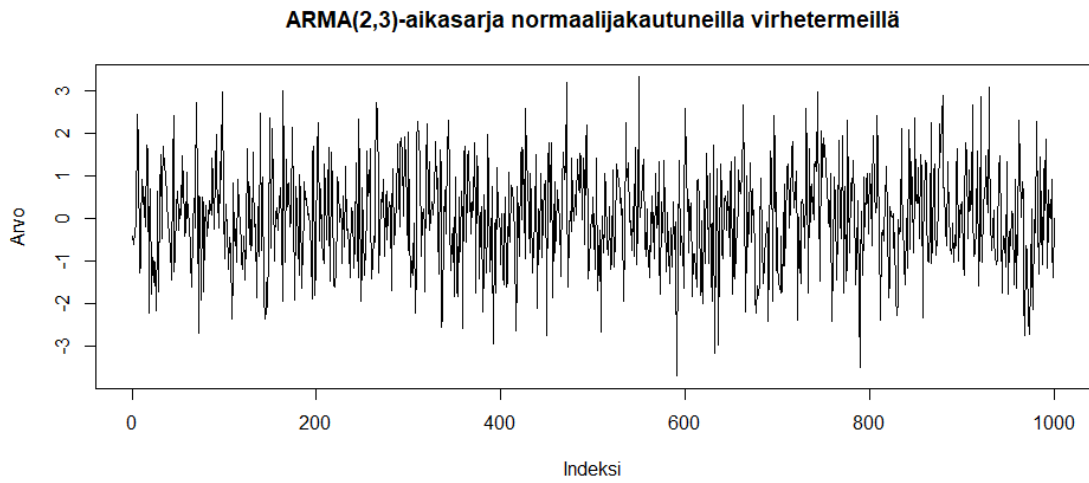


Kuva 2: Histogrammi gammajakautuneista virhetermeistä

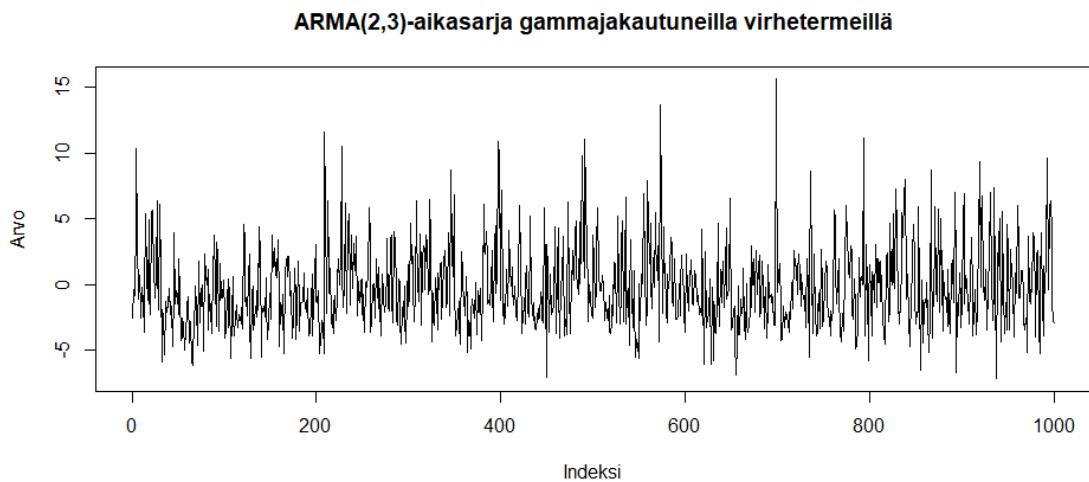


Kuva 3: Histogrammi tasajakautuneista virhetermeistä

Myös tasajakauman kolme ensimmäistä termiä luotiin samalla tavalla. Esimerkiksi ARMA(2,3)-aikasarjasta, jonka virhetermit olivat tasajakautuneita, on esitelty Kuvassa 6. Kaikkien aikasarjojen tapauksessa voidaan todeta, että niissä ei ollut trendiä, eikä aikasarjoista nopeasti katsomalla voi päätellä, mikä aikasarja on kyseessä. Mikä erotti gamma- ja tasajakautuneiden virhetermien avulla luodun aikasarjan normaalijakautuneiden virhetermien aikasarjasta, oli varianssin suuruus. Vertailuaikasarjan tapauksessa gamma- ja tasajakauman virheiden varianssi oli jopa kahdeksankertainen verrattuna normaalijakaumaan. Lisäksi näiden jakaumatyyppien muoto aiheutti yhdistettynä suureen varianssiin sen, että yksittäiset virheet saattoivat olla suuria ja niitä esiintyi useammin kuin normaalijakauman tapauksessa, jossa suurin osa arvoista osui jakauman odotusarvon ympärille. Näin ollen gamma- ja

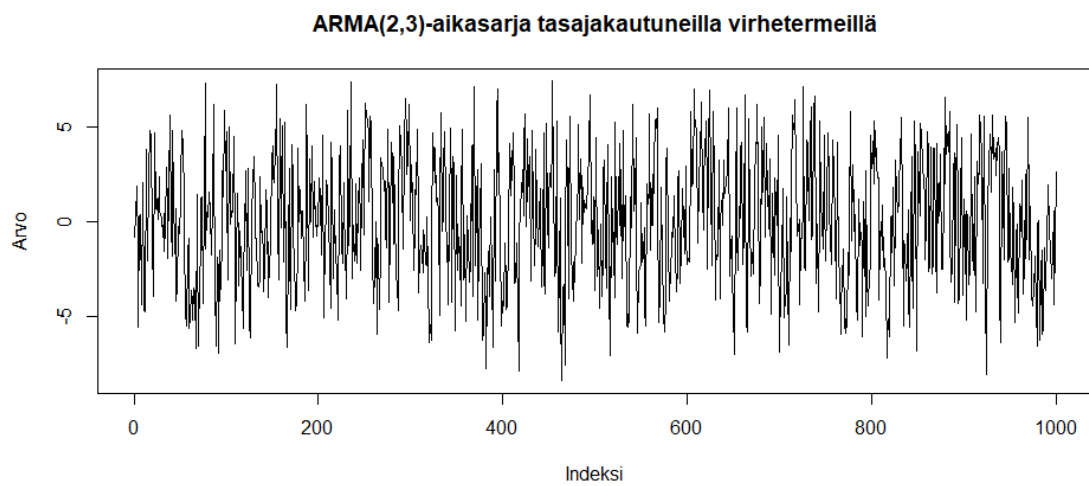


Kuva 4: Esimerkki luodusta aikasarjasta, jonka virhetermit olivat normaalijakautuneita



Kuva 5: Esimerkki luodusta aikasarjasta, jonka virhetermit olivat gammajakautuneita

tasajakautuneiden virheiden tapauksessa aikasarjaan muodostui satunnaisesti suuria yksittäisiä piikkejä.



Kuva 6: Esimerkki luodusta aikasarjasta, jonka virhetermit olivat tasajakautuneita

4.1 Virhejakaumatyyppien varioinnin vaikutus luottamusvälin hyvyyteen

Tarkastellaan seuraavaksi, miten virhejakaumatyyppien varioiminen vaikutti luottamusvälien hyvyyteen. Takaisinotanta-algoritmi suoritettiin Osion 3.2 mukaisella algoritmilla 100 kertaa. Joka kerralla tutkittava aineisto luotiin satunnaisesti uudelleen. Algoritmin raskaus havaittiin heti, sillä 100 iteraatiosyklin ajaminen vei aikaa lähes 7 tuntia.

Vertailukohdan luottamustasoksi valittiin 80 % luottamustaso. Tuloksena saatiin tieto siitä, kuinka monta kertaa sadasta alkuperäiset ARMA(2,3)-mallin parametrit olivat luottamusvälin sisällä. Tulokset on esitelty Taulukossa 1.

Taulukko 1: Tutkittujen virhejakaumien tulosten yhteenveto siitä, kuinka monta kertaa sadasta luottamusväli kattoi oikean parametrin. Aikasarjan pituus = 1000, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	85	87	89
ϕ_2	80	78	81
θ_1	86	86	85
θ_2	82	85	81
θ_3	80	79	78

Havaitaan, että kaikilla jakaumatyypeillä 80 % luottamusväli oli likimain oikea, kun tarkastellaan, kuinka monta kertaa luottamusväli kattoi tutkittavan parametrin. Keskimääräisesti voidaan todeta, että parametri oli hieman useammin luottamusvälin sisällä kuin sen olisi pitänyt olla, mutta ero ei ollut merkittävä. Näin ollen 80 % luottamustasolle luottamusväli oli melko hyvä.

Seuraavaksi tarkastellaan luottamusvälejä hieman tarkemmin. Taulukossa 2 on esitelty tunnuslukuja luottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Tulokset on esitelty neljän desimaalin tarkkuudella. Selitetään taulukon sarakkeet vasemmalta oikealle:

1. Tutkittava parametri
2. Pisimmän luottamusvälin pituus
3. Lyhimmän luottamusvälin pituus
4. Luottamusvälin pituuden keskiarvo
5. Luottamusvälin pituuden varianssi
6. Luottamusvälin alarajan keskiarvo
7. Luottamusvälin ylärajan keskiarvo

Taulukko 2: Tunnuslukuja takaisinotantaluottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	0.8194	0.2030	0.3845	0.0174	0.2574	0.6419
ϕ_2	0.6481	0.1540	0.3241	0.0107	-0.5188	-0.1947
θ_1	0.7904	0.1587	0.3611	0.0196	-0.1236	0.2375
θ_2	0.7869	0.1002	0.3402	0.0171	-0.2567	0.0835
θ_3	0.4792	0.0778	0.2089	0.0043	0.4240	0.6330

Taulukossa 3 on vastaavasti esitelty samat tulokset gammajakautuneiden virhetermien tapauksessa. ARMA-aikasarjan parametrien tunnusluvut näyttävät olevan pääpiirteittäin samaa suuruusluokkaa gammajakauman tapauksessa normaalijakautuneisiin virheisiin verrattuna. Gammajakauman tapauksessa pisin luottamusväli oli hieman pidempi kuin normaalijakauman tapauksessa termeillä ϕ_1 ja θ_1 . Ero saattaa selittyä satunnaisuudella. Tätä hypoteesia tukee myös se, että ala- ja ylärajan keskiarvoissa sekä luottamusvälin keskiarvopituudessa erot olivat hyvin pienet.

Taulukko 3: Tunnuslukuja takaisinotantaluottamusväleistä gammajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	0.8889	0.1853	0.3723	0.0206	0.2774	0.6397
ϕ_2	0.5624	0.1386	0.3104	0.0095	-0.5046	-0.1942
θ_1	0.8212	0.1428	0.3484	0.0205	-0.1240	0.2243
θ_2	0.7674	0.1485	0.3343	0.0181	-0.2692	0.0651
θ_3	0.3806	0.0794	0.1981	0.0042	0.4328	0.6310

Taulukossa 4 on esitelty avainluvut tasajakautuneiden virhetermien tapauksessa. Tulokset olivat jälleen samansuuntaisia verrattuna normaalijakautuneisiin virheisiin. Pisimmät luottamusvälit olivat muutamien parametrien osalta hieman pidempiä. Keskiarvopituuksissa merkittäviä eroja muihin virhejakaumatyyppisiin ei ollut havaittavissa.

Ajojen aikana havaittiin, että joskus R:n sisäänrakennettu arima()-funktio ei osannut löytää stationaarisia ratkaisuja, päätyen mahdollisesti nolnaan tai liian pitkään luottamusväliin. Virheitä esiintyi ajojen aikana muutamia kertoja, mutta virhetilanteen sattuessa ajettiin arima()-funktio uudelleen ja virheellinen tulos ohitettiin. 80 % luottamustason tapauksessa virheitä ei sattunut.

Nolla osui luottamusvälille Taulukon 5 mukaisesti. Nollan osuminen luottamusvälille on kiinnostavaa tutkia, koska jos nolla osuu luottamusvälille, voidaan todeta, että parametrillä ei ole tilastollisesti merkitsevää eroa nollasta [18]. Havaittiin, että niillä parametreillä, jotka jo alkujaan olivat lähellä nolaa, luottamusväli kattoi nollan useimmin. Näitä parametrejä olivat $\theta_1 = 0.02$ ja $\theta_2 = 0.15$. Tuloksista voitiin

Taulukko 4: Tunnuslukuja takaisinotantaluottamusväleistä tasajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.0940	0.1297	0.3545	0.0265	0.2692	0.6236
ϕ_2	0.6614	0.1470	0.3115	0.0094	-0.5150	-0.2035
θ_1	1.0718	0.1240	0.3339	0.0274	-0.1002	0.2338
θ_2	0.7739	0.1235	0.3302	0.0185	-0.2509	0.0793
θ_3	0.3510	0.0773	0.1927	0.0032	0.4318	0.6245

havaita, että virhejakaumatyyppi ei juurikaan vaikuttanut siihen, kattoiko luottamusväli nollan vai ei, vaan kaikille virhejakaumatyypeille nollan osumistodennäköisyys välille oli likimain sama. Useimmiten luottamusväli kattoi nollan parametrien θ_1 ja θ_2 tapauksissa. Kuitenkin muutamissa tapauksissa luottamusväli kattoi nollan myös parametreille ϕ_1 ja ϕ_2 . Sen sijaan parametrin θ_3 tapauksessa nolla ei ollut luottamusvälillä kertakaan minkään virhejakaumatyyppin tapauksessa.

Taulukko 5: Luottamusväli kattoi nollan (kerrat sadasta). Aikasarjan pituus = 1000, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	7	9	7
ϕ_2	8	5	5
θ_1	85	81	84
θ_2	71	65	72
θ_3	0	0	0

4.2 Luottamustason varioinnin vaikutus luottamusvälin hyvyteen

4.2.1 50 % luottamustaso

Tarkastellaan seuraavaksi, kuinka hyvin takaisinotantaluottamusvälit voitiin estimoida 50 % luottamustasolle. Huomionarvoista on se, että tavallisesti ei ole mielekästä tarkastella näin pientä luottamustasoa. Tulokset tälle luottamustasolle on esitelty Taulukossa 6.

Taulukko 6: Tutkittujen virhejakaumien tulosten yhteenveto siitä, kuinka monta kertaa sadasta luottamusväli kattoi oikean parametrin. Aikasarjan pituus = 1000, luottamustaso = 50 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	41	33	37
ϕ_2	44	44	47
θ_1	40	42	36
θ_2	41	37	36
θ_3	44	46	42

Havaitaan, että luottamusväli kattoi tutkittavan parametrin 50 % luottamustasolla huomattavasti harvemmin, kun verrataan 80 % luottamustasoon. 50 % luottamustason tapauksessa osuminen oli huomattavasti harvinaisempaa. Näin sen kuuluukin olla luottamustason laskiessa, mutta huomionarvioista on se, että väli lyheni liikaa. Normaalijakauman tapauksessa luottamusväli kattoi tutkittavan parametrin suhteellisen tasaisesti kaikkien ARMA-parametrien tapauksessa, eli väli kattoi jokaisen parametrin vain hieman yli 40 kertaa sadasta. Gamma- ja tasajakaumien tapauksissa hajontaa parametrien välillä oli vastaavasti hieman enemmän. Esimerkiksi gammajakauman ϕ_1 -parametrin luottamusväli kattoi vain 33 kertaa ja θ_2 -parametri 37 kertaa sadasta. Muut gammajakautuneiden virhetermien ARMA-parametrit luottamusväli kattoi hieman yli 40 kertaa. Tasajakautuneiden virheiden tapauksessa samat ϕ_1 - ja θ_2 -parametrit erottuivat joukosta alle 40 % osumistodennäköisyydellä. Lisäksi parametri θ_1 jäi alle 40 kerran sadasta tasajakautuneiden virhetermien tapauksessa.

Taulukko 7: Tunnuslukuja takaisinotantaluottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 50 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alараја}$	$ka_{yläraја}$
ϕ_1	0.7217	0.0494	0.1303	0.0063	0.4055	0.5358
ϕ_2	0.2495	0.0436	0.1064	0.0017	-0.3730	-0.2666
θ_1	0.7241	0.0417	0.1198	0.0058	-0.0159	0.1039
θ_2	0.4972	0.0312	0.1137	0.0046	-0.1810	-0.0673
θ_3	0.1416	0.0226	0.0630	0.0005	0.4795	0.5425

Tarkastellaan seuraavaksi luottamusvälien avainlukuja. Normaalijakauman 50 % luottamustason tulokset on esitelty Taulukossa 7. Normaalijakauman tapauksessa havaitaan, että luottamusväli oli selvästi lyhyempi verrattuna 80 % luottamustasoon. AR-parametrien sekä ensimmäisen MA-parametrin tapauksessa keskiarvopituus oli 0.3 ja 0.4 välillä. Nyt samojen parametrien osalta keskiarvopituus oli 0.1 ja 0.15 välillä. Toisen MA-parametrin pituus oli 80 % luottamustason tapauksessa noin 0.2 ja 50 % luottamustasolla se oli noin 0.06. Myös kaikkien parametrien pituuksien varianssi oli merkittävästi pienempi kuin vertailutapauksessa. Takaisinotanta-algoritmin antamat pisimmät mahdolliset pituudet olivat yhä varsin pitkiä 50 % luottamustason tapauksessa. Merkillepantavaa on se, että kaikkien parametrien keskiarvoala- ja ylärajat muodostivat uuden luottamusvälin, jonka pituus oli huomattavan lyhyt ja oikeat parametrit osuvat tämän luottamusvälin sisään. Esimerkiksi ensimmäiselle AR-parametrille tämä uusi arvioitu väli olisi 0.41-0.54 ja oikea parametri (0.5) osuisi tämän välin sisään. Kuitenkin aineisto luotiin joka iteraation alussa uudelleen ja tätä mahdollisuutta ei oikean datan kanssa olisi, joten tämän löydöksen sovellusmahdollisuudet ovat varsin rajatut.

Taulukko 8: Tunnuslukuja takaisinotantaluottamusväleistä gammajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aika-sarjan pituus = 1000, luottamustaso = 50 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{ylaraja}$
ϕ_1	0.5113	0.0498	0.1276	0.0055	0.4075	0.5350
ϕ_2	0.2768	0.0419	0.1062	0.0019	-0.3661	-0.2599
θ_1	0.4697	0.0388	0.1190	0.0049	-0.0159	0.1030
θ_2	0.4504	0.0299	0.1100	0.0040	-0.1834	-0.0734
θ_3	0.1801	0.0237	0.0677	0.0007	0.4737	0.5413

Myös gammajakautuneiden tapauksessa luottamusvälien pituudet olivat laskeneet merkittävästi verrattuna gammajakautuneiden tuloksiin 80 % luottamustason tapauksessa. Gammajakautuneiden 50 % luottamusvälin tulokset on esitelty Taulukossa 8. Kun vertaillaan 50 % luottamustason tuloksia gamma- ja normaalijakautuneiden virhetermien tapauksissa, huomataan, että parametrien ϕ_1 ja θ_1 pisimmät luottamusvälit olivat huomattavasti lyhyempiä kuin normaalijakauman tapauksessa. Muiden parametrien osalta erot olivat pienempiä. Pisimmät luottamusvälit olivat vain hieman pidempiä kuin normaalijakautuneiden virheiden tapauksessa. Keskiarvopituuksien osalta merkittävää eroa ei ollut havaittavissa, joten ero selittynee takaisinotanta-algoritmiin liittyvällä satunnaisuudella. Kun satunnaisuuden vaikutus jätetään huomiotta, tulokset olivat hyvin samankaltaiset kuin normaalijakauman tapauksessa - myös pituuden varianssi oli hyvin linjassa normaalijakauman tuloksiin.

Tasajakautuneiden virhetermien tapauksessa havaittiin edellisten tapaan, että luottamusvälien pituudet olivat merkittävästi lyhentyneet verrattuna 80 % luottamustason pituuksiin. Tasajakautuneiden tulokset on esitelty Taulukossa 9. Merkillepantavaa on se, että pisimmät välit olivat huomattavasti lyhyempiä kuin gamma- sekä normaalijakaumien tapauksissa. Lisäksi pituuksien varianssit saivat huomattavasti pienempiä

Taulukko 9: Tunnuslukuja takaisinotantaluottamusväleistä tasajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 50 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	0.3849	0.0390	0.1122	0.0024	0.4306	0.5428
ϕ_2	0.2687	0.0438	0.1083	0.0019	-0.3787	-0.2704
θ_1	0.3775	0.0419	0.1022	0.0020	-0.0151	0.0872
θ_2	0.3035	0.0361	0.1057	0.0028	-0.1705	-0.0648
θ_3	0.1893	0.0310	0.0695	0.0007	0.4761	0.5456

arvoja verrattuna muiden virhejakaumiin. Keskiarvopituuksissa ei ollut merkittävää eroa verrattuna gamma- ja normaalijakaumiin. Myös tasajakautuneiden tapauksessa luottamusvälien maksimipituuksien erot saattoivat selittyä satunnaisuudella, mutta tasajakautuneiden näytti tuottavan hieman lyhyempiä luottamusvälejä kuin gamma- ja normaalijakauma.

Nolla osui luottamusvälille huomattavasti harvemmin kuin 80 % luottamustasolla. Tulokset nollan esiintymiselle 80 % luottamustasolla on esitelty Taulukossa 10. Tätä selittänee lyhentynyt luottamusvälin pituus. Luottamusväli kattoi nollan parametreille θ_1 ja θ_2 noin puolet niistä kerroista verrattuna 80 % luottamustasoon. Muilla parametreilla osumista tapahtui vähän tai ei laisinkaan.

Taulukko 10: Luottamusväli kattoi nollan (kerrat sadasta). Aikasarjan pituus = 1000, luottamustaso = 50 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakautuneet
ϕ_1	1	3	0
ϕ_2	0	1	0
θ_1	41	39	35
θ_2	17	18	20
θ_3	0	0	0

4.2.2 90 % luottamustaso

Seuraavaksi tarkastellaan, kuinka takaisinotantaluottamusvälit käyttäytyivät 90 % luottamustasolla. Tulokset tälle luottamustasolle on esitelty Taulukossa 11.

Taulukko 11: Tutkittujen virhejakaumien tulosten yhteenveto siitä, kuinka monta kertaa sadasta luottamusväli kattoi oikean parametrin. Aikasarjan pituus = 1000, luottamustaso = 90 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	99	97	99
ϕ_2	96	96	91
θ_1	99	97	100
θ_2	95	94	93
θ_3	97	96	95

Havaitaan, että luottamusväli kattoi tutkittavan parametrin huomattavasti useammin, kun verrataan 80 % luottamustason tuloksiin. 90 % luottamustason tapauksessa osuminen oli huomattavasti todennäköisempää. Tämä oli odotettava tulos luottamustason kasvaessa, mutta huomionarvoista on se, että väli piteni liikaa. Voidaan sanoa, että 90 % luottamustason tapauksessa luottamusvälit olivat konservatiivisia, eli väli kattoi oikean parametrin todennäköisemmin kuin luottamustason osoittamalla todennäköisyydellä. Normaalijakauman tapauksessa väli kattoi parametrin jokaisen ARMA-parametrin osalta yli 95 kertaa sadasta, ja parametrien keskinäisessä osumistodennäköisyydessä ei havaittu merkittäviä eroja. Gammajakauman tulokset olivat lähestulkoon identtiset normaalijakauman kanssa, ja näiden kahden jakaumatyyppien erot voivat selittyä pelkällä satunnaisuudella. Tasajakaumassa voitiin sen sijaan havaita isompaa hajontaa eri parametrien välillä. Esimerkiksi θ_1 -parametrin luottamusväli kattoi joka kerta, kun taas ϕ_2 -parametrin vain 91 kertaa sadasta. Tässäkin tapauksessa erot kahteen muuhun jakaumaan olivat melko pieniä.

Tarkastellaan seuraavaksi luottamusvälien avainlukuja. Normaalijakautuneiden virheiden tapauksessa luottamusvälit olivat huomattavasti pidempiä kuin 80 % luottamustason tapauksessa. Nämä tulokset on esitelty Taulukossa 12. Pisimmät välit olivat jopa yli 0.3 yksikköä pidempiä kuin 80 % luottamustasolla. Keskiarvopituus vastaavasti 80 % luottamustasolla oli noin 0.35 ja nyt hieman alle 0.6, joten luottamusvälit olivat pidentyneet voimakkaasti. Näin pitkät luottamusvälit alkavat olla jo epämieluisia, sillä oikean parametrin estimointi on haastavaa.

Gammajakautuneilla virheillä luottamusvälit kasvoivat samankaltaisesti. Tulokset gammajakautuneille virheille on esitelty Taulukossa 13. Suurimmat erot normaali- ja gammajakauman välillä syntyivät pisimpien luottamusvälien pituuksien tapauksissa. Ne voivat kuitenkin selittyä vain sattumalla. Luottamusvälien keskimääräinen pituus oli lähestulkoon sama kuin normaalijakautuneilla virheillä. Myös keskimääräiset luottamusvälin ylä- ja alarajat olivat miltei samat.

Tasajakautuneille virheille kehitys oli hyvin samansuuntaista. Tulokset tasajakautuneille virheille on esitelty Taulukossa 14. Luottamusvälit olivat pidempiä kuin 80 % luottamustasolla. Erot gamma- ja normaalijakaumiin olivat melko pieniä.

Taulukko 12: Tunnuslukuja takaisinotantaluottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 90 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.4401	0.2825	0.6129	0.0354	0.1014	0.7143
ϕ_2	0.7822	0.2823	0.5109	0.0143	-0.6220	-0.1111
θ_1	1.3915	0.2472	0.5841	0.0353	-0.1951	0.3889
θ_2	1.1560	0.2309	0.5820	0.0328	-0.3471	0.2349
θ_3	0.6352	0.1742	0.3576	0.0067	0.3580	0.7157

Taulukko 13: Tunnuslukuja takaisinotantaluottamusväleistä gammajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 90 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.2733	0.2444	0.6521	0.0368	0.0600	0.7121
ϕ_2	0.9232	0.2710	0.5028	0.0154	-0.6149	-0.1121
θ_1	1.2585	0.3203	0.6312	0.0371	-0.1946	0.4366
θ_2	1.1082	0.2529	0.5899	0.0289	-0.3397	0.2502
θ_3	0.6245	0.1525	0.3654	0.0086	0.3526	0.7180

Edelleen keskiarvopituus oli samaa suuruusluokkaa kuin muilla virhejakaumatyypeillä. Näin ollen voidaan todeta, että kaikille virhejakaumatyypeille luottamustason muuttaminen johti samankaltaiseen käyttäytymiseen luottamusväleissä, eikä suoraan luottamusvälien tunnuslukuja katsomalla voida päätellä, mitkä tulokset tulivat mistäkin virhejakaumasta. Luottamustaso sen sijaan vaikutti hyvinkin luottamusvälien pituuteen, mutta ennen kaikkea myös hyvyyteen.

Taulukko 14: Tunnuslukuja takaisinotantaluottamusväleistä tasajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 1000, luottamustaso = 90 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.2678	0.2717	0.6106	0.0320	0.1120	0.7226
ϕ_2	0.8798	0.2441	0.5111	0.0145	-0.6352	-0.1240
θ_1	1.3488	0.2246	0.5749	0.0358	-0.1941	0.3808
θ_2	0.9646	0.2713	0.5660	0.0227	-0.3177	0.2484
θ_3	0.6840	0.1949	0.3753	0.0092	0.3474	0.7227

Nolla osui luottamusvälille huomattavasti useammin kuin 80 % luottamustasolla. Tulokset nollan esiintymiselle luottamusvälillä on esitelty Taulukossa 15. Tätä selettäneen vastaavasti pidentynyt luottamusvälin pituus. Luottamusväli kattoi nollan parametreille θ_1 ja θ_2 lähes joka kerta. Myös kattamistodennäköisyyden kasvua havaittiin myös parametreillä ϕ_1 ja ϕ_2 . Sen sijaan parametrin θ_3 kohdalla luottamusväli

ei kattanut nollaa kertaakaan millään virhejakaumalla.

Taulukko 15: Luottamusväli kattoi nollan (kerrat sadasta). Aikasarjan pituus = 1000, luottamustaso = 90 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	28	34	18
ϕ_2	20	17	14
θ_1	98	97	100
θ_2	98	92	99
θ_3	0	0	0

4.3 Alkuperäisen aikasarjan pituuden varioinnin vaikutus luottamusvälin hyvyyteen

4.3.1 500 elementin aikasarja

Seuraavaksi tarkastellaan, kuinka alkuperäisen tutkittavan aikasarjan lyhentäminen vaikutti luottamusvälien hyvyyteen. Tulokset luottamusvälien hyvyydestä 500 elementin mittaiselle aikasarjalle on esitelty Taulukossa 16.

Taulukko 16: Tutkittujen virhejakaumien tulosten yhteenveto siitä, kuinka monta kertaa sadasta luottamusväli kattoi oikean parametrin. Aikasarjan pituus = 500, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	86	86	81
ϕ_2	74	88	78
θ_1	85	84	88
θ_2	85	90	75
θ_3	80	88	82

Tuloksista voidaan todeta, että pääosin luottamusvälit olivat melko hyviä. Lievää konservatiivisuutta oli havaittavissa. Normaalijakauman ϕ_1 -parametri erottui joukosta, sillä tällä parametrillä luottamusväli kattoi parametrin vain 74 kertaa sadasta. Lisäksi gammajakauman tapauksessa luottamusväli kattoi θ_2 -parametrin jopa 90 kertaa sadasta. Tasajakauman tapauksessa eroja oli molempiin suuntiin - osan parametreista luottamusväli kattoi liian usein ja osan liian harvoin. Näyttäisi hieman siltä, että tuloksissa olisi hajontaa enemmän kuin vertailukohdassa.

Seuraavaksi tarkastellaan avainlukuja normaalijakautuneille virheille. Tulokset on esitelty Taulukossa 17. Tarkasteltaessa pisimpiä välien pituuksia, havaitaan, että 500 elementin aikasarjalla pisimmät välit olivat huomattavasti pidempiä kuin vertailukohdassa. Pisimmät välit olivat kauttaaltaan noin 1.5-1.9 kertaa pidempiä 500 elementin aikasarjalla kuin 1000 elementin aikasarjalla. Lyhyimpien luottamusvälien pituus oli myös pidempi kuin vertailukohdassa. Sama trendi oli havaittavissa myös keskiarvopituudessa. Kaikkien parametrien luottamusvälien keskiarvopituus oli huomattavasti suurempi. Sama luonnollisesti päti pituuden variansseihin. Tarkasteltaessa keskimääräisiä ylä- ja alarajoja luottamusväleille, havaittiin, että luottamusvälin kasvua oli tapahtunut sekä ylös- että alaspäin, mutta eri parametreilla eri tavoin. Kasvu ei ollut symmetristä. Alaraja oli siirtynyt ylärajaa enemmän parametreilla ϕ_1 ja ϕ_2 ja loppuilla vastaavasti toisinpäin.

Myös gammajakautuneiden virhetermien tapauksessa luottamusvälien pituudet olivat suurentuneet verrattuna 1000 elementin aikasarjaan samalla luottamustasolla. Gammajakautuneiden virheiden aineiston avainluvut on esitelty Taulukossa 18. Pituuksien kasvu oli samankaltaista kuin normaalijakautuneilla virheillä, gamma- ja normaalijakauman välillä ei tuloksissa ollut suurta eroa keskenään. Luottamusvälien pituuksien keskiarvoissa oli havaittavissa vain marginaalisia eroja, joita selittänee parhaiten sattuma.

Taulukko 17: Tunnuslukuja takaisinotantaluottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 500, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.4637	0.2395	0.5741	0.0564	0.1105	0.6847
ϕ_2	1.0653	0.1764	0.4569	0.0291	-0.6243	-0.1674
θ_1	1.4577	0.2501	0.5599	0.0585	-0.1715	0.3884
θ_2	1.4762	0.1841	0.5446	0.0537	-0.3084	0.2362
θ_3	0.6998	0.1284	0.3428	0.0124	0.3776	0.7205

Taulukko 18: Tunnuslukuja takaisinotantaluottamusväleistä gammajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 500, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.4904	0.2312	0.5520	0.0546	0.1066	0.6585
ϕ_2	1.0190	0.2433	0.4760	0.0234	-0.6079	-0.1318
θ_1	1.3101	0.1965	0.5374	0.0576	-0.1492	0.3882
θ_2	1.4159	0.2038	0.5565	0.0539	-0.3283	0.2282
θ_3	0.8393	0.1492	0.3224	0.0125	0.3767	0.6991

Taulukossa 19 on esitelty avainluvut tasajakautuneille virheille. Trendi oli jälleen sama kuin gamma- ja normaalijakaumissa - luottamusvälit olivat huomattavasti pidentyneet. Tasajakautuneiden virheidenkään tapauksessa eroa muihin jakaumatyyppisiin ei juurikaan ollut ja erot näihin voitaneen selittää sattumalla. Jakaumatyyppistä riippumatta luottamusvälit pitenevät merkittävästi verrattuna pidempään aikasarjaan.

Taulukko 19: Tunnuslukuja takaisinotantaluottamusväleistä tasajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 500, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	1.3449	0.2948	0.5464	0.0612	0.1225	0.6690
ϕ_2	1.0393	0.2155	0.4383	0.0185	-0.6212	-0.1828
θ_1	1.3942	0.1819	0.5245	0.0660	-0.1510	0.3735
θ_2	1.0671	0.1705	0.5126	0.0461	-0.2878	0.2248
θ_3	0.7719	0.0983	0.3215	0.0135	0.3958	0.7173

Luottamusväli kattoi nollan keskimäärin hieman useammin kuin 1000 elementin aikasarjalla. Tulokset nollan esiintymiselle luottamusvälillä on esitelty Taulukossa 20. Suurin ero osumistodennäköisyydessä oli havaittavissa parametreilla ϕ_1 ja ϕ_2 , joilla osumistodennäköisyys moninkertaistui.

Ajon aikana havaittiin yksi virhe. Virhe saattoi johtua sattumasta tai siitä, että

Taulukko 20: Luottamusväli kattoi nollan (kerrat sadasta). Aikasarjan pituus = 500, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	27	26	25
ϕ_2	20	16	10
θ_1	87	84	83
θ_2	90	83	82
θ_3	2	2	0

lyhyempi luottamusväli johti epätarkkuuteen ja suurempaan hajontaan.

4.3.2 100 elementin aikasarja

Sitten tarkastellaan, miten luottamusvälit käyttäytyivät 100 elementin aikasarjalla. Tämä aikasarja alkaa olla melko lyhyt ja huomattavan lyhyiden otosten osuus takaisinotannassa korostuu. Tulokset 100 elementin aikasarjalle on esitelty Taulukossa 21.

Taulukko 21: Tutkittujen virhejakaumien tulosten yhteenveto siitä, kuinka monta kertaa sadasta luottamusväli kattoi oikean parametrin. Aikasarjan pituus = 100, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	77	81	87
ϕ_2	81	87	85
θ_1	93	92	92
θ_2	89	94	89
θ_3	91	87	89

Tuloksista voidaan havaita se, että pääosin luottamusvälit olivat edelleen melko hyviä luottamusvälin kattaessa oikean ARMA-parametrin likimain oikean määrän. Kuitenkin satunnaisuuden vaikutus alkoi näkyä, koska tuloksissa oli enemmän hajontaa kuin 500 elementin aikasarjalla. Lievää konservatiivisuutta oli tälläkin lyhennetyllä aikasarjalla havaittavissa. Huomionarvoista on se, että θ_1 ja θ_2 -parametreilla luottamusväli kattoi oikea parametrin jopa reilusti yli 90 kertaa sadasta. Syynä tähän lienevät pidentyneet luottamusvälien pituudet.

Seuraavaksi tarkasteltiin avainlukuja 100 elementin aikasarjan tapauksessa. Normaalijakautuneille virhetermeille nämä tulokset on esitelty Taulukossa 22. Havaittiin, että luottamusvälien pituudet olivat kasvaneet valtaviin mittoihin. Jopa keskiarvopituudet olivat kasvaneet niin pitkiksi, että tuloksia ei kunnolla pystynyt enää hyödyntämään luottamusvälien määrittämisessä. Keskiarvopituus ylitti yhden parametrin ϕ_1 , θ_1 ja θ_2 tapauksissa. Sama luottamusvälien räjähtäminen oli näkyvillä luottamusvälien keskiarvoylä- ja alarajoissa.

Taulukko 22: Tunnuslukuja takaisinotantaluottamusväleistä normaalijakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 100, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alajara}$	$ka_{yläraja}$
ϕ_1	2.0662	0.4756	1.1095	0.1174	-0.4061	0.7034
ϕ_2	1.5854	0.2242	0.7532	0.0800	-0.7768	-0.0236
θ_1	2.3093	0.4607	1.2044	0.1458	-0.2926	0.9118
θ_2	2.7682	0.3313	1.2588	0.2443	-0.5621	0.6967
θ_3	1.8363	0.2582	0.8644	0.0925	0.0333	0.8977

Gammajakautuneille virheet vastaavat tulokset on esitelty Taulukossa 23. Tulokset olivat samansuuntaiset kuin normaalijakautuneiden virheiden tapauksessa, eli

luottamusvälien pituus oli kasvanut merkittävästi verrattuna pidempään alkuperäiseen aikasarjaan. Ajon aikana kerätyt pisimmät luottamusvälit olivat jopa hieman pidempiä kuin normaalijakautuneilla virheillä. Keskiarvopituuksissa tulokset sitä vastoin olivat samansuuntaisia. Tässäkään tapauksessa ei ollut mielekästä käyttää tuloksia oikean parametrin määrittämiseen.

Taulukko 23: Tunnuslukuja takaisinotantaluottamusväleistä gammajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 100, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	2.5904	0.4841	1.1048	0.1865	-0.3407	0.7641
ϕ_2	1.3730	0.2902	0.7405	0.0637	-0.7744	-0.0339
θ_1	2.5975	0.4590	1.1939	0.2190	-0.3370	0.8569
θ_2	2.8645	0.4590	1.2008	0.2446	-0.5435	0.6572
θ_3	1.9995	0.2886	0.8666	0.1129	0.0372	0.9038

Tasajakautuneiden virheiden tapauksessa tulokset on esitelty Taulukossa 24. Tulokset olivat edelleen samansuuntaisia kuin muilla virhejakaumilla. Luottamusvälien pituudet olivat huomattavasti pidempiä kuin pidemmällä alkuperäisellä aikasarjalla. Pituudet olivat kasvaneet myös tasajakautuneiden virhetermien tapauksessa samalla tavalla kuin muilla virheillä, joskin pisimpien luottamusvälien pituudet olivat lähempänä normaalijakaumaa kuin gammajakaumaa. Keskiarvopituudet olivat lähestulkoon samat kaikille kolmelle virhejakaumatyypille.

Taulukko 24: Tunnuslukuja takaisinotantaluottamusväleistä tasajakautuneiden virhetermien tapauksessa. Luvut on pyöristetty neljän desimaalin tarkkuuteen. Aikasarjan pituus = 100, luottamustaso = 80 %.

	Pisin pituus	Lyhin pituus	ka_{pituus}	Pituuden var.	$ka_{alaraja}$	$ka_{yläraja}$
ϕ_1	2.4628	0.4189	1.1889	0.1601	-0.3921	0.7968
ϕ_2	1.5393	0.3172	0.7558	0.0802	-0.7767	-0.0209
θ_1	2.7241	0.3525	1.2608	0.2081	-0.3650	0.8958
θ_2	2.5241	0.3997	1.2359	0.2653	-0.5385	0.6973
θ_3	1.9998	0.2542	0.8548	0.1242	0.0371	0.8919

Nolla osui luottamusväleille kaikkien parametrien osalta huomattavasti useammin kuin missään muussa tarkastellussa tapauksessa. Nollan osuminen luottamusvälille on esitelty Taulukossa 25. Räjähäneet luottamusvälien pituudet selittävät tämän. Nolla osui niin usein luottamusvälille, että oikeastaan minkään parametrin osalta ei voida enää hahmottaa tilastollisesti merkitsevää eroa nolasta.

Ajon aikana parametrejä estimoivan funktion suorituksessa havaittiin kolme virhettä. Syynä virheille lienee se, että takaisinotannassa tuli niin lyhyitä aikasarjoja, että niissä ei ollut tarpeeksi informaatiota sarjan estimoimiseksi.

Taulukko 25: Nolla kattoi luottamusvälin (kerrat sadasta). Aikasarjan pituus = 100, luottamustaso = 80 %.

Parametri	Normaalijakauma	Gammajakauma	Tasajakauma
ϕ_1	87	78	83
ϕ_2	47	44	40
θ_1	91	91	92
θ_2	97	98	97
θ_3	46	47	45

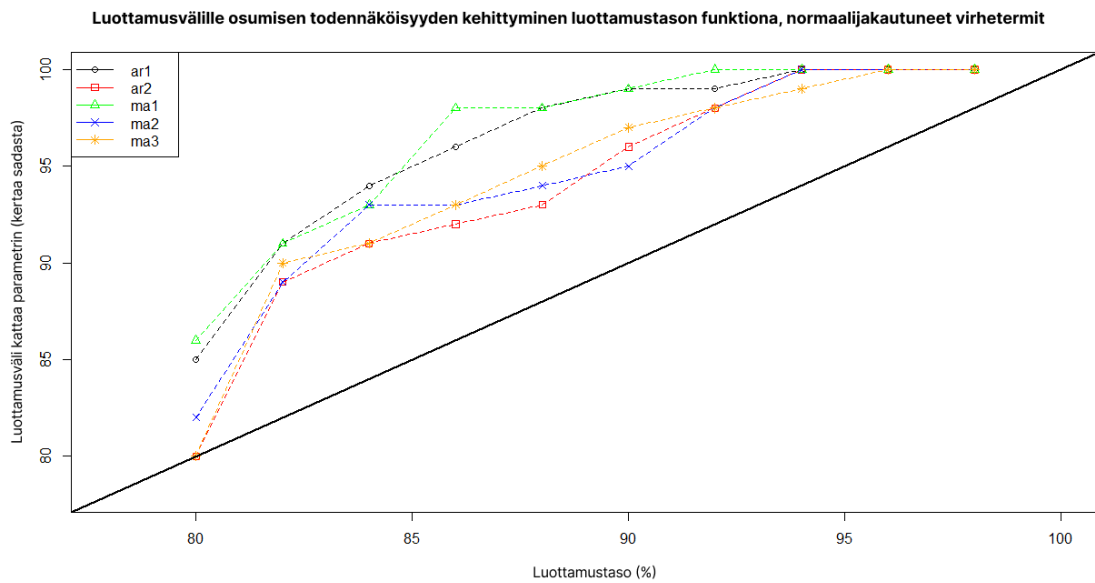
4.4 Luottamusvälille osumisen todennäköisyyden kehittymisen luottamustason funktiona

Tutkimuksen aiemmissa vaiheissa havaittiin, että luottamusvälille osuminen muuttui sitä konservatiivisemmaksi, mitä suurempi luottamustaso oli. Esimerkiksi 50 % luottamustason tapauksessa luottamusväli kattoi oikeat ARMA-parametrit jokaisen tutkitun jakauman tapauksessa vain hieman yli 40 kertaa sadasta, eli tällöin luottamusväli kattoi parametrit hieman harvemmin kuin olisi pitänyt. Vastaavasti 80 % luottamustason tapauksessa luottamusväli kattoi parametrit hieman yli 80 kertaa sadasta jokaisen jakauman tapauksessa. 90 % luottamustason tapauksessa luottamustaso kattoi oikean parametrit jopa merkittävästi useammin kuin olisi pitänyt. Näin ollen voidaan arvioida, että takaisinotanta-algoritmi on parhaimmillaan, eli se antaa oikean osumistodennäköisyyden noin 75 % luottamusvälillä - tällöin väli kattaa oikean parametrin noin 75 kertaa sadasta. Seuraavaksi tutkittiin, miten luottamusvälille osumisen todennäköisyys kehittyi 80-98 % luottamustasoille.

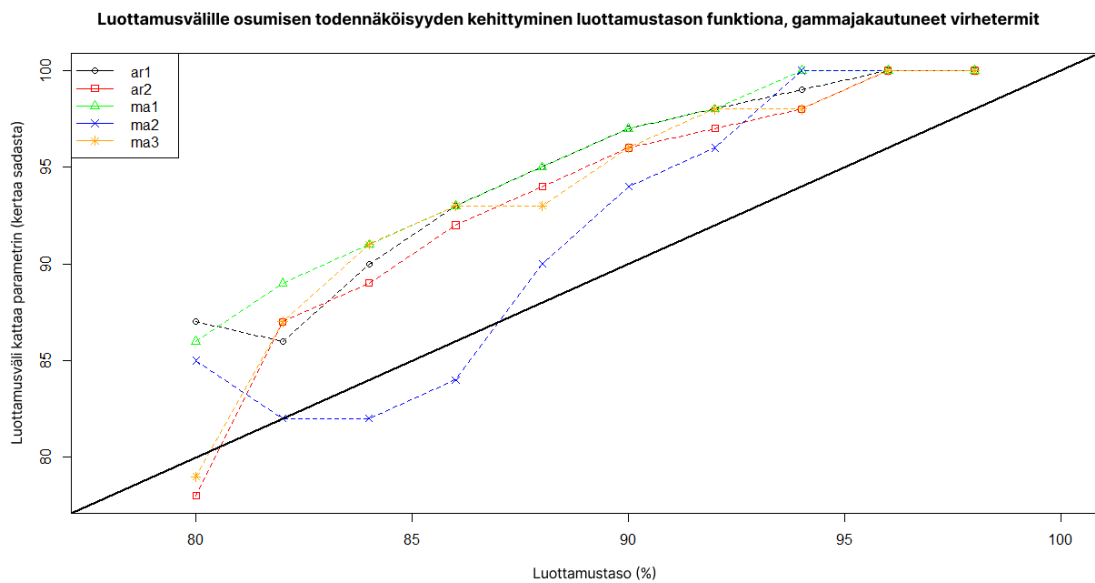
Kuvassa 7 on esitelty luottamusvälille osumisen todennäköisyys luottamustason funktiona normaalijakautuneiden virhetermien tapauksessa. Havaittiin, että parametrien osumistodennäköisyyden kasvu oli aluksi voimakasta, mutta hidastui nopeasti, kun lähestyttiin 100 % luottamustasoa. Näyttäisi siis siltä, että kun luottamustaso on 80 % ja 84 % välillä, todennäköisyys osua välille kasvaa huomattavasti nopeammin kuin luottamustaso, eli kulmakertoimella, joka on suurempi kuin 1. Kun verrataan tätä havaintoa tutkimukseen 50 % luottamustasolla, voidaan arvioida osumistodennäköisyyden kasvavan luottamustasoa nopeammin aina 50 % alkaen. Tätä tukee havainto siitä, että 50 % luottamustason tapauksessa osumistodennäköisyys oli huomattavasti alle 50 % ja 80 % luottamustason tapauksessa niukasti yli 80 %. Kun luottamustaso ylitti 84 %, alkoi osumistodennäköisyyden kasvu hidastua, lähestyen 100 % osumistodennäköisyyttä. Kuvasta 7 voidaan havaita, että 100 % osumistodennäköisyys saavutettiin 96 % luottamustasolla.

Vastaavasti Kuvassa 8 on esitelty luottamusvälille osumisen todennäköisyys luottamustason funktiona gammajakautuneiden virhetermien tapauksessa. Kuvasta voidaan havaita, että kuvaajien muodot olivat pääasiassa hyvin samankaltaisia kuin normaalijakautuneiden virheiden tapauksessa. Kuitenkin θ_2 -parametri erottui joukosta poikkeavalla kuvaajallaan, jossa luottamusvälille osumisen todennäköisyys aluksi jopa pieneni. Todennäköisin selitys tälle on satunnaisuus, mutta sen selvittäminen vaatii lisätutkimuksia. Muiden parametrien kasvu oli aluksi voimakasta ja jälleen hidastui lähestyttäessä 100 % luottamustasoa. Myös gammajakauman tapauksessa voidaan havaita, että 100 % osumistodennäköisyys saavutettiin 96 % luottamustasolla.

Kuvassa 9 on esitelty parametrien luottamusvälille osumisen todennäköisyydet, kun virheet olivat tasajakautuneita. Kuvaajien välillä oli selkeästi enemmän hajontaa kuin normaalijakauman tapauksessa, mutta useimpien parametrien osumistodennäköisyys kasvoi nopeammin kuin luottamustaso. Kasvu oli myös samanmuotoista kuin muiden virhejakaumatyyppien tapauksissa, eli aluksi nopeaa, ja hidastuvaa loppua kohden. ϕ_2 - ja θ_2 -parametrien osumistodennäköisyydet kasvoivat kuitenkin hitaammin kuin muut. Kaikki parametrit kuitenkin saavuttivat 100 % osumistoden-



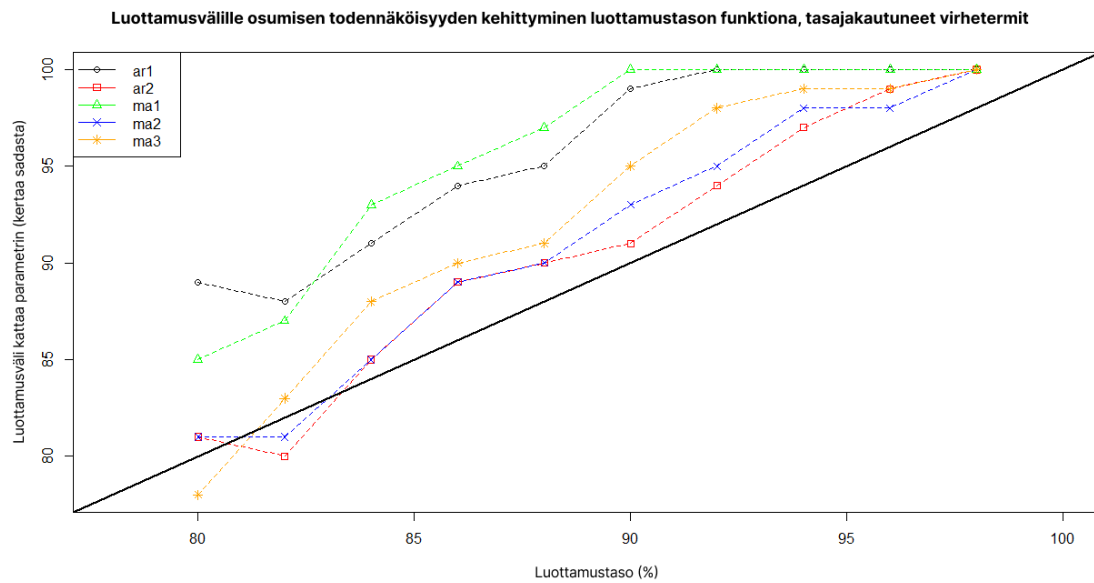
Kuva 7: Kerrat luottamusvälien sisällä eri luottamustasoilla normaalijakautuneilla virhetermeillä



Kuva 8: Kerrat luottamusvälien sisällä eri luottamustasoilla gammajakautuneilla virhetermeillä

näköisyyden 98 % luottamustasolla.

Kaikki tulokset antavat osviittaa siitä, että luottamusväliille osumisen todennäköisyyden kasvu ei ole lineaarista, vaikka valittu luottamustaso kasvaisi lineaarisesti. Lähes kaikkien parametrien osalta kuvaajissa oli havaittavissa konservatiivisuutta 80-98 % luottamustasoilla. Tutkimuksesta voidaan kuitenkin päätellä, että osumisto-



Kuva 9: Kerrat luottamusvälien sisällä eri luottamustasoilla tasajakautuneilla virhetermeillä

dennäköisyyden kasvu on yli 50 % luottamustasoilla nopeampaa kuin luottamustason kasvu ja se hidastuu lähestyttäessä raja-arvoa, eli 100 % luottamustasoa.

5 Yhteenveto

Tutkimuksen tavoitteena oli tutkia ARMA-mallin takaisinotantaluottamusvälien hyvyyttä tarkastelemalla luottamusvälien pituutta ja sitä kuinka monta kertaa luottamusväli kattoi oikean parametrin varioimalla virhejakaumaa, luottamustasoa ja alkuperäisen aikasarjan pituutta. Tuloksena saatiin, että takaisinotanta-algoritmi toimii aikasarjoille silloin, kun takaisinotannassa otokset ovat peräkkäisten havaintojen muodostamia intervaleja ja myös silloin, kun aikasarjan virhetermit tulevat jostakin muusta kuin normaalijakaumasta. Virhejakaumalla ei siis ollut merkittävää vaikutusta tuloksiin. Tämä olikin oletettavissa olemassa olevan tutkimustiedon perusteella. Riippumattomuus aikasarjan virheiden jakaumasta vahvistaa takaisinotannan kuvaa monikäyttöisellä ja sovellusrikkaana algoritmina.

Luottamusvälien hyvydessä havaittiin kuitenkin epälineaarista käyttäytymistä eri luottamustasoilla. Esimerkiksi 50 % luottamustasolla luottamusväli kattoi oikean parametrin harvemmin kuin sen olisi pitänyt, mutta yli 80 % luottamustasolla useammin kuin sen olisi pitänyt. Parhaimmillaan algoritmi oli siis noin 75 % luottamustasolla. Arvioitiin, että tällä luottamustasolla luottamusväli kattoi oikean parametrin juuri noin 75 kertaa sadasta. Yli 90 % luottamustasoilla luottamusvälit olivat huomattavan konservatiivisia, eli pidempiä kuin niiden olisi tarvinnut olla. Kuitenkin tavallisesti tutkimukset keskittyvät korkeille luottamustasoille. Esimerkiksi 50 % luottamustasolla ei juurikaan käytännön sovelluksia ole. Havainnot 50 %, 80% ja 90% luottamustasoilla johtivat lisätutkimuksiin, joista luottamusvälille osumisen todennäköisyyden ja luottamustason välinen epälinearisuus oli vahvistettavissa.

Alkuperäisen aikasarjan pituus näytti lisäävän hajontaa tuloksiin kääntäen verrannollisesti. Mitä lyhyempi aikasarja, sitä enemmän virheitä ja hajontaa tuloksissa oli. Parhaat tulokset saatiin 1000 elementin vertailuaikasarjalla. Kun alkuperäisen aikasarjan pituutta lyhennettiin vain 100 elementin mittaiseksi, luottamusvälien pituudet räjähtivät, eivätkä ne olleet enää käyttökelpoisia.

Tulevaisuudessa lisätutkimuksia voisi suorittaa useammalle todennäköisyysjakamalle. Tällöin riippuvuutta luottamustason ja oikean parametrin osumisen todennäköisyyden välillä voitaisiin tutkia tarkemmin. Takaisinotanta-algoritmin laskennan raskaus vaikutti merkittävästi tulosten saamiseen, koska laskenta tässä tutkimuksessa suoritettiin kannettavalla tietokoneella. Lisätutkimuksia tehdessä tätä voisi helposti keventää tekemällä laskenta pilvessä tai hyödyntämällä tehokkaampia tietokoneita.

Lisätutkimuksia voisi suorittaa myös varioimalla takaisinotanta-algoritmin otoksen kokoa. Nyt otoskoko valittiin satunnaisesti jokaisen takaisinotantakerroksen alussa. Voisi olla mielenkiintoista nähdä, päästäänkö parempiin tuloksiin esimerkiksi pitämällä otoskoko vakiona ja kuinka pitkä otoskoon tulisi olla, jotta tulokset pysyvät luotettavina. Nyt alkuperäisen aikasarjan lyhentäminen johti siihen, että lyhyiden satunnaisotosten määrä kasvoi ja tällöin luottamusvälien pituudet kasvoivat merkittävästi.

Takaisinotanta tiedettiin entuudestaan monikäyttöiseksi algoritmiksi. Se osoitautui tässäkin tutkimuksessa melko luotettavaksi tavaksi estimoida parametrejä ARMA(2,3)-mallille.

Viitteet

- [1] Michael G Anderson, Ning Zhou, John W Pierre, ja Richard W Wies. Bootstrap-based confidence interval estimates for electromechanical modes from multiple output analysis of measured ambient data. *IEEE Transactions on Power Systems*, 20(2):943–950, 2005.
- [2] Patrice Bertail ja Anna E Dudek. Optimal choice of bootstrap block length for periodically correlated time series. 2022.
- [3] Edward Carlstein. The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *The Annals of Statistics*, pages 1171–1179, 1986.
- [4] Sangit Chatterjee. Bootstrapping ARMA models: Some simulations. *IEEE transactions on systems, man, and cybernetics*, 16(2):294–299, 1986.
- [5] David Roxbee Cox ja David Victor Hinkley. *Theoretical statistics*. CRC Press, 1979.
- [6] Thomas J DiCiccio ja Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 11(3):189–228, 1996.
- [7] Bradley Efron. Bootstrap methods: another look at the jackknife. Teoksessa *Breakthroughs in statistics: Methodology and distribution*, pages 569–593. Springer, 1992.
- [8] Peter Hall. On the number of bootstrap simulations required to construct a confidence interval. *The Annals of Statistics*, pages 1453–1462, 1986.
- [9] G Gardner AC Harvey ja GDA Phillips. An algorithm for exact maximum likelihood estimation of autoregressive-moving average models by means of Kalman filtering. *Applied Statistics*, 29(3):311–322, 1980.
- [10] Joel L Horowitz. The bootstrap. Teoksessa *Handbook of econometrics*, volume 5, pages 3159–3228. Elsevier, 2001.
- [11] Rob J Hyndman ja George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018. Viitattu 26.7.2023. Saatavissa: <http://OTexts.com/fpp2>.
- [12] Hans R Kunsch. The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, pages 1217–1241, 1989.
- [13] Regina Y Liu ja Kesar Singh. Using iid bootstrap inference for general non-iid models. *Journal of statistical planning and inference*, 43(1-2):67–75, 1995.
- [14] Regina Y Liu, Kesar Singh, et al. Moving blocks jackknife and bootstrap capture weak dependence. *Exploring the Limits of Bootstrap*, 225:248, 1992.

- [15] Dimitris N Politis ja Joseph P Romano. The stationary bootstrap. *Journal of the American Statistical association*, 89(428):1303–1313, 1994.
- [16] Joseph P Romano. A circular block-resampling procedure for stationary data. *Exploring the Limits of Bootstrap*, 270:263, 1992.
- [17] RC Souza ja AC Neto. A bootstrap simulation study in ARMA (p, q) structures. *Journal of Forecasting*, 15(4):343–353, 1996.
- [18] Lisa Sullivan. Confidence intervals. Viitattu 26.7.2023. Saatavissa: https://sphweb.bumc.bu.edu/otlt/mph-modules/bs/bs704_confidence_intervals/BS704_Confidence_Intervals_print.html.
- [19] Dheeraj Vaidya. Gamma distribution. *WallStreetMojo*. Viitattu 26.7.2023. Saatavissa: <https://www.wallstreetmojo.com/gamma-distribution/>.
- [20] Abdelhak M Zoubir ja Boualem Boashash. The bootstrap and its application in signal processing. *IEEE signal processing magazine*, 15(1):56–76, 1998.

A Käytetty lähdekoodi

Alla esitelty tutkimuksessa käytetty R-kielillä toteutettu ohjelmakoodi. Ohjelmakoodi on saatavilla myös osoitteessa <https://github.com/NiemiJere/kandidaatintyo>

```
# Tällä koodilla tutkitaan ARMA-mallin takaisinotannon
# (bootstrapping) luottamusvälejä. Tutkimuksessa luodaan
# eripituisia aikasarjoja erilaisilla luottamustasoilla eri
# virhetermien jakaumille. Tutkittavat jakaumat ovat
# normaalijakauma, vino gammajakauma sekä tasajakauma.
#
# © Jere Niemi, Aalto-yliopisto

# Asetetaan mallille toistettavuus
set.seed(123)

# Tässä tunnetun ARMA(2,3)-aikasarjan parametrit.
alpha <- c(0.5, -0.3)
beta <- c(0.02, -0.15, 0.5)

# Gammajakauman parametrit
shape <- 2
rate <- 0.5

# Tasajakauman parametrit
maximum <- 5
minimum <- -5

# Luottamusväli
confidence_interval <- 0.9
# confidence_interval <- 0.8
# confidence_interval <- 0.5

# Asetetaan takaisinotantojen määrä
bootstrap_rounds <- 1000

# Valitaan aikasarjan pituus
showcase_ts_length <- 1000
# showcase_ts_length <- 500
# showcase_ts_length <- 100

# Funktio luo aikasarjan annettujen parametrien mukaisesti
create_arma_ts <- function(initial_values, a, b, res, len){
  time_series <- matrix(NA, nrow=len, ncol=1)
  time_series[1, 1] <- initial_values[1]
  time_series[2, 1] <- initial_values[2]
  time_series[3, 1] <- initial_values[3]

  for(i in 4:len){
    time_series[i, 1] <-
      a[1] * time_series[i - 1] + a[2] * time_series[i - 2] +
      b[1] * res[i - 1] + b[2] * res[i - 2] + b[3] * res[i - 3] +
      res[i]
  }
}
```

```

    }
    return(time_series)
  }

# Apufunktio, joka siirtää jakaumaa siten, että jakauman
# keskiarvo on 0.
shift <- function(values, shift_amount){
  result <- values-shift_amount
  return(result)
}

# Funktio luo gammajakauman virhetermit ja siirtää ne siten,
# että palautettavan jakauman keskiarvo on 0.
generate_gamma_distributed_residuals <- function(
  amount,
  shape,
  rate
){
  gamma_random_numbers <- rgamma(amount, shape, rate)
  return(shift(gamma_random_numbers, shape/rate))
}

# ARMA-mallin takaisinotantafunktio
estimate_arma_parameters <- function(rounds, series, ar, ma){
  ts_min_length <- 10
  arma_coef_amount <- ar + ma
  ARMA <- matrix(NA, nrow=rounds, ncol=arma_coef_amount)
  for(i in 1:rounds){
    series_length <- sample(ts_min_length:length(series), 1)
    starting_point <- sample(1:(length(series) - series_length), 1)
    bootstrap_series <- series[
      starting_point:(starting_point + series_length)
    ]
    bootstrap_fit <- arima(
      bootstrap_series,
      method="ML",
      order=c(ar, 0, ma)
    )
    for(j in 1:arma_coef_amount){
      ARMA[i,j] <- bootstrap_fit$coef[j]
    }
  }
  return(ARMA)
}

# Funktio järjestää matriisin alkiot suuruusjärjestykseen
sort_params <- function(series){
  sorted_matrix <- matrix(NA, nrow=nrow(series), ncol=ncol(series))
  for(i in 1:ncol(series)){
    sorted_matrix[,i] <- sort(series[, i])
  }
  return(sorted_matrix)
}

```

```

# Funktio palauttaa halutun luottamusvälin arvot
get_confidence_interval <- function(series, interval){
  low_bound_index <- round(((1 - interval)/2) * length(series)) + 1
  up_bound_index <- round(((1 - ((1 - interval)/2)) * length(series))
  return(c(series[low_bound_index], series[up_bound_index]))
}

# Funktio laskee luottamusvälit kaikille ARMA-mallin parametreille
calculate_bounds <- function(series, confidence_interval){
  bounds_per_param <- matrix(NA, nrow=ncol(series), ncol=2)
  for(i in 1:ncol(series)){
    bounds_per_param[i,] <- get_confidence_interval(
      series[,i],
      confidence_interval
    )
  }
  return(bounds_per_param)
}

# Apufunktio, joka kertoo, onko oikea parametri
# luottamusvälin sisällä
parameter_is_in_bounds <- function(bounds, param){
  return(param >= bounds[1] & param <= bounds[2])
}

calculate_confidence_interval_length <- function(bound){
  return(bound[2] - bound[1])
}

is_zero_in_bound <- function(bound){
  return(bound[2] >= 0 & bound[1] < 0)
}

# Seuraavaa osiota käytetään datan esittelyyn

normally_distributed_residuals <- rnorm(showcase_ts_length)

gamma_distributed_residuals <-
  generate_gamma_distributed_residuals(
    showcase_ts_length,
    shape,
    rate
  )

uniformly_distributed_residuals <- runif(
  showcase_ts_length,
  min=minimum,
  max=maximum
)

data_normal_distribution <- create_arma_ts(
  rnorm(3),
  alpha,
  beta,

```

```

normally_distributed_residuals,
showcase_ts_length
)

data_gamma_distribution <- create_arma_ts(
  generate_gamma_distributed_residuals(3, shape, rate),
  alpha,
  beta,
  gamma_distributed_residuals,
  showcase_ts_length
)

data_uniform_distribution <- create_arma_ts(
  runif(3, min=minimum, max=maximum),
  alpha,
  beta,
  uniformly_distributed_residuals,
  showcase_ts_length
)

plot(
  data_normal_distribution,
  type="l",
  main="ARMA(2,3)-aikasarja┐normaalijakautuneilla┐virhetermeillä",
  ylab = 'Arvo',
  xlab = 'Indeksi'
)

plot(
  data_gamma_distribution,
  type="l",
  main="ARMA(2,3)-aikasarja┐gammajakautuneilla┐virhetermeillä",
  ylab = 'Arvo',
  xlab = 'Indeksi'
)

plot(
  data_uniform_distribution,
  type="l",
  main="ARMA(2,3)-aikasarja┐tasajakautuneilla┐virhetermeillä",
  ylab = 'Arvo',
  xlab = 'Indeksi'
)

hist(
  normally_distributed_residuals,
  main="Normaalijakautuneet┐virhetermit┐(odotusarvo=0,┐var=1)",
  ylab = 'Määrä',
  xlab = 'Arvo'
)

hist(
  gamma_distributed_residuals,
  main="Gammajakautuneet┐virhetermit┐(odotusarvo=0,┐var=8)",

```

```

    ylab = 'Määrä',
    xlab = 'Arvo'
  )

  hist(
    uniformly_distributed_residuals,
    main="Tasajakautuneet virhetermit (odotusarvo=0, var=8.3)",
    ylab = 'Määrä',
    xlab = 'Arvo'
  )

  # Datat esittelyosio päättyy

  # Takaisinotantaiteraatiot alkavat

  # Ajetaan iteraatio 100 kertaa ja katsotaan, monta kertaa kukin
  # parametri on luottamusvälin sisällä
  iteration_rounds <- 100

  times_within_bounds_normal_distribution <- matrix(
    0,
    nrow=length(alpha)+length(beta),
    ncol=1
  )

  times_within_bounds_gamma_distribution <- matrix(
    0,
    nrow=length(alpha)+length(beta),
    ncol=1
  )

  times_within_bounds_uniform_distribution <- matrix(
    0,
    nrow=length(alpha)+length(beta),
    ncol=1
  )

  bounds_normal_distribution <- matrix(
    0,
    nrow=iteration_rounds,
    ncol=2*(length(alpha) + length(beta))
  )

  bounds_gamma_distribution <- matrix(
    0,
    nrow=iteration_rounds,
    ncol=2*(length(alpha) + length(beta))
  )

  bounds_uniform_distribution <- matrix(
    0, nrow=iteration_rounds,
    ncol=2*(length(alpha) + length(beta))
  )

  bound_lengths_normal_distribution <- matrix(
    0,

```



```

    nrow=iteration_rounds,
    ncol=length(alpha) + length(beta)
  )

  bound_lengths_gamma_distribution <- matrix(
    0,
    nrow=iteration_rounds,
    ncol=length(alpha) + length(beta)
  )

  bound_lengths_uniform_distribution <- matrix(
    0,
    nrow=iteration_rounds,
    ncol=length(alpha) + length(beta)
  )

  errors_normal_distribution <- 0
  errors_gamma_distribution <- 0
  errors_uniform_distribution <- 0

  zeros_within_bounds_normal_distribution <- matrix(
    0,
    nrow=length(alpha) + length(beta),
    ncol=1
  )

  zeros_within_bounds_gamma_distribution <- matrix(
    0,
    nrow=length(alpha) + length(beta),
    ncol=1
  )

  zeros_within_bounds_uniform_distribution <- matrix(
    0,
    nrow=length(alpha) + length(beta),
    ncol=1
  )

  original_params <- c(alpha, beta)

  print(
    paste('Script is about to run with conf=', confidence_interval)
  )

  print(
    paste(
      'Script is about to run with ts_length=',
      showcase_ts_length
    )
  )

  for(i in 1:iteration_rounds){

```

```

normally_distributed_residuals <- rnorm(showcase_ts_length)
gamma_distributed_residuals <-
  generate_gamma_distributed_residuals(
    showcase_ts_length, shape, rate
  )
uniformly_distributed_residuals <-
  runif(showcase_ts_length, min=minimum, max=maximum)

data_normal_distribution <- create_arma_ts(
  rnorm(3),
  alpha,
  beta,
  normally_distributed_residuals,
  showcase_ts_length
)
data_gamma_distribution <- create_arma_ts(
  generate_gamma_distributed_residuals(3, shape, rate),
  alpha,
  beta,
  gamma_distributed_residuals,
  showcase_ts_length
)
data_uniform_distribution <- create_arma_ts(
  runif(3, min=minimum, max=maximum),
  alpha, beta,
  uniformly_distributed_residuals,
  showcase_ts_length
)

succeeded <- FALSE
maxTries <- 1000
counter <- 0

# Normaalijakauma
repeat{
  tryCatch({
    counter <- counter + 1
    if (counter == maxTries) {
      print(
        paste(
          'Maximum amount of iterations reached.',
          'No stationary result found.'
        )
      )
    }
  }
  params <- estimate_arma_parameters(
    bootstrap_rounds,
    data_normal_distribution,
    length(alpha),
    length(beta)
  )
  succeeded <- TRUE
}, error=function(e) {

```

```

    errors_normal_distribution <- errors_normal_distribution + 1
    cat("An error occurred:", conditionMessage(e), "\n")
  })
  if(succeeded | counter >= maxTries){
    break
  }
}
sorted_params <- sort_params(params)
calculated_bounds <-
  calculate_bounds(sorted_params, confidence_interval)
for(j in 1:nrow(calculated_bounds)){
  if(
    parameter_is_in_bounds(
      calculated_bounds[j,], original_params[j]
   )){
    times_within_bounds_normal_distribution[j, 1] <-
      times_within_bounds_normal_distribution[j, 1] + 1
  }
  if(is_zero_in_bound(calculated_bounds[j,])){
    zeros_within_bounds_normal_distribution[j, 1] <-
      zeros_within_bounds_normal_distribution[j, 1] + 1
  }
  bound_lengths_normal_distribution[i, j] <-
    calculate_confidence_interval_length(calculated_bounds[j,])
  bounds_normal_distribution[i, 2*j - 1] <-
    calculated_bounds[j, 1]
  bounds_normal_distribution[i, 2*j] <- calculated_bounds[j, 2]
}

counter <- 0

# GammaJakauma
repeat{
  tryCatch({
    counter <- counter + 1
    if (counter == maxTries) {
      print(
        paste(
          'Maximum amount of iterations reached.',
          'No stationary result found.'
        )
      )
    }
  }
  params <- estimate_arma_parameters(
    bootstrap_rounds,
    data_gamma_distribution,
    length(alpha),
    length(beta)
  )
  succeeded <- TRUE
}, error=function(e) {
  cat("An error occurred:", conditionMessage(e), "\n")
  errors_gamma_distribution <- errors_gamma_distribution + 1
})

```

```

    if(succeeded | counter >= maxTries){
      break
    }
  }
  sorted_params <- sort_params(params)
  calculated_bounds <-
    calculate_bounds(sorted_params, confidence_interval)
  for(j in 1:nrow(calculated_bounds)){
    if(parameter_is_in_bounds(
      calculated_bounds[j,], original_params[j]
    )){
      times_within_bounds_gamma_distribution[j, 1] <-
        times_within_bounds_gamma_distribution[j, 1] + 1
    }
    if(is_zero_in_bound(calculated_bounds[j,])){
      zeros_within_bounds_gamma_distribution[j, 1] <-
        zeros_within_bounds_gamma_distribution[j, 1] + 1
    }
    bound_lengths_gamma_distribution[i, j] <-
      calculate_confidence_interval_length(calculated_bounds[j,])
    bounds_gamma_distribution[i, 2*j - 1] <-
      calculated_bounds[j, 1]
    bounds_gamma_distribution[i, 2*j] <- calculated_bounds[j, 2]
  }

  counter <- 0

# Tasajakauma
repeat{
  tryCatch({
    counter <- counter + 1
    if (counter == maxTries) {
      print(
        paste(
          'Maximum amount of iterations reached.',
          'No stationary result found.'
        )
      )
    }
  }
  params <- estimate_arma_parameters(
    bootstrap_rounds,
    data_uniform_distribution,
    length(alpha), length(beta)
  )
  succeeded <- TRUE
}, error=function(e) {
  cat("An error occurred:", conditionMessage(e), "\n")
  errors_uniform_distribution <-
    errors_uniform_distribution + 1
})
  if(succeeded | counter >= maxTries){
    break
  }
}
}

```

```

sorted_params <- sort_params(params)
calculated_bounds <-
  calculate_bounds(sorted_params, confidence_interval)
for(j in 1:nrow(calculated_bounds)){
  if(parameter_is_in_bounds(
    calculated_bounds[j,], original_params[j])
  ){
    times_within_bounds_uniform_distribution[j, 1] <-
      times_within_bounds_uniform_distribution[j, 1] + 1
  }
  if(is_zero_in_bound(calculated_bounds[j,])){
    zeros_within_bounds_uniform_distribution[j, 1] <-
      zeros_within_bounds_uniform_distribution[j, 1] + 1
  }
  bound_lengths_uniform_distribution[i, j] <-
    calculate_confidence_interval_length(calculated_bounds[j,])
  bounds_uniform_distribution[i, 2*j - 1] <-
    calculated_bounds[j, 1]
  bounds_uniform_distribution[i, 2*j] <- calculated_bounds[j, 2]
}

print(paste(round((i/iteration_rounds) * 100), '%_completed'))
}

# Tulosten printtaamista

# Tarkastellaan yleisesti rajojen pituuksia
print(bounds_normal_distribution)
print(bound_lengths_normal_distribution)

print(bounds_gamma_distribution)
print(bound_lengths_gamma_distribution)

print(bounds_uniform_distribution)
print(bound_lengths_uniform_distribution)

# Rajojen avainluvut normaalijakaumalle
for(i in 1:ncol(bound_lengths_normal_distribution)){
  print(paste('index:', i))
  print(max(bound_lengths_normal_distribution[,i]))
  print(min(bound_lengths_normal_distribution[,i]))
  print(mean(bound_lengths_normal_distribution[,i]))
  print(var(bound_lengths_normal_distribution[,i]))
}

# Rajojen avainluvut gammajakaumalle
for(i in 1:ncol(bound_lengths_gamma_distribution)){
  print(paste('index:', i))
  print(max(bound_lengths_gamma_distribution[,i]))
  print(min(bound_lengths_gamma_distribution[,i]))
  print(mean(bound_lengths_gamma_distribution[,i]))
  print(var(bound_lengths_gamma_distribution[,i]))
}

```

```

# Rajojen avainluvut tasajakaumalle
for(i in 1:ncol(bounds_uniform_distribution)){
  print(paste('index:', i))
  print(max(bounds_uniform_distribution[,i]))
  print(min(bounds_uniform_distribution[,i]))
  print(mean(bounds_uniform_distribution[,i]))
  print(var(bounds_uniform_distribution[,i]))
}

# Rajat normaali-jakaumalle
for(i in 1:ncol(bounds_normal_distribution)){
  print(paste('index:', i))
  print(mean(bounds_normal_distribution[,i]))
}

# Rajat gammajakaumalle
for(i in 1:ncol(bounds_gamma_distribution)){
  print(paste('index:', i))
  print(mean(bounds_gamma_distribution[,i]))
}

# Rajat tasajakaumalle
for(i in 1:ncol(bounds_uniform_distribution)){
  print(paste('index:', i))
  print(mean(bounds_uniform_distribution[,i]))
}

# Luottamusväli kattaa nollan
print(zeros_within_bounds_normal_distribution)
print(zeros_within_bounds_gamma_distribution)
print(zeros_within_bounds_uniform_distribution)

# Kerrat rajojen sisällä eri jakaumatyypeille
print(times_within_bounds_normal_distribution)
print(times_within_bounds_gamma_distribution)
print(times_within_bounds_uniform_distribution)

# Kuinka monta virhettä esiintyi
print(errors_normal_distribution)
print(errors_gamma_distribution)
print(errors_uniform_distribution)

# Lisätutkimus: luottamusvälille osumisen todennäköisyys
# luottamustason funktiona

intervals <- c(80, 82, 84, 86, 88, 90, 92, 94, 96, 98)

n_mat <- matrix(0, nrow=5, ncol=10)

# Tulokset kaikille eri jakaumatyypeille (normaali-, gamma- ja
# tasajakauma), kommentoi haluttu palanen näkyviin

'n_mat[1,] <- c(85, 91, 94, 96, 98, 99, 99, 100, 100, 100)

```

```

n_mat[2,] <- c(80, 89, 91, 92, 93, 96, 98, 100, 100, 100)
n_mat[3,] <- c(86, 91, 93, 98, 98, 99, 100, 100, 100, 100)
n_mat[4,] <- c(82, 89, 93, 93, 94, 95, 98, 100, 100, 100)
n_mat[5,] <- c(80, 90, 91, 93, 95, 97, 98, 99, 100, 100)

n_mat[1,] <- c(87, 86, 90, 93, 95, 97, 98, 99, 100, 100)
n_mat[2,] <- c(78, 87, 89, 92, 94, 96, 97, 98, 100, 100)
n_mat[3,] <- c(86, 89, 91, 93, 95, 97, 98, 100, 100, 100)
n_mat[4,] <- c(85, 82, 82, 84, 90, 94, 96, 100, 100, 100)
n_mat[5,] <- c(79, 87, 91, 93, 93, 96, 98, 98, 100, 100)

n_mat[1,] <- c(89, 88, 91, 94, 95, 99, 100, 100, 100, 100)
n_mat[2,] <- c(81, 80, 85, 89, 90, 91, 94, 97, 99, 100)
n_mat[3,] <- c(85, 87, 93, 95, 97, 100, 100, 100, 100, 100)
n_mat[4,] <- c(81, 81, 85, 89, 90, 93, 95, 98, 98, 100)
n_mat[5,] <- c(78, 83, 88, 90, 91, 95, 98, 99, 99, 100)

main_str <- paste(
  "Luottamusvälille osumisen todennäköisyyden kehittyminen",
  "luottamustason funktiona, [jakaumatyyppi] virhetermit"
)

plot(
  intervals,
  n_mat[1,],
  xlim = c(78, 100),
  ylim = c(78, 100),
  xlab = "Luottamustaso (%)",
  ylab = "Luottamusväli kattaa parametrin (kertausadasta)",
  main = main_str
)

points(intervals, n_mat[1,], col="black")
lines(intervals, n_mat[1,], col="black", lty=2)

points(intervals, n_mat[2,], col="red", pch=0)
lines(intervals, n_mat[2,], col="red", lty=2)

points(intervals, n_mat[3,], col="green", pch=2)
lines(intervals, n_mat[3,], col="green", lty=2)

points(intervals, n_mat[4,], col="blue", pch=4)
lines(intervals, n_mat[4,], col="blue", lty=2)

points(intervals, n_mat[5,], col="orange", pch=8)
lines(intervals, n_mat[5,], col="orange", lty=2)

abline(coef = c(0, 1), lwd = 2)

legend(
  "topleft",
  legend=c("ar1", "ar2", "ma1", "ma2", "ma3"),
  col=c("black", "red", "green", "blue", "orange"),
  pch=c(1, 0, 2, 4, 8),

```

```
lty=c(1,1,1,1,1),  
ncol=1  
)
```