



Kauno technologijos universitetas
Matematikos ir gamtos mokslų fakultetas

**Investicinio portfelio vertės prognozavimas Markovo
grandinių Monte Karlo metodu**

Baigiamasis magistro projektas

Lukas Lukašinskas
Projekto autorius

Lekt. dr. Mantas Landauskas
Vadovas

Kaunas, 2018



Kauno technologijos universitetas

Matematikos ir gamtos fakultetas

Investicinio portfelio vertės prognozavimas Markovo grandinių Monte Karlo metodu

Baigiamasis magistro projektas

Taikomoji matematika (621G10003)

Lukas Lukašinskas
Projekto autorius

Lekt. dr. Mantas Landauskas
Vadovas

Doc. dr. Audrius Kabašinskas
Recenzentas

Kaunas, 2018



Kauno technologijos universitetas
Matematikos ir gamtos mokslų fakultetas
Lukas Lukašinskas

Investicinio portfelio vertės prognozavimas Markovo grandinių Monte Karlo metodu

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Luko Lukašinsko, baigiamasis projektas tema „Investicinio portfelio vertės prognozavimas Markovo grandinių Monte Karlo metodu“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Turinys

Įvadas	10
1. Literatūros apžvalga	11
1.1. Efektyvios rinkos hipotezė.....	11
1.2. Fundamentalioji ir techninė analizės.....	12
1.3. Markovo grandinių Monte Karlo metodai	13
1.4. Markovo grandinių Monte Karlo algoritmai.....	13
1.4.1. Siūlomo skirstinio parinkimas ir pradinė reikšmė	15
1.4.2. Markovo grandinių Monte Karlo metodų konvergavimas ir diagnostika	16
1.4.3. Markovo grandinių Monte Karlo metodų taikymai.....	18
1.5. Literatūros apibendrinimas.....	19
2. Medžiagos ir tyrimo metodai	20
2.1. Programinė įranga	20
2.2. Ortogonalinių eilučių įvertis.....	20
2.2.1. Čebyševio polinomiali	21
2.3. Monte Karlo metodas	22
2.4. Metropolio-Hastingso algoritmas.....	22
2.5. MCMC konvergavimas	23
2.5.1. Geweke metodas	23
2.5.2. Gelmano ir Rubino metodas	24
2.5.3. Heidelbergerio ir Welcho metodas	24
2.6. Skirstinio aproksimavimas Čebyševio polinomais	25
2.7. Grandinės generavimas ir konvergavimas	26
2.8. Akcijų vertės prognozavimo modelis Markovo grandinių Monte Karlo metodu	27
2.9. Monte Karlo modelis akcijų kainoms prognozuoti.....	27
2.10. Istoriniai duomenys investicinio portfelio sudarymui.....	28
2.11. Prognozavimo tikslumo matai.....	29
3. Tyrimų rezultatai ir jų aptarimas	30
3.1. Aproksimacijos eilės nustatymas	30
3.2. Markovo grandinės konvergavimo tyrimas	33
3.2.1. Grafinis konvergavimo nustatymas bei skaitinės charakteristikos.....	33
3.2.2. Konvergavimo diagnostikos	37
3.2.3. Konvergavimas panaudojant tikslo funkciją	39

3.3. Prognozavimas Markovo grandinių Monte Karlo metodu.....	41
3.4. Prognozavimas Monte Karlo metodu.....	43
3.5. Investicinio portfelio sudarymas	45
Išvados	49
Diskusija	51
Literatūros sąrašas	53
Pranešimai konferencijose	56
Priedai.....	57

Paveikslų sąrašas

1 pav. 2017 metų akcijų kainų dinamika: Microsoft Cooperation (a); Barclays Bank PLC (b).....	28
2 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų gražų (kairėje) ir transformuotų į intervalą $[-1; 1]$ (dešinėje) gražų histogramos	30
3 pav. Microsoft Corporation (viršuje) ir Barclays Bank PLC (apačioje) akcijų tikslo funkcijos TK proklausomybė nuo aproksimacijos eilės K , kai pasirenkami svoriai yra $\alpha = (0.01; 6; 10)$, $\beta = (1; 1; 1)$	31
4 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų gražų histogramos ir aproksimacijos Čebyševio polinomais palyginimas, kai parametrai atitinkamai lygūs $K = 33$, $K = 25$	33
5 pav. Microsoft Corporation slenkančio vidurkio ir autokorelianės funkcijos grafikai, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f).....	34
6 pav. Barclays Bank PLC slenkančio vidurkio ir autokorelianės funkcijos grafikai, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f).....	35
7 pav. Microsoft Corporation (kairėje) ir Barclays Bank PLC (dešinėje) iteracijų ir gautųjų reikšmių grafikas, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f).....	36
8 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų gražų konvergavimo tyrimas pasitelkiant Gelmano ir Rubino diagnostika	37
9 pav. Microsoft Corporation ir Barclays Bank PLC akcijų gražų tikslo funkcijų reikšmių grafikas pritaikius slenkantį vidurkį, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f)	40
10 pav. Microsoft Corporation (viršuje) ir Barclays Bank PLC (apačioje) akcijų gražų histograma ir jų kainų prognozė MCMC metodu.....	41
11 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų prognozavimo metrikų priklausomybė nuo trajektorijų skaičiaus	42
12 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų kainų prognozė Monte Karlo modeliu	44
13 pav. Klasikinio Monte Karlo metodo Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų prognozavimo tikslumų matų priklausomybė nuo generuojamų trajektorijų skaičiaus	45
14 pav. Investicinio portfelio istorinė ir prognozuojamos vertės	46
15 pav. Investicinio portfelio vertės tikslumo priklausomybės nuo prognozuojamo laikotarpio	46
16 pav. Investicinio portfelio vertės tikslumo matų priklausomybė nuo trajektorijų skaičiaus	47
17 pav. Modelių pasikliautinųjų intervalų pločio priklausomybė nuo prognozuojamo laikotarpio..	48

Lentelių sąrašas

1 lentelė. Microsoft Corporation (MC) ir Barclays Bank PLC (BB) akcijų optimalių aproksimacijų eilės.....	32
2 lentelė. Markovo grandinių efektyvūs imties dydžiai bei atmetimo santykiai.....	37
3 lentelė. Geweke diagnostikos statistikos reikšmės.....	38
4 lentelė. Heidelbergerio ir Welcho konvergavimo diagnostikos rezultatai.....	39
5 lentelė. Microsoft Corporation ir Barclays Bank PLC akcijų gražų normalumo testai.....	43
6 lentelė. Microsoft Corporation ir Barclays Bank PLC akcijų gražų normalumo testai duomenims nuo 2017m. lapkričio 9 d. iki iki 2017m. gruodžio 29 d.	43

Lukašinskas, Lukas. Investicinio portfelio vertės prognozavimas Markovo grandinių Monte Karlo metodu. Magistro baigiamasis projektas / vadovas lekt. Mantas Landauskas; Kauno technologijos universitetas, Matematikos ir gamtos mokslų fakultetas.

Studijų kryptis ir sritis (studijų krypties grupė): fiziniai mokslai, matematika.

Reikšminiai žodžiai: Markovo grandinių Monte Karlo metodas, Metropolio-Hastingso algoritmas, ortogonalinių eilučių tankio įvertis, techninė analizė, akcijų grąžų prognozavimas.

Kaunas, 2018. 73 p.

Santrauka

Akcijų rinka yra žinoma kaip visada besikeičianti ir nepastovi, rizikinga ir priklausanti nuo įvairių veiksnių. Investicinio portfelio, sudaryto iš akcijų, vertės prognozavimas gali tapti iššūkiu dėl rinkos permainų, nes vieno teisingo modelio akcijų vertei prognozuoti nėra.

Šiame darbe turimą investicinį portfelį sudaro penki vienetai Microsoft Corporation bei penki vienetai Barclays Bank PLC įmonių akcijų. Investicinio portfelio vertė ir ateities akcijų kainos yra prognozuojamos 18 dienų atliekant techninę analizę – panaudojant buvusias istorines akcijų kainas. Istorinės akcijų grąžos yra aproksimuojamos ortogonalinių eilučių tankio įvertis, o atsitiktinėms grąžoms gauti yra panaudojamas vienas plačiausiai taikomų Markovo grandinių Monte Karlo metodų – nepriklausomas Metropolio-Hastingso algoritmas. Be to akcijų kainoms gauti yra panaudojamas ir klasikinis Monte Karlo modelis. Ortogonalinių eilučių tankio įvertinimo tyrimui, nepriklausomo Metropolio-Hastingso algoritmo realizacijai bei klasikinio Monte Karlo metodo akcijų kainai prognozuoti yra pasitelkiama atvirojo kodo RStudio programinė įranga, o gautosios akcijų kainos yra palyginamos su buvusiomis istorinėmis kainomis.

Lukašinskas, Lukas. Forecasting the value of investment portfolio by Markov chain Monte Carlo method. Master's Final Degree Project / supervisor lect. Mantas Landauskas; The Faculty of Mathematics and Natural sciences, Kaunas University of Technology.

Study field and area (study field group): Physical sciences, Mathematics

Keywords: Markov Chain Monte Karlo method, Metropolis-Hastings algorithm, orthogonal series density estimate, technical analysis, forecasting of stock returns.

Kaunas, 2018. 73 pages.

Summary

Stock market is known as always changing, unstable, risky and dependent on variety of factors. Forecasting the value of an investment portfolio can become a challenge due to market changes. As a result, there is no correct model for predicting the share prices.

In this work, investment portfolio consists of five units of Microsoft Corporation and five units of Barclays Bank PLC shares. The value of the investment portfolio and future stock prices are forecasted for 18 days by performing technical analysis and using historical stock prices. Historical stock returns are approximated by orthogonal series density estimator and the independent Metropolis-Hastings algorithm, one of the most prominent Markov Chain Monte Carlo methods, is used to generate random stock returns. In addition, classical Monte Carlo model is also used for prediction of stock values. Orthogonal series density estimate, independent Metropolis-Hastings algorithm and classical Monte Carlo method is implemented in the open source RStudio software and share values computed are compared to historical prices.

Ivadas

Akcijų rinka yra svarbi ekonomikos dalis, vaidinanti svarbų vaidmenį šalies pramonės ir prekybos augimui, todėl akcijų kaita yra teigiamos ir neigiamos ekonominės veiklos rodiklis. Tai yra pagrindinė priežastis, dėl kurios vyriausybė, pramonė ir netgi centriniai šalies bankai atidžiai stebi rinkos įvykius. Akcijų rinka yra svarbi ne tik pramonės ar ekonomikos, bet ir investiciniu požiūriu, prekiaujant akcijomis, prognozuojant investicinio portfelio vertę ar investuojant į įmonės veiklą.

Vienas iš būdų įvertinti ir prognozuoti akcijų kainas yra techninė analizė, neatsižvelgianti į ekonomikos ar rinkos rodiklius ir kuriai atlikti yra panaudojamos istorinės praeities kainos ir sandorių apimtys. Šio darbo tikslas – panaudoti istorines akcijų kainas ir pasirinktą MCMC metodą suprognozuoti investicinio portfelio vertę.

MCMC algoritmai – tai populiarūs algoritmų klasė šiandieninėje statistikoje, suteikianti galimybę išgauti stebėjimus iš sudėtingų tikimybių skirstinių. Metodo veikimo metu yra sugeneruojama atsitiktinė imtis iš pasirinkto skirstinio. Panaudojant šiuos metodus galima sugeneruoti atsitiktines akcijų grąžas ir prognozuoti akcijų vertes, todėl darbo metu buvo išskelti šie uždaviniai:

- aproksimuoti investicinio portfelio akcijų grąžas ortogonalio eilučių aproksimacija;
- Rstudio programiniame pakete realizuoti pasirinktą MCMC metodą ir sugeneruoti atsitiktines akcijų grąžų sekas;
- ištirti gautųjų sekų konvergavimą ir pasirinktą atsitiktinių skaičių seką pritaikyti akcijų vertėms prognozuoti;
- panaudoti klasikinį Monte Karlo metodą akcijų vertėms gauti ir palyginti abiejų modelių gautąsias kainas su istorinėmis.

1. Literatūros apžvalga

1.1. Efektyvios rinkos hipotezė

Atsižvelgiant į sudėtingą vertybinių popierių rinką, akcijų kainos prognozavimas laikomas vienu iš sudėtingiausių užduočių atliekant finansines prognozes, kadangi jų vertė priklauso nuo įvairių ekonominių, politinių bei socialinių veiksnių. Daugelio investuotojų noras yra laikytis bet kokio prognozavimo metodo, kuris užtikrintų lengvą pelną ir sumažintų investavimo riziką, esančią vertybinių popierių rinkoje. Tikslios nepastovumo prognozės yra būtinos ir svarbios finansinių priemonių kainų nustatyme siekiant suformuoti veiksmingas draudimosi strategijas.

Viena investavimo teorijų, laikomų šiuolaikinės finansinės teorijos kertiniu akmeniu, paaiškinanti vertybinių popierių rinkos elgseną ir susijusi su galimybėmis prognozuoti akcijų rinkas yra efektyvios rinkos hipotezė. Ši investavimo teorija teigia, kad nauja informacija, lengvai įtraukta į saugumo kainas ir rinkos veiklą ar istorinių ir esamų duomenų analizę, negali padėti investuotojams prognozuoti ateitį arba uždirbti didesnę nei vidutinį pagal riziką pelną ar nugalėti rinką. Remiantis šia hipoteze, vienintelis būdas gauti didesnę grąžą yra tik tuo atveju, jeigu įsigyjamos rizikingesnės investicijos. Paprastai efektyvios rinkos hipotezė yra skirstoma į tris kategorijas, priklausomai nuo to, kokia informacija yra svarstoma [1]:

1. **Silpnos formos efektyvios rinkos hipotezė:** ši hipotezė teigia, kad dabartines akcijų vertes visiškai atspindi ankstesnių akcijų verčių seka praeityje, todėl istorinių akcijų kainų analizė neturi prasmės. Remiantis istorinių kainų informacija, negali būti uždirbamas didelės rizikos pelnas, t. y., istorinėse kainose negali būti prognozuojamo modelio, o grąžos normos nepriklauso viena nuo kitos. Atsitiktinės grąžos normos bei nenuspėjamos kainų elgsenos nepriklausomumas yra veiksniai, rodantys silpnos formos rinkos efektyvumą. Ši hipotezė yra siejama su atsitiktinio klaidžiojimo hipoteze, nes ji teigia, kad akcijų kainos kinta atsitiktiniu būdu, todėl neįmanoma prognozuoti būsimų kainų pokyčių.
2. **Pusiau stiprios formos efektyvios rinkos hipotezė:** tokia hipotezės forma teigia, kad kainos visiškai atitinka visą viešai prieinamą informaciją. Daroma prielaida, kad viešai prieinama informacija yra apskaitoma akcijų kainomis, o techninė ar fundamentalioji analizė, kuri remiasi viešai prieinama informacija, negali būti naudojama prognozuojant būsimus kainų svyravimus arba nustatant neskaidrias akcijas. Reikia atkreipti dėmesį, kad pusiau stipri rinkos efektyvumo forma apima ir silpnos formos efektyvios rinkos hipotezę.
3. **Stiprios formos efektyvios rinkos hipotezė:** čia įtraukiama ne tik viešoji, visiems pasiekiamą, bet ir privačioji informacija, kurią paprastai turi įmonių ar korporacijų pareigūnai. Tokia rinkos efektyvumo forma teigia, kad ši informacija atsispindi akcijų kainose. Tai

reiškia, kad jei investuotojas turi vidinę informaciją (pvz., informaciją apie būsimųjų akcijų dydį) ir gali ją taikyti prekybai, jis negalės uždirbti didesnio pelno, nes visa informacija jau ir taip atsispindi akcijų kainose. Šios formos hipotezė apima ir prieš tai dvi minėtas formas.

Nors efektyvios rinkos hipotezė yra šiuolaikinės finansinės teorijos kertinis akmuo ir pagrindas, ji yra prieštaringa, dažnai ginčytina ir kritikuojama pasitelkus kitu, priešingu požiūriu, kad ateities sandorių kainos yra bent iš dalies nuspėjamos remiantis ankstesnių akcijų kainų tendencijomis [2]. Toks priešingas požiūris akcijų kainų prognozavimui tirti yra taikomas atliekant fundamentalią arba techninę analizę.

1.2. Fundamentalioji ir techninė analizė

Fundamentalioji analizė – tai metodas, kuris bando nustatyti saugią vertę, sutelkiant dėmesį į pagrindinius veiksnius, darančius įtaką įmonės faktiniam verslui ir jo ateities perspektyvoms [3]. Šis terminas tiesiog reiškia, kad čia pasitelkiama ne tik rinkos, bet ir finansinio subjekto ekonominiai duomenys ar veiklos rodikliai, todėl šios analizės uždavinys yra nustatyti tikrosios akcijos vertę. Atliekant fundamentalią analizę, atsižvelgiama į bendrą įmonės veiklą ir jos finansines ataskaitas, įskaitant visas naujausias žinias apie bendrovę, taip pat įvertinami įmonės pardavimai, įmonės praeities pajamos, numatomi dividendai, tikėtinos palūkanų normos ir rizikos vertinimas bei prognozuojamos įvairios ilgalaikės rinkos tendencijos arba atliekama lyginamoji analizė pagal metus ar tam tikrus pramonės standartus. Akcijų kainos neišvengiamai priklauso nuo visų šių skaičių, o tik atlikus jų analizę galima suprasti, kaip kainos reaguoja į tam tikrus bendrovės finansinių rezultatų pokyčius.

Pagrindiniai fundamentaliosios analizės privalumai yra tie, jog atliekant šią analizę galima išsiaiškinti ar įmonė yra finansiškai patikima, užtikrinanti stabilų finansinį pajamų augimą, tačiau toks procesas yra gana ilgas bei sudėtingas [3]. Be to, net jei akcijos, lygint su kitomis panašia veikla užsiimančiomis įmonėmis, atrodo finansiškai sveikos ir pigios, tai nereiškia, kad vertė visuomet didės, nes akcijų kainą taip pat lemia jų pasiūla ir paklausa.

Techninė analizė yra vertybinių popierių įvertinimo metodas analizuojant rinkos aktyvumą, praeities kainas ir sandorių apimtį. Kitaip tariant, techninė analizė grindžiama prielaida, kad visi svarbūs pagrindiniai veiksniai jau atsispindi akcijų kainoje, todėl iš esmės ji apibrėžiama kaip istorinių akcijų tendencijų naudojimas ir interpretavimas, siekiant prognozuoti būsimą kainą. Skirtingai nei fundamentalioji analizė, bandanti įvertinti saugumo vertę, šios analizės metu dėmesys sutelkiamas į kainų kitimo diagramas, grafikus ir įvairias analitines priemones.

Pagrindinis techninės analizės privalumas yra jos paprasta logika ir taikymas, nes čia neatsižvelgiama į visus ekonominius, rinkos, technologinius ir kitus veiksnius, galinčius daryti įtaką bendrovei ir pramonei, o sutelkiamas dėmesys į duomenis apie kainas bei parduodamą kiekį ateities kainoms įvertinti. Techninėje analizėje laikoma, kad kainos turi tam tikrų judėjimo modelių, jos nėra visiškai atsitiktinės ir istorija gali pasikartoti, tačiau tokios analizės trūkumas yra tas, kad ji neturi akademinio ar mokslinio patvirtinimo ir jos negalima patvirtinti jokiais racionaliais argumentais.

Analizuojant akcijas fundamentalioje analizėje paprastai yra vertinami ilgesni laikotarpiai nei techninėje. Naudojant fundamentalią analizę vertinami kelerių metų praeities duomenys, o investuojant į pasirinktos įmonės akcijas tikimasi, jog akcijų vertė didės, net jei tai gali užtrukti kelerius metus. Tuo tarpu techninė analizė naudoja santykinai trumpus laikotarpius (pvz.: keletas dienų, savaitių ar mėnesių) ir čia bandoma surasti akcijas, kurias būtų galima parduoti per trumpą laiką už didesnę kainą [3].

1.3. Markovo grandinių Monte Karlo metodai

Markovo grandinių Monte Karlo (angl. Markov Chain Monte Karlo (MCMC)) metodai yra algoritmų klasė, skirta modeliuoti autokoreliuotiems duomenims iš tikimybinių skirstinių, todėl šiandieninis spartus vystymasis kompiuterių moksle ir MCMC programinėje įrangoje suteikia galimybę sumodeliuoti sudėtingiausius modelius. Šie metodai leidžia generuoti priklausomą atsitiktinę seką iš sudėtingų stochastinių modelių. Gautoji seka gali būti panaudota apibendrinant pasiskirstymą (pvz., generuojant histogramą) arba apskaičiuojant integralą (pvz., tikėtiną vertę).

Šie algoritmai dažniausiai naudojami stebėjimų generavimui iš daugiamačių skirstinių, ypač kai jų matmenų skaičius yra didelis. Jų vykdymas yra skaičiuojamojo pobūdžio, todėl reikalingi ištekliai yra tiesiogiai susiję su problemos dimensija (daugiamatiškumu). MCMC suteikia galimybę įvertinti dominantį skirstinį bei nėra apriojamas taškinių įverčių ar asimptoninių standartinių paklaidų, todėl algoritmas gali būti taikomas bet kokiam tikimybiniam modeliui.

1.4. Markovo grandinių Monte Karlo algoritmai

Atsitiktinėms sekoms gauti yra įvairių MCMC algoritmų. Vienas pirmųjų, o vėliau apibendrintų, yra Metropolisio-Hastingso algoritmas [4]. Algoritmo metu yra atsitiktinai pasirinktas taškas kandidatas Y iš siūlomo skirstinio $q(\cdot | X_t)$ ir priimamas su tikimybe $\alpha(X_t, Y)$:

$$\alpha(X_t, Y) = \min \left(1, \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)} \right),$$

čia $\pi(\cdot)$ – tikslinis skirstinys. Atskiras šio algoritmo atvejis yra Metropolio algoritmas, nagrinėjantis tik simetrinius siūlomus skirstinius (pvz., daugiamatis normalusis skirstinys), t. y. kai $q(X|Y) = q(Y|X)$ su visais X bei Y [4]. Šiuo atveju taško kandidato Y priėmimo tikimybė $\alpha(X, Y)$ yra apibrėžiama tokiu būdu:

$$\alpha(X, Y) = \min\left(1, \frac{\pi(Y)}{\pi(X)}\right).$$

Specialusis Metropolio algoritmo atvejis yra atsitiktinio klaidžiojimo Metropolio algoritmas [5], kuriame siūlomas skirstinys yra $q(Y|X) = q(|X - Y|)$. Renkantis siūlomą skirstinį šiame algoritme, gali tekti atidžiai pasirinkti mastelį. Atsargus siūlomas skirstinys generuojantis mažus žingsnius $Y - X_t$ paprastai turės didelę priėmimo tikimybę $\alpha(X_t, Y)$, tačiau konverguos lėtai, o generuojant didelius žingsnius dažnai bus judama link skirstinio uodegų, tokiu būdu suteikiant mažas $\pi(Y)/\pi(X)$ reikšmes ir mažą priėmimo tikimybę $\alpha(X, Y)$. Už šį MCMC metodą geriau konverguojantis yra koreguotas Metropolio Langevino algoritmas [6].

Metropolio-Hastingso algoritmo atvejis, kuomet siūlomas skirstinys yra $q(Y|X) = q(Y)$ vadinamas nepriklausomu Metropolio-Hastingso algoritmu [7]. Pažymėjus $w(X) = \pi(X)/q(X)$, priėmimo (įtraukimo į sugeneruotų atsitiktinių dydžių aibę) tikimybė gali būti užrašyta taip:

$$\alpha(X, Y) = \min\left(1, \frac{w(Y)}{w(X)}\right).$$

Apskritai, nepriklausomas Metropolio-Hastingso algoritmas gali veikti labai gerai arba labai blogai. Siekiant, kad šis atrankos būdas veiktų gerai, $q(\cdot)$ turėtų būti gera $\pi(\cdot)$ aproksimacija, arba, dar geriau, jei $q(\cdot)$ turi sunkesnes uodegas nei $\pi(\cdot)$ [8]. Tarkime, kad $q(\cdot)$ turi lengvesnę uodegą, o X priklauso $\pi(\cdot)$ uodegai. Dauguma kandidatų nebus uodegose, nes $w(X)$ reikšmė bus žymiai didesnė už $w(Y)$. Taip sumažinama taško kandidato priėmimo tikimybė $\alpha(X, Y)$. Sunkių uodegų siūlomi skirstiniai padeda išvengti ilgo periodo užstrigimų uodegose dėl padidėjusio bendro atmetimo kandidatų skaičiaus.

Originalus Metropolio-Hastingso algoritmas atnaujina visą atsitiktinių dydžių vektorių, taikydamas vieną priėmimo arba atmetimo kriterijų. Visų atsitiktinių dydžių atrinkimas vienu metu gali būti problemiškas, kai yra daugiau parametrų. Tokiais atvejais patogiau kiekvieną elementą atnaujinti atskirai, tai vadinama vieno komponento Metropolio-Hastingso algoritmu [9]. Nors tam tikrose situacijose vienos komponentės atranka gali pagreitinti Markovo grandinės konvergavimą, tačiau reikėtų pažymėti, kad tai yra sudėtingesnis skaičiavimo metodas, o skaičiavimo laikas tampa problema.

Specialusis vieno komponento Metropolio-Hastingso algoritmo atvejis yra Gibbso atranka [10], dažniausiai naudojama Bajeso statistikoje. Šio algoritmo vykdymo metu visi galimi stebėjimai yra priimami. Algoritmas reikalauja visiškų sąlyginių tikimybių skirstinių žinojimo nurodant visus parametrus. Tai gali būti problematiška, kuomet sąlyginiai skirstiniai nėra standartiniai [11].

Kadangi MCMC metodų metu svarbu nustatyti teisingą siūlomą skirstinį, adaptuojamų MCMC algoritmų pagrindinė idėja yra atnaujinti siūlomą skirstinį su žiniomis, turimomis apie tikslinį skirstinį [12, 13, 14]. Algoritmas automatizuoja ir tobulina šį procesą pritaikant MCMC metodus – jų vykdymo metu yra bandoma įvertinti geriausius modelio parametrus. Pagrindinis šio metodo privalumas yra tas, kad jį panaudojant yra išvengiama komplikuoto siūlomo skirstinio nustatymo. Nors šis algoritmas yra lengvai naudojamas ir greitas, tačiau grandinė nėra visiškai ergodinė.

1.4.1. Siūlomo skirstinio parinkimas ir pradinė reikšmė

Tinkamo siūlomo skirstinio parinkimas yra esminis Markovo grandinės konvergavimo faktorius. Kuo šis skirstinys bus panašesnis į tikslinį, tuo greičiau Markovo grandinė pateks į stacionarų skirstinį, tačiau paprastai tikslinio skirstinio forma ir dydis dažnai yra nežinomi [15], o tai ypač svarbu, kuomet susiduriama su daugiamatiškumu. Paprastai siūlomas skirstinys yra pasirenkamas iš tam tikros, žinomos skirstinių klasės [16].

Visi MCMC algoritmai reikalauja pradinės reikšmės, nuo kurios yra pradeda konstruoti Markovo grandinė. Kai grandinė sukverguoja į stacionarų pasiskirstymą, gautieji stebėjimai bus nepriklausomi nuo pradinės vertės. Paprastai, nuo algoritmo vykdymo pradžios pirmieji gautieji stebėjimai gali tiksliai neatspindėti norimo skirstinio, todėl yra pašalinami. Toks periodas, iki kurio grandinė „pamiršta“ savo pradinę reikšmę yra vadinamas deginimo (angl. burn-in) periodu.

Visgi, pradinių verčių pasirinkimas kartais gali daryti įtaką grandinės veikimui, o ypač konvergavimo aptikimo greičiui ir ilgam deginimo periodui. Literatūroje pradinės vertės siūloma pasirinkti tokiais būdais:

- leidžiant kelias lygiagrečias grandines su labai skirtingais, tarpusavyje didelės dispersijos pradiniais taškais, kurie gali būti gauti taikant modos suradimo (angl. mode-finding) algoritmą [17];
- naudojant imituoto atkaitinimo algoritmus siekiant gauti didelės dispersijos pradinių reikšmių stebėjimus [18];
- naudojant didžiausio tikėtimumo metodo įverčius [18];
- remiantis bendrąja informacija ir žiniomis apie turimus duomenis [18].

1.4.2. Markovo grandinių Monte Karlo metodų konvergavimas ir diagnostika

Kaip jau buvo minėta, periodas iki Markovo grandinės konvergavimo į stacionarų pasiskirstymą yra vadinamas deginimo periodu. Gautieji stebėjimai, susidarę deginimo metu, nepriklauso galutiniam pasiskirstymui, todėl atmesti pirmuosius metodo veikimo metu gautuosius narius yra įprasta procedūra, nes pradinės vertės šiems stebėjimams pernelyg stipriai daro įtaką ir nepateikia geros informacijos apie dominantį tikslinį skirstinį. Tam tikra stebėta vertė iš tikslinio skirstinio yra gaunama tik tada, kai Markovo grandinės iteracijų skaičius artėja prie begalybės, o to praktikoje pasiekti neįmanoma.

Konvergavimo bei gautųjų pirmųjų stebėjimų, nepriklausančių stacionariam skirstiniui, skaičiaus nustatymas dažnai yra subjektyvus, nes tiksliai nustatyti, kada Markovo grandinė pateko į stacionarų pasiskirstymą yra sunku. Idealus sprendimas nustatyti konvergavimą yra analitiškai apskaičiuoti MCMC algoritmo konvergavimo rodiklį ir tuomet paimti pakankamą iteracijų skaičių, atitinkantį nustatytus atitikties kriterijus, tačiau bendruoju atveju to padaryti neįmanoma [7]. 1992 m. buvo pasiūlyta, jog pakanka išmesti pirmuosius 1-2% iteracijų, kai grandinė yra pakankamai ilga, jeigu išvengiama ekstremalių pradinių reikšmių [19]. Kadangi konvergavimo periodas priklauso nuo pradinės iteracijos reikšmės, todėl blogas iteracijos spėjimas gali prailginti šį periodą, kuris būtų ilgesnis nei 2% grandinės.

Veiksmingesnis metodas yra tiesiog nubraižyti keletą MCMC metodo vykdymo metu sugeneruotų gautųjų atsitiktinių skaičių laiko eilutes ir vizualiai nustatyti, kada grandinė konverguoja nei atmesti dalį grandinės, todėl konvergavimo diagnostikai atlikti gali būti pasitelkiami ir grafiniai metodai. Dar vienas būdas yra brėžti gautųjų stebėjimų vidurkių grafiką ir juos atvaizduoti nustatant kas kelintą k -ąją iteraciją. Taip pat dažnai konvergavimo diagnostikoje naudojamas autokoreliacinės funkcijos (geriau žinomos trumpiniu ACF) grafikas. Šis įrankis nėra tiksli konvergavimo diagnostika, tačiau netiesiogiai padeda įvertinti MCMC metodų konvergavimą, nes metodo metu sugeneruojamos labai koreliuotos vertės [14].

Konvergavimui tirti pasitelkiami ir dispersijos santykiai. Šių metodų metu vertinama ir lyginama dispersija tarp keleto sugeneruotų grandinių su skirtingomis pradinėmis iteracijos vertėmis bei gautosios sekos viduje esančių reikšmių. Vienas žymiausių metodų, kuriame naudojama ši idėja, yra Gelmano ir Rubino metodas [17], kurio metu lyginamos kelios sugeneruotos eilutės, gautos pasirinkus skirtingas, tarpusavyje didelės dispersijos, pradines iteracijos vertes. Šiame metode vertinamas antros eilės momentas (dispersija), tačiau literatūroje svarstoma naudoti ir kitos eilės momentus [20].

Taip pat egzistuoja metodai, kuriuose gautoji atsitiktinių skaičių eilutė traktuojama kaip laiko eilutė, o jos stacionarumas remiasi spektrine analize. Vadinasi, pagrindinė tokių metodų idėja yra sugeneruoti stacionarią laiko eilutę MCMC metodo pagalba. Vienas populiariausių metodų yra Geweke konvergavimo diagnostika [21]. Šis metodas pagrįstas vienos grandinės naudojimu ir yra tinkamas, kuomet nagrinėjamas stebėtos imties vidurkio (arba tam tikros funkcijos) konvergavimas. Grandinė yra padalijama į dvi dalis, yra pirmieji 10% ir paskutiniai 50% iteracijų. Jei visa grandinė yra stacionari, pradžios ir pabaigos verčių vidurkiai eilutėje turėtų būti panašūs. Kitas metodas ištirti konvergavimą buvo pasiūlytas Heidelbergerio ir Welcho [22]. Šis metodas iš esmės naudoja Cramer-von-Mises statistiką norint ištirti nulinę hipotezę ar Markovo grandinė yra iš stacionaraus pasiskirstymo.

Konvergavimo diagnostikai atlikti galima rasti ir būdų, besiremiančių Markovo grandinių teorija. Viena populiariausių tokių diagnostikų yra Raftery ir Lewis metodas [23], kuriame naudojama dviejų būsenų Markovo grandinės teorija, skirta vienos ilgos Markovo grandinės konvergavimui nustatyti. Šios diagnostikos metu yra apskaičiuojamas iteracijų bei deginimo periodo skaičius, taip pat įvertinamas priklausomumo faktorius. Markovo grandinių teorija remiasi ir Garreno bei Smitho diagnostika, tačiau šis metodas skaičiavimo pagrindu yra sudėtingesnis ir naudojamas tik Gibbso atrankoje [24].

Galima išskirti ir kitus metodus, naudojamus specifiniams MCMC algoritmams. Pavyzdžiui, Ritterio ir Tannerio, Liu, Liu, ir Rubino diagnostikos yra skirtos tik Gibbso atrankoms. Abu šie metodai empiriniai, kurių pagrindą sudaro pereinamasis branduolys (angl. transition kernel) [25].

Idealiu atveju sakoma, kad Markovo grandinė yra konvergavusi tuomet, kai gautieji stebėjimai buvo gauti iš viso skirstinio, tačiau aptikti konvergavimą yra sunku, kadangi tikslinio skirstinio forma nėra žinoma. Realizuojant MCMC metodus, dažnai kyla klausimas, kada sustabdyti Markovo grandinę. Viena iš ilgos grandinės generavimo problemų yra tai, kad algoritmas gali neanalizuoti viso tikimybinio skirstinio, nors gali atrodyti, jog grandinė yra konvergavusi [18]. Ši problema dažniausiai sprendžiama paleidžiant kelias grandines lygiagrečiai su skirtinga pradine verte, todėl skirtingų Markovo grandinių palyginimas padeda geriau suprasti konvergavimą.

Apibendrinant verta paminėti, kad konvergavimo diagnostikos yra ypač nepatikimos tuo atveju, jei grandinė konverguoja labai lėtai, kadangi jos remiasi gana maža gautųjų stebėjimų apimtimi.

1.4.3. Markovo grandinių Monte Karlo metodų taikymai

MCMC metodai plačiai naudojami išgauti empirinius įverčius ir padaryti išvadą apie daugiamačius skirstinius, kurie dažniausiai sutinkami Bajeso statistiniame modeliavime, taip pat deterministinių ir stochastinių diferencialinių lygčių skaitmeninėse integracijose [26].

MCMC taikymo sričių galima rasti ir skaičiuojamoje biologijoje, fizikoje ar chemijoje ir jų sistemų modeliavime. Daugelis biologinių procesų yra modeliuojami kaip įprastų diferencialinių lygčių sistemos, sprendžiamos MCMC metodu [27]. Bajeso statistika ir MCMC metodai yra naudojami įvairiuose modeliuose, tokiuose kaip deguonies režimo prognozavimas [28], siekiant sumažinti netikslus ekosistemos modelio parametrus, sukeliančius modelio imitavimo neapibrėžtumą [29] bei įvertinant įvairius žemės, klimato sistemų modelius [30], vertinant cheminių medžiagų išleidimą į aplinką [31] ar Hamiltono dinamikoje [32].

Kita taikymo sritis yra medicina. Šioje srityse MCMC metodai naudojami prognozuojant ligos tendencijas ateityje arba siekiant atlikti tinkamą prevencinį darbą ir planuoti teikti specializuotą medicininę priežiūrą pacientams [33], ligų raidas po įvykusių infekcijų [34] ar siekiant nustatyti pirminių navikų buvimą ir ligų diagnozavimą [35, 36]. Taikymų galima rasti ir farmacijoje atliekant kiekybinės rizikos analizę [37].

Šių metodų taikymo galimybes galima rasti ir finansų srityje. MCMC gali būti pritaikomas modelio parametrus įvertinti palūkanų normų modeliavime [38]. Kita svarbi taikymo sritis – naudoti MCMC metodus empiriniuose įmonių finansuose vertinant įvairių sudėtingų modelių parametrus [39] ar akcijų indeksus su šuoliais [40].

MCMC metodai taikomi ir astronomijoje. Panaudojant šiuos metodus, čia atliekama duomenų analizė ir sprendžiami tokie uždaviniai, kaip kinematinis ir dinaminis galaktikos modeliavimas, kai siekiama suprasti galaktikų kilmę ir evoliuciją, žvaigždžių būseną ir sudėtį, masę, amžių ir kitą informaciją [41].

MCMC algoritmų taikymas galimas ir kriptografijoje. Realus pranešimo iššifravimo pavyzdys MCMC metodu: kalėjimo psichologas atvyksta į Stanfordo universitetą su užkoduotomis kalinių rašytomis žinutėmis, o Marcas Koramas panaudodamas Metropolio algoritmą jas iššifruoja [42]. Panaudojant MCMC metodą, toks šifravimo būdas yra bandomas naudoti perkėlimo ir pakartotinio perkėlimo šiframs [43].

1.5. Literatūros apibendrinimas

Akcijų kainų prognozavimas yra sudėtingas uždavinys, kadangi jų vertė priklauso nuo įvairių faktorių. Priklausomai nuo investuotojo poreikio galima prognozuoti ilgalaikes arba trumpalaikes akcijų kainų kitimo tendencijas, pasitelkiant fundamentaliąją arba techninę analizes, įvairias hipotezes bei prielaidas apie akcijų rinką.

Markovo grandinių Monte Karlo metodai turi plačias taikymo galimybes praktikoje ir įvairiose mokslo srityse, ypač kai susiduriama su daugiamatiškumu. Atsitiktinių skaičių sekai generuoti pateikiami įvairūs algoritmai, naudojami specifiniais atvejais. Labai svarbu MCMC metodo vykdymo metu pasirinkti tinkamą siūlomą skirstinį bei pradines reikšmes, nuo kurių priklauso grandinės konvergavimas, nes pirmosios grandinės reikšmės neapibūdina tikslinio skirstinio.

Siekiant įvertinti grandinės konvergavimą reikalinga atlikti įvairias konvergavimo diagnostikas. Bendroju atveju, vieno metodo, nustatančio, kada grandinė pateko į stacionarų pasiskirstymą, nėra, todėl konvergavimui ištirti reikalinga pasitelkti kelias diagnostikas. Šios diagnostikos yra ypač nepatikimos, kai grandinė konverguoja labai lėtai, todėl įvertinti, kiek pirmųjų iteracijų reikia atmesti yra sunku, o ir konvergavimo nustatymas dažnai yra subjektyvus.

2. Medžiagos ir tyrimo metodai

2.1. Programinė įranga

Darbui atlikti pasitelkiama RStudio programinė įranga. Tai nemokama, atviro kodo integruotoji kūrimo aplinka, skirta atlikti įvairius statistinius skaičiavimus. Šioje programinėje aplinkoje naudojama R programavimo kalba, kuri plačiai taikoma statistikos programinės įrangos kūrimui bei duomenų analizei. R galimybės gali būti smarkiai išplečiamos panaudojant vartotojų sukurtus paketus – funkcijų ir duomenų rinkinius, kurie leidžia naudoti specializuotus statistinius metodus, grafinius įrenginius, importo ar eksporto galimybes, ataskaitų teikimo įrankius. Kadangi R – atvirojo kodo programinė įranga, todėl MCMC konvergavimui iširti yra panaudojamas paketas *coda* su joje esančiomis konvergavimo diagnostikomis: *gelman.plot()*, *geweke.diag()* bei *heidel.diag()*. Masyvo konvertavimui į MCMC objektą panaudojama funkcija *as.mcmc()*.

2.2. Ortogonalų eilučių įvertis

Ortogonalų eilučių tankio įvertis yra galinga neparimetrinė vertinimo metodika, leidžianti išanalizuoti ir pateikti duomenis be išankstinės informacijos apie pagrindinio tankio formą [44]. Adaptyviojo ortogonalų eilučių tankio įverčio konstravimo idėja paaiškinama klasikiniu pavyzdžiu, tiesiogiai generuojant imtį iš vienmačio tankio.

Ortogonalų eilučių pasirinkimas pirmiausia priklauso nuo tankio padėties. Apskritai, kai tankis yra intervale $(-\infty; +\infty)$ arba $(0; +\infty)$, tuomet rekomenduojama naudoti Hermito arba Laguerro eilutes. Jeigu tankis f turi aprėžtą apibrėžimo sritį, pavyzdžiui $[0; 1]$, tuomet rekomenduojama rinktis trigonometrines (arba Furjė) eilutes. Be to populiariu rinktis klasikinius ortogonaluosius polinomus, tokius kaip Ležandro, Gegenbauerio, Jakobio ar Čebyševio.

Tarkime, kad bandome įvertinti tankį f vienetiniame intervale $[0; 1]$. Ortogonalios eilutės apibrėžia tankį f , įvertinant jo Furjė plėtinio koeficientus. Pagrindinė ortogonalų polinomų eilutės metodo idėja yra ta, kad bet kuri kvadrinė integruojama funkcija f , šiuo atveju tankio funkcija, gali būti suprojektuota panaudojus ortogonalų bazę $\{\varphi_j\}$ [45]:

$$f(x) = \sum_{j=0}^{\infty} \theta_j \varphi_j(x),$$

čia

$$\theta_j = \int_0^1 \varphi_j(x) f(x) dx.$$

Paprastai parametras θ_j yra vadinamas j -uoju tankio f Furjė koeficientu.

Tarkime, kad X yra atsitiktinis dydis su tankiu f , todėl koeficientas θ_j gali būti užrašytas kaip vidurkis:

$$\theta_j = \int_0^1 \varphi_j(x) f(x) dx = E\{\varphi_j(X)\}.$$

Gautasis pagrindinis rezultatas bet kokiam ortogonalinių eilučių skaičiavimui suteikia galimybę apskaičiuoti θ_j panaudojant imties X_1, \dots, X_n iš tankio f vidurkį, o gautasis įvertis yra nepaslinktasis:

$$\hat{\theta}_j = \frac{1}{n} \sum_{i=1}^n \varphi_j(X_i).$$

Norint gauti rezultatyvų tankio f įvertinimą, būtina pritaikyti žemo dažnio filtrą koeficientų sekai $\hat{\theta}_j$. Lengviausias būdas tai padaryti yra sutrumpinti sumą $\sum_{j=0}^{\infty} \hat{\theta}_j \varphi_j$. Pasirinkus sveikąjį skaičių K , vadinamą aproksimacijos eile (angl. cutoff point), nusakančia glotnumą, gaunamas toks tankio įvertis:

$$\hat{f}(x) = \sum_{j=1}^K \hat{\theta}_j \varphi_j(x).$$

2.2.1. Čebyševio polinomial

Ortogonalinių polinomų seka – tai tokia seka, kurioje du skirtingi polinomial sekoje yra ortogonalūs vienas kitam (skaliarinė sandauga lygi nuliui). Viena iš tokių sekų yra Čebyševio polinomial, kurie gali būti apibrėžiami rekursiškai [46]. Paprastai išskiriami pirmo ir antro tipo Čebyševio polinomial, tačiau kai kuriose knygose ir daugumoje straipsnių (kaip ir šiame darbe), sąvoka „Čebyševio polinomial“ vartojama tik pirmo tipo Čebyševio polinomui $T_n(x)$, apibrėžiamu tokiu būdu:

$$T_n(x) = \cos n\theta, \quad x = \cos \theta.$$

Jei kintamasis x yra intervale $[-1; 1]$, tuomet atitinkamas kintamasis θ yra iš intervalo $[0; \pi]$.

Visgi gerai žinoma, kad $\cos n\theta$ yra n -ojo laipsnio $\cos \theta$ polinomial:

$$\begin{aligned} n = 0: & \quad \cos 0\theta = 1, \\ n = 1: & \quad \cos 1\theta = \theta, \\ n = 2: & \quad \cos 2\theta = 2 \cos^2 \theta - 1, \\ n = 3: & \quad \cos 3\theta = 4 \cos^3 \theta - 3 \cos \theta, \end{aligned}$$

.....

Panaudojant pažymėjimą iš apibrėžimo, kad $x = \cos \theta$, gauname:

$$\begin{aligned} n = 0: & \quad T_0(x) = 1, \\ n = 1: & \quad T_1(x) = x, \\ n = 2: & \quad T_2(x) = 2x^2 - 1, \end{aligned}$$

$$n = 3: \quad T_3(x) = 4x^3 - 3x,$$

.....

Praktikoje toks kiekvieno $T_n(x)$ skaičiavimas nėra patogus, todėl pritaikius trigonometrines tapatybes

$$\cos n\theta + \cos(n-2)\theta = 2 \cos \theta \cos(n-1)\theta$$

bei panaudojus pažymėjimą iš apibrėžimo, kad $x = \cos \theta$, gauname

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), \quad n = 2, 3, 4, \dots$$

Gautoji formulė rekursyviai generuoja visus polinomus, kai turimos šios pradinės sąlygos:

$$T_0(x) = 1, \quad T_1(x) = x.$$

2.3. Monte Karlo metodas

Monte Karlo metodai – tai metodai, skirti įvertinti ir apskaičiuoti integralus tais atvejais, kai analizinis (analitinis) sprendinys negali būti apskaičiuotas. Su tuo dažnai susiduriama Bajeso metoduose, kai uždavinio problema yra daugiamatė, todėl daugiamatžio integralo integracija yra negalima. Su šiais integralais taip pat susiduriama ieškant atsitiktinio kintamojo ribinio pasiskirstymo arba ieškant jo tikėtinos reikšmės.

Monte Karlo metodas remiasi didžiųjų skaičių dėsniumi, kuris sako, kad esant dideliame nepriklausomų vienodai pasiskirsčiusių dydžių skaičiui labai tikėtina, jog jų aritmetinis vidurkis mažai skirsis nuo tikrojo vidurkio. Todėl, kai stebėjimai X_t yra nepriklausomi, didžiųjų skaičių dėsnis užtikrina, kad ši aproksimacija gali būti labai tiksli, didinant stebėjimų skaičių n . Ši skaitinė aproksimacija apibūdinama tokiu būdu:

$$E[f(X)] \approx \frac{1}{n} \sum_{i=1}^n f(X_i).$$

Šioje lygtyje X_i žymi gautuosius stebėjimus iš tikimybinio skirstinio $\pi(x)$, o n – bendrą imties skaičių.

Monte Karlo metodų taikymas reikalauja generuoti imtis iš norimo (tikslinio) pasiskirstymo su teisingu dažniu. Tai gali būti problematiška, jei apibūdinti pasiskirstymą yra sunku. Šios problemos sprendimas gali būti Markovo grandinių naudojimas, kurios leidžia gauti imtis iš bet kokio pasiskirstymo tipo.

2.4. Metropolio-Hastingso algoritmas

Vykdamas Metropolio-Hastingso algoritmą, kiekvieno laiko momentu t , kita būseną X_{t+1} yra pasirenkama iš pradžių atrenkant tašką kandidatą Y iš siūlomo skirstinio $q(\cdot | X_t)$. Atkreipiame

dėmesį, kad siūlomas skirstinys gali priklausyti nuo dabartinės būsenos X_t . Kandidatas taškas Y yra priimamas su tikimybe $\alpha(X_t, Y)$:

$$\alpha(X_t, Y) = \min\left(1, \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)}\right),$$

čia $\pi(\cdot)$ – tikslinis skirstinys.

Jei taškas kandidatas yra priimamas, tuomet kita būsena tampa $X_{t+1} = Y$, kitu atveju, kai šis taškas yra atmetamas, grandinė nepasikeičia, t. y. $X_{t+1} = X_t$.

Taigi, Metropolio-Hastingso algoritmas vykdomas tokiu būdu:

1. Inicijuojamas X_0 . Nustatomas $t = 0$.
2. Kartojama:
 - a. Taškas Y stebimas iš pasiskirstymo $q(\cdot | X_t)$, $Y \sim q(\cdot | X_t)$.
 - b. Sugeneruojamas atsitiktinis kintamasis U iš tolydaus skirstinio, $U \sim Uniform(0, 1)$.
 - c. Tikrinama sąlyga ar $U \leq \alpha(X_t, Y)$:
 - i. Jei sąlyga teisinga, tai $X_{t+1} = Y$;
 - ii. Kitu atveju $X_{t+1} = X_t$;
 - d. pakeičiamas $t = t + 1$.

Siūlomas skirstinys $q(\cdot | \cdot)$ gali turėti bet kokia formą, o grandinės stacionarus skirstinys bus $\pi(\cdot)$.

2.5. MCMC konvergavimas

MCMC metodo metu tam tikra stebėta vertė iš tikslinio skirstinio $\pi(\cdot)$ yra gaunama tik tada, kai Markovo grandinės iteracijos artėja prie begalybės. Kadangi paprasto atsakymo, kiek atmesti iteracijų, nėra, todėl daugiausiai siekiama analizuoti grandinės konvergavimo charakteristikas. Toliau pateiksime populiariausius metodus grandinės konvergavimui tirti.

2.5.1. Geweke metodas

Geweke pasiūlė konvergavimo tyrimą, pagrįstą standartiniais laiko eilučių metodais [21]. Šis metodas pagrįstas vienos grandinės naudojimu ir yra tinkamas, kuomet nagrinėjamas stebėtos imties vidurkio (arba tam tikros funkcijos) konvergavimas. Grandinė yra padalijama į dvi dalis, kuriuose yra pirmieji 10% ir paskutiniai 50% iteracijų. Jei visa grandinė yra stacionari, pradžios ir pabaigos verčių vidurkiai eilutėje turėtų būti panašūs. Konvergavimo diagnostika Z yra skirtumas tarp dviejų vidurkių, padalintų iš jų asimptotinės standartinės paklaidos. Standartinė paklaida apskaičiuojama pagal spektrinį tankį, kai argumentas (dažnumas) yra lygus nuliui, todėl atsižvelgiama į bet kokią autokoreliaciją. Kai $n \rightarrow \infty$, tuo atveju, jei grandinė konverguoja, statistika Z yra pasiskirsčiusi pagal

normalųjį skirstinį $N(0, 1)$. Vadinasi, jeigu Z reikšmės patenka į skirstinio $N(0, 1)$ uodegas, tuomet tai parodo, kad grandinė dar nėra konvergavusi.

Verta paminėti, jog R programiniame pakete, *coda* bibliotekoje, atliekant Geweke diagnostiką, grandinė gali būti padalinta nebūtinai į nurodytus 10% bei 50% procentų – galima nustatyti ir kitus parametrus.

2.5.2. Gelmano ir Rubino metodas

Šis konvergavimo diagnostika nustato, kada Markovo grandinė „pamiršta“ savo pradinius taškus, lyginant kelias sekas, sugeneruotas iš skirtingų pradinių taškų [17]. Tarkime, jog turime m lygiagrečių sugeneruotų eilučių, kurių kintamojo X ilgis yra n . Šias vertes pažymime x_{ij} , kai $i = 1, \dots, m, j = 1, \dots, n$. Dispersija tarp sekų B ir vidinės sekos dispersija W apskaičiuojamos tokiu būdu:

$$B = \frac{n}{m-1} \sum_{i=1}^m (\bar{x}_i - \bar{x}_..)^2,$$

$$W = \frac{1}{m} \sum_{i=1}^m s_i^2, \quad s_i^2 = \frac{1}{n-1} \sum_{j=1}^n (x_{ij} - \bar{x}_i)^2.$$

Iš šių dviejų dedamųjų apskaičiuojamas dydžio X dispersijos įvertis tiksliniame skirstinyje:

$$S(X) = \frac{n-1}{n} W + \frac{1}{n} B.$$

Dėl pradinių verčių per didelės dispersijos, įvertis pervertina tikrąją dispersiją, tačiau yra nepaslinktasis, jei pradinis pasiskirstymas lygus stacionariniam pasiskirstymui. Vadinasi, santykis tarp viršutinės ir apatinės dydžio X standartinio nuokrypio ribos, vadinamos apytiksliu potencialaus skalės mažinimo ar susitraukimo koeficientu (angl. shrink factor), yra apibrėžiamas taip:

$$\sqrt{\bar{R}} = \sqrt{\frac{S(X)}{W}}.$$

Kai grandinė konverguoja, susitraukimo koeficientas mažėja iki 1, tai reiškia, kad lygiagrečios Markovo grandinės iš esmės yra sutampančios. Jei šis koeficientas yra didelis, tuomet atsitiktinius dydžius reikia generuoti ilgiau, siekiant konverguoti į stacionarų pasiskirstymą.

2.5.3. Heidelbergerio ir Welcho metodas

Šis metodas yra grandinių ilgio diagnostika, kuri iš esmės naudoja Cramer-von-Mises statistiką norint iširti nulinę hipotezę ir išsiaiškinti ar Markovo grandinė yra iš stacionaraus pasiskirstymo [22]. Jei nulinė hipotezė atmetama duotajam kintamajam, tuomet testas kartojamas išmetus pirmuosius 10% sugeneruotų stebėjimų. Jei nulinė hipotezė vėl atmetama, testas kartojamas vėl išmetus pirmas 10%

iteracijų. Šis procesas kartojamas tol, kol hipotezė apie turimą imtį iš stacionaraus skirstinio yra priimama arba 50% iš grandinėje esančių narių jau yra atmesti. Jei nulinė hipotezė vis dar atmetama, tuomet grandinė yra nestacionari ir turi būti vykdoma ilgiau.

Kai hipotezė priimama, tuomet atliekamas pusės pločio testas (angl. halfwidth test), kuris bando nustatyti, ar grandinė buvo paleista pakankamai ilgai, kad būtų pasiektas tikslumo lygis. Šiame teste panaudojama tenkinusi stacionarumo testą grandinės dalis ir apskaičiuojamas $1 - \alpha$ pasikliautinis vidurkio intervalas. Tuomet pusė šio intervalo pločio palyginama su gautųjų stebėjimų vidurkiu. Jeigu santykis tarp šių dydžių yra mažesnis už ε , testas yra išlaikomas. Priešingu atveju laikoma, kad imtis nėra pakankamai ilga, todėl reikalinga sugeneruoti didesnę imtį siekiant padidinti tikslumą.

2.6. Skirstinio aproksimavimas Čebyševio polinomais

Tyrimas pradedamas konstruojant tankio įvertinio funkciją $\pi(x)$, kuriai sukurti reikalinga pasirinkti aproksimacijos eilę K :

$$\pi(x) = \sum_{j=1}^K \hat{A}_j T_j(x).$$

Kadangi kintamieji Čebyševio polinomai $T_j(x)$ konverguoja intervale $[-1; 1]$, todėl, siekiant apskaičiuoti koeficientų seką \hat{A}_j , turimus duomenis y reikalinga transformuoti į šį intervalą panaudojus formulę:

$$\tilde{y} = 2 \frac{y - \min y}{\max y - \min y} - 1,$$

čia y – turimi duomenys, \tilde{y} – jau transformuoti duomenys į intervalą $[-1; 1]$.

Panaudojant transformuotus duomenis \tilde{y} , apskaičiuojama Furjė koeficientų seka \hat{A}_j :

$$\hat{A}_j = \frac{1}{n} \sum_{i=1}^n T_j(y_i).$$

Tankio įverčio funkcijos plotas visame intervale $[-1; 1]$ turi būti lygus 1, todėl duomenis papildomai reikalinga sunormuoti:

$$y' = \frac{\pi(\tilde{y}) - \min \pi(\tilde{y})}{\int_{-1}^1 (\pi(\tilde{y}) - \min \pi(\tilde{y})) d\tilde{y}}.$$

Kadangi nėra aišku, kokią optimalią aproksimacijos eilę K naudoti, todėl jos suradimui bus panaudojama pasirinkta tikslo funkcija $T(K)$, priklausanti nuo parametro K :

$$T(K) = \sum_{i=1}^3 \alpha_i |\hat{\mu}_i - \mu_i| + \sum_{i=1}^3 \beta_i |\hat{\mu}_i' - \mu_i'|,$$

čia α_i, β_i – laisvai pasirenkami koeficientai, μ_n, μ_n' – teoriniai (iš turimų duomenų) n -tos eilės pradiniai bei centriniai momentai, $\hat{\mu}_n, \hat{\mu}_n'$ – empiriniai n -tos eilės pradiniai bei centriniai ortogonalųjų eilučių aproksimacijos momentai. Aproksimacijos eilė K bus optimali, kai tikslo funkcijos reikšmė $T(K)$ bus mažiausia (skirtumų tarp momentų sumos bus mažiausios).

Momentai μ_n , esantys apie tam tikrą reikšmę c , apskaičiuojami tokiu būdu:

$$\mu_n = \int_{-\infty}^{+\infty} (y - c)^n \pi(y) dy.$$

Empiriniams momentams $\hat{\mu}_n$ bei $\hat{\mu}_n'$ apskaičiuoti yra panaudojami atsitiktiniai sugeneruoti taškai. Vertinant turimų duomenų histogramą, apibrėžiamas stačiakampis koordinačių taškų plokštumoje, todėl jo viduje yra generuojama daugybė taškų su atsitiktinėmis koordinatėmis (x, y) ir atrenkami tik tinkami taškai, esantys žemiau ortogonalųjų eilučių aproksimacijos. Kadangi tikslo funkcijoje lyginami teoriniai ir empiriniai momentai, todėl empirinius duomenis transformuojame į akcijų gražos normos intervalą panaudojant formulę:

$$x' = \frac{x + 1}{2} (\max y - \min y) + \min y.$$

Vadinasi, empiriniai momentai $\hat{\mu}_n$ apie tam tikrą reikšmę c apskaičiuojami taip:

$$\hat{\mu}_n = \int_{-\infty}^{+\infty} (x' - c)^n \pi(x') dx'.$$

2.7. Grandinės generavimas ir konvergavimas

MCMC atsitiktinių skaičių generavimui yra panaudojamas nepriklausomas Metropolio-Hastingso metodas. Jo metu generuojami kandidatai taškai (akcijų gražos) iš siūlomo atkarpomis tolydaus skirstinio $q(\cdot)$ ir tikrinama ar gautieji skaičiai yra iš tikslinio skirstinio – Čebyševio ortogonalųjų polinomų aproksimacijos $\pi(\cdot)$.

Be esančių metodų, grandinės konvergavimui tirti pasitelkiami statistiniai momentai ir bus stebima, kaip kinta momentų skirtumai pasirenkant tam tikras eilutės dalis. Vadinasi, toks tyrimo būdas bus atliekamas taip:

- Pasirenkami skaičiai k bei l atitinkamai nurodantys iš kiek sugeneruotų skaičių skaičiuojami momentai ir koku žingsniu, t. y., kas kiek iteracijų jie perskaičiuojami.
- Iš turimos grandinės $X = \{x_1, x_2, \dots, x_n\}$, kurią sudaro n narių, kiekvieną kartą skaičiuojami momentai pasirinktiems duomenims Y :

$$Y_i = \{y_{l \cdot i + 1}, y_{l \cdot i + 2}, \dots, y_{k + l \cdot i}\}, \quad i = 0, 1, \dots, \frac{n - k}{l}.$$

- Pasirinktiems duomenims iš aibės Y_i apskaičiuojami pradiniai $\hat{\mu}_{j; Y_i}$ bei centriniai momentai $\hat{\mu}_{j; Y_i}'$.
- Sukonstruojama tikslo funkcija

$$T(i) = \sum_{j=1}^3 |\hat{\mu}_{j; Y_i} - \mu_j| + \sum_{j=1}^3 |\hat{\mu}_{j; Y_i}' - \mu_j'|,$$

vertinanti skirtumus tarp empirinių ir sugeneruotų Metropolio-Hastingso algoritmo metu gautų dydžių.

- Braižomas tikslo funkcijos $T(i)$ priklausomybės grafikas nuo žingsnio i ir nagrinėjama gautoji priklausomybė.

2.8. Akcijų vertės prognozavimo modelis Markovo grandinių Monte Karlo metodu

Pasirinktų akcijų prognozavimui pasirenkamas multiplikatyvusis modelis, kuris yra apibrėžiamas taip [47]:

$$S_n = S_{n-1} \cdot v_n + S_{n-1},$$

čia S_n – akcijos vertė laiko momentu n , v_n – akcijos graža. Būsimoji akcijos vertė S_n priklauso nuo prieš tai buvusios akcijos vertės $n - 1$ laiko momentu ir akcijos gražos. Kadangi pastaroji yra atsitiktinė, ją prognozuosime MCMC metodu. Akcijų gražų normų tikimybinis pasiskirstymas įvertinamas ortogonalio eilučių tankio įverčiu:

$$v_n \sim \pi(\cdot)$$

Taškai kandidatai bus generuojami iš dalimis tolydaus skirstinio:

$$q(x) = \begin{cases} q_1, & x_0 < x \leq x_1 \\ q_2, & x_1 < x \leq x_2 \\ \dots & \dots \\ q_n, & x_{n-1} < x \leq x_n \end{cases}.$$

2.9. Monte Karlo modelis akcijų kainoms prognozuoti

Klasikinis Monte Karlo metodas akcijų vertėms prognozuoti yra apibrėžiamas tokiu būdu [47]:

$$S(t + \Delta t) = S(t) e^{(\delta + \frac{1}{2}\sigma^2)\Delta t + \sigma\sqrt{\Delta t}Z},$$

čia $S(t)$ – akcijos vertė laiko momentu, t , Z – atsitiktinės reikšmės iš standartinio normaliojo skirstinio, Δt – prognozavimo žingsnis, σ – logaritmuotų akcijų kainų standartinis nuokrypis, δ – gražos vidurkis (tolygiųjų palūkanų norma), apskaičiuojamas taip:

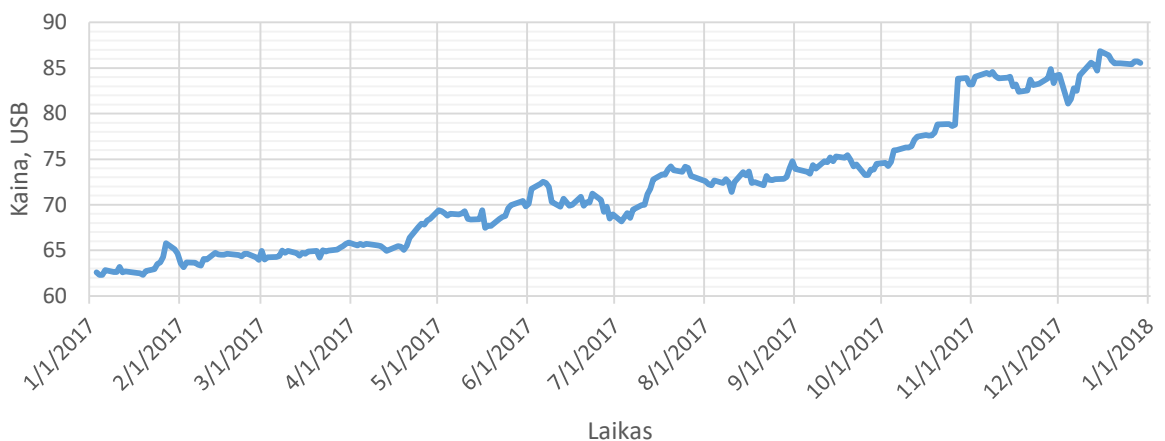
$$\delta = \frac{1}{n} \sum_{i=1}^n \log \left(\frac{S_i}{S_{i-1}} \right),$$

čia n – akcijų gražų kiekis. Pagrindinis šio modelio privalumas yra jo lengvas naudojimas, tačiau jis yra statistiškai pagrįstas tuomet, kai logaritmuotos akcijų gražų normos yra pasiskirsčiusios pagal normalųjį dėsnį.

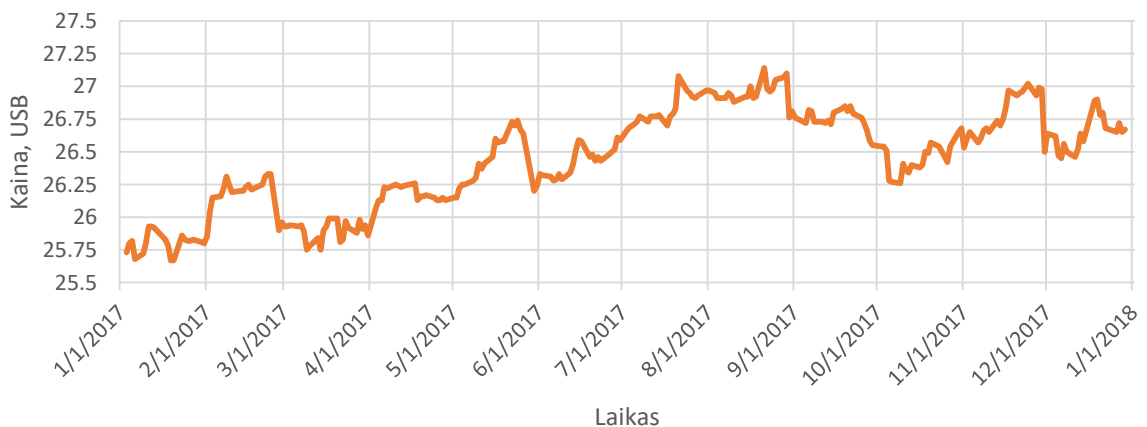
2.10. Istoriniai duomenys investicinio portfelio sudarymui

Investicinį portfelį sudaro penki vienetai didžiausios programinės įrangos gamintojos Microsoft Corporation bei penki vienetai tarptautinės finansinių paslaugų grupės Barclays Bank PLC akcijų. Gautajai investicinio portfelio vertei prognozuoti pasitelkiami 2017 metų (nuo sausio 1 d. iki gruodžio 29 d.) akcijų verčių duomenys pateikti 1 bei 2 prieduose. Investicinis portfelis bus prognozuojamas 18 dienų į priekį (nuo gruodžio 30 d. iki sausio 14 d.).

Pasirinktų duomenų akcijų dinamika atvaizduojama 1 paveiksle.



(a)



(b)

1 pav. 2017 metų akcijų kainų dinamika: Microsoft Corporation (a); Barclays Bank PLC (b)

2.11. Prognozavimo tikslumo matai

Modelių palyginimas ar prognozių tikslumas gali būti nustatomas atsižvelgiant į tai, kaip gerai modelis atitinka naujus duomenis, kurie nebuvo naudojami apibrėžiant modelį, todėl prognozės tikslumui įvertinti yra panaudojamos įvairios metrikos.

Tarkime, kad a_t – tikroji prognozuojamo vieneto reikšmė laiko momentu t , f_t – suprognozuota vertė laiko momentu t , o n – suprognozuotų reikšmių kiekis. Apibrėžiame metrikas, kurios bus panaudojamos šiame darbe:

- vidutinė procentinė paklaida (angl. Mean Percentage Error) MPE:

$$MPE = \frac{100\%}{n} \sum_{t=1}^n \frac{a_t - f_t}{a_t};$$

- vidutinė procentinė absoliutinė paklaida: (angl. Mean Absolute Percentage Error) MAPE:

$$MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{a_t - f_t}{a_t} \right|;$$

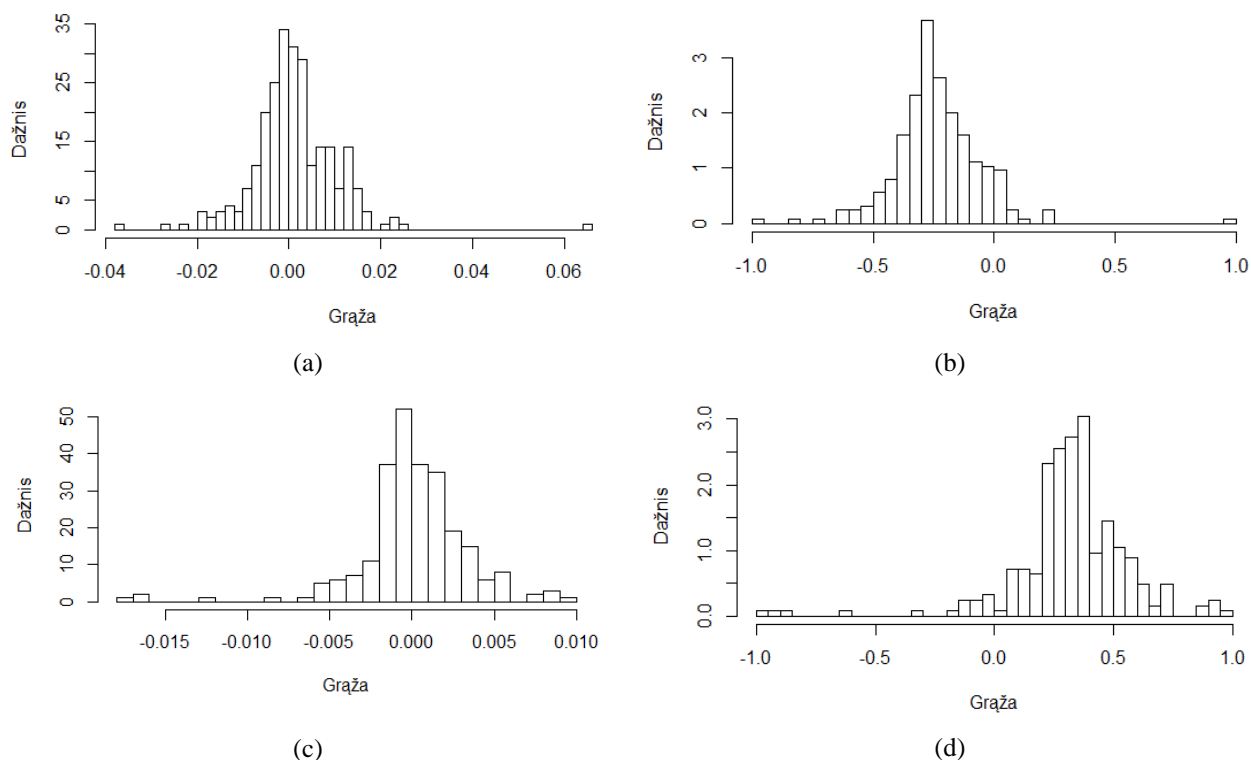
- vidutinė kvadratinė paklaida (angl. Mean Square Error) MSE:

$$MSE = \frac{1}{n} \sum_{t=1}^n (a_t - f_t)^2.$$

3. Tyrimų rezultatai ir jų aptarimas

3.1. Aproximacijos eilės nustatymas

Tyrimas pradedamas aproksimuojant turimų 2017-ųjų metų akcijų gražų tankį ortogonaliais Čebyševio polinomais. Prieš atliekant aproksimaciją duomenys turi būti transformuojami į intervalą $[-1; 1]$. Transformuotų duomenų histogramos yra atvaizduojamos 2 paveiksle.



2 pav. Microsoft Corporation (a, b) ir Barclays Bank PLC (c, d) akcijų gražų (kairėje) ir transformuotų į intervalą $[-1; 1]$ (dešinėje) gražų histogramos

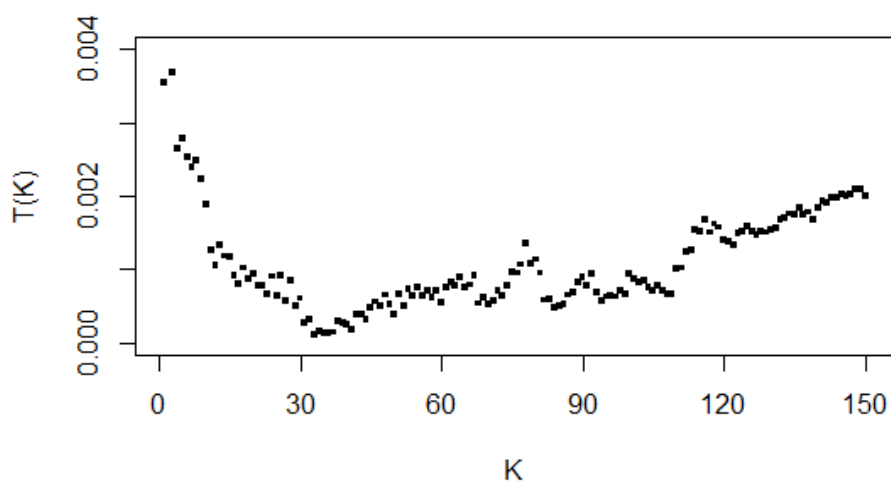
Turimus duomenis reikalinga aproksimuoti panaudojant optimalią aproksimacijos eilę K , todėl yra panaudojama tikslo funkcija $T(K)$:

$$T(K) = \sum_{i=1}^3 \alpha_i |\hat{\mu}_i - \mu_i| + \sum_{i=1}^3 \beta_i |\hat{\mu}'_i - \mu'_i|.$$

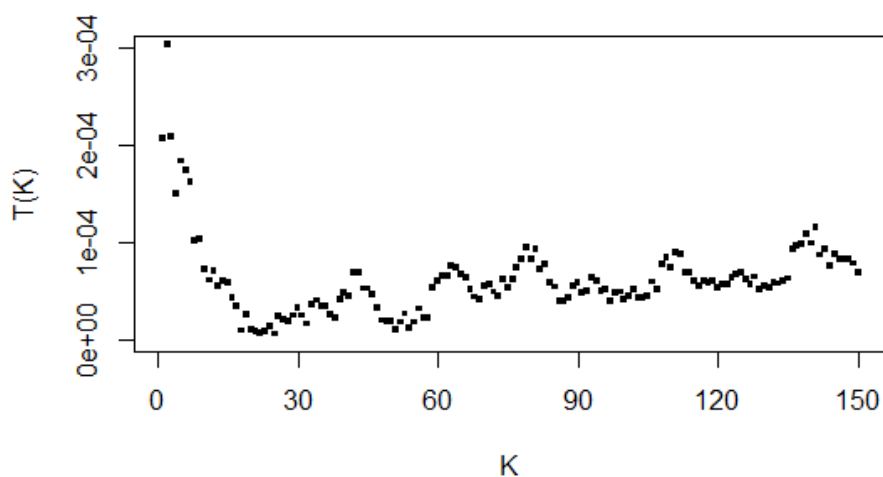
Empiriniams momentams gauti yra sugeneruojami 200000 atsitiktinių taškų $(x; y)$ transformuotų duomenų koordinatinių plokštumoje. Iš sugeneruotų taškų atrenkami tik tinkami taškai (esantys žemiau ortogonalinių eilučių tankio aproksimacijos) momentams apskaičiuoti. Kadangi tikslo funkcijoje vertinami skirtumai tarp momentų, todėl aproksimacijos eilė bus optimali su ta K reikšme, kuomet skirtumų tarp momentų sumos bus mažiausios, t. y. kartu ir funkcijos $T(K)$ reikšmė bus mažiausia. Vadinas, reikalinga nagrinėti tikslo funkcijos $T(K)$ priklausomybę nuo parametro K . Tikslo funkcijos reikšmėms apskaičiuoti pasirenkamos atsitiktinės parametrų vertės:

$$\alpha = (0.01; 6; 10), \quad \beta = (1; 1; 1).$$

Šios funkcijos su apibėžtais svoriais priklausomybės grafikas nuo aproksimacijos eilės K yra atvaizduojamas 3 paveiksle.



(a)



(b)

3 pav. Microsoft Corporation (viršuje) ir Barclays Bank PLC (apačioje) akcijų tikslo funkcijos $T(K)$ priklausomybė nuo aproksimacijos eilės K , kai pasirinkami svoriai yra $\alpha = (0.01; 6; 10)$, $\beta = (1; 1; 1)$

Iš 3 paveikslo galima pastebėti, jog vienareikšmiškai pasirinkti atsitiktinę aproksimacijos eilę yra netikslinga. Analizuojant tikslo funkcijos priklausomybę galima pastebėti, jog pasirinkus per mažą eilę, gautasis tankio įvertis nėra optimalus, o skirtumai taip teorinių ir empirinių momentų yra didžiausi. Kita vertus, pasirinkti per didelę aproksimacijos eilę K taip pat nėra gerai, nes šiam argumentui didėjant, tikslo funkcijos reikšmė pradeda didėti.

Kadangi ieškoma aproksimacijos eilė apibūdina skirstinį, o skirtumų svoriai buvo pasirinkti atsitiktinai, todėl reikalinga išbandyti ir kitus parametru α_i bei β_i variantus. Kiekvieną kartą pasirinkus svorius sugeneruojami vis kiti atsitiktiniai taškai $(x; y)$. Tokie bandymai yra atliekami

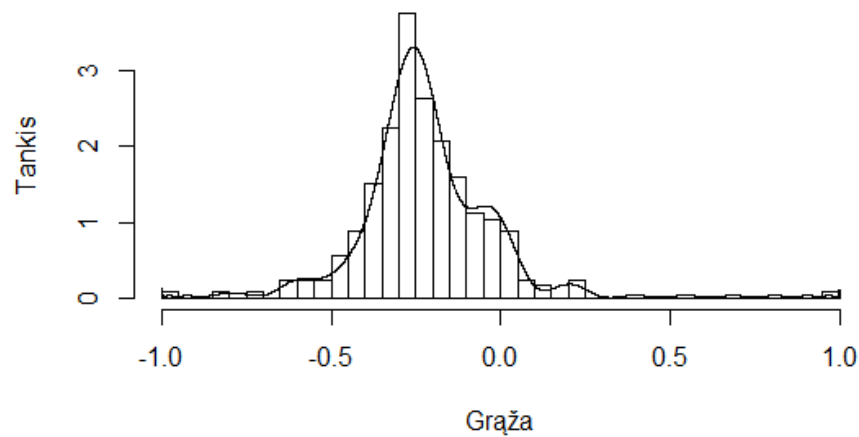
penkis kartus. Iš gautųjų bandymų yra išrenkamos didžiausios gautosios aproksimacijos eilės (tariame, jog aukštesnė aproksimacijos eilė geriau apibūdina turimus duomenis). Gautieji rezultatai yra pateikiami 1 lentelėje.

1 lentelė. Microsoft Corporation (MC) ir Barclays Bank PLC (BB) akcijų optimalių aproksimacijų eilės

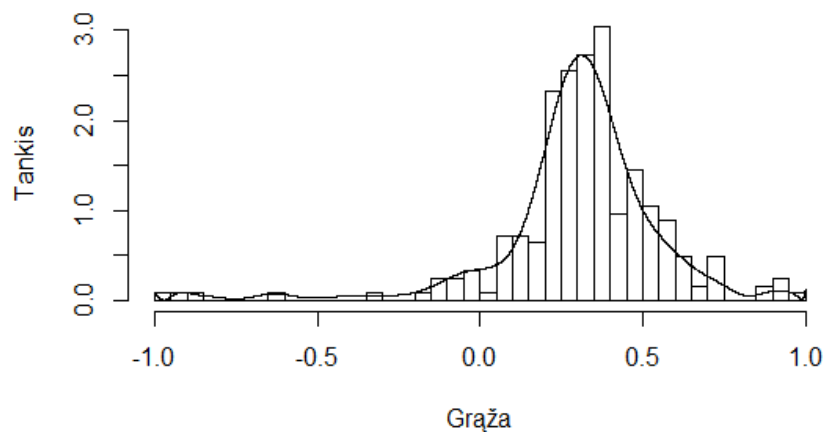
α / β	(15; 4; 3)		(0.01; 70; 10)		(0.7; 5; 0.9)		(44; 0; 90)		(8; 2; 0.3)		(1; 1; 1)	
	MC	BB	MC	BB	MC	BB	MC	BB	MC	BB	MC	BB
(0.003; 2; 0.8)	33	25	33	22	33	22	33	22	33	22	33	25
(0.01; 6; 10)	33	25	33	22	33	22	33	22	33	22	33	25
(0.05; 0.04; 12)	33	22	33	22	33	22	33	24	33	24	33	24
(4; 6; 13)	33	1	33	24	33	1	33	1	33	1	33	1
(0; 26; 9)	33	22	33	25	33	22	33	22	33	22	33	23
(16; 0.02; 0.65)	33	1	33	24	33	1	33	1	33	1	33	1
(8; 13; 34)	33	1	33	24	33	24	33	1	33	1	33	1
(1; 1; 1)	33	24	33	22	33	24	33	1	33	1	33	24

Iš gautųjų rezultatų 1 lentelėje galima pastebėti, jog Microsoft Corporation akcijų gražų aproksimavimui naudotieji svoriai įtakos nedaro. Visais atvejais bandymų metu yra gaunama, kad optimali aproksimacijos eilė yra $K = 33$. Tačiau vertinant Barclays Bank PLC duomenų aproksimacijos eilę, gaunami įvairesni rezultatai. Priklausomai nuo svorių kinta ir parametro reikšmė, todėl iš visų bandymų yra išrenkamas pats didžiausias gautas rezultatas $K = 25$. Kadangi nėra gaunamas tik vienas rezultatas todėl daroma prielaida, jog aukštesnė aproksimacijos eilė geriau apibūdina turimas akcijų gražas. Galima daryti išvadą, kad bendruoju atveju, atsitiktinis parametru α bei β pasirinkimas nėra tinkamas, nes keičiant šiuos parametrus nėra gaunamas tik vienas rezultatas.

Abu gautieji parametrai yra panaudojami akcijų gražų normoms aproksimuoti, todėl toliau 4 paveiksle atvaizduojama turima transformuotų duomenų histograma bei Čebyševio polinomu aproksimacija. Stebint 4 paveiksle esančius grafikus galima pamatyti, kad gautosios aproksimacijos eilės visai gerai apibūdina turimas transformuotas akcijų gražas.



(a)



(b)

4 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų grąžų histogramos ir aproksimacijos Čebyševio polinomais palyginimas, kai parametrai atitinkamai lygūs $K = 33$, $K = 25$

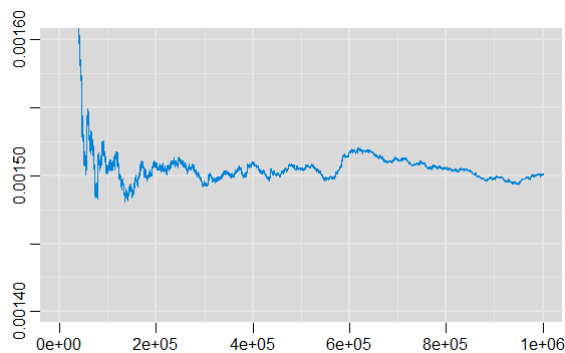
3.2. Markovo grandinės konvergavimo tyrimas

Gautosios Markovo grandinės konvergavimo tyrimui atlikti kiekvienai akcijų grąžai yra sugeneruojamos trys atskiros, susidedančios iš 1000000 atsitiktinių skaičių grandinės, kiekviena jų yra generuojama pasirinkus skirtingus pradinis iteracijų taškus X_0 , lygius 0.1, -0.01 bei 0. Kadangi bus naudojama Gelmano ir Rubino konvergavimo diagnostika, kuri rekomenduoja rinktis didelės dispersijos pradines iteracijų vertes, todėl pirmasis pradinis taškas pasirenkamas neįprastas akcijų grąžai, kiti du taškai – atsižvelgiant į realias galimas akcijų grąžas iš turimų duomenų histogramų.

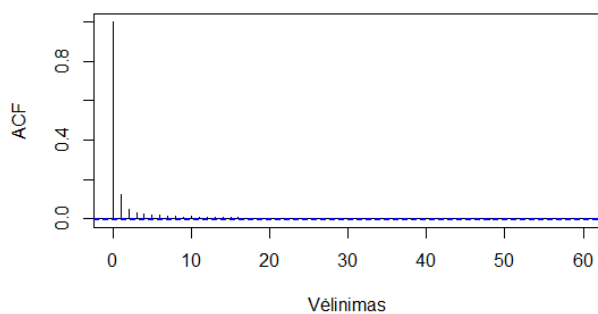
3.2.1. Grafinis konvergavimo nustatymas bei skaitinės charakteristikos

Iš pradžių konvergavimui tirti bus pasitelkiami grafiniai metodai: braižomi eksponentiniai skenkančio vidurkio bei autokoreliacinės funkcijos (ACF) grafikai. Slenkančio vidurkio grafikai naudojami įvertinti duomenų trendo kryptį. Kadangi grandinė turi konverguoti į stacionarųjį skirstinį, todėl vidurkis turi nusistovėti. ACF grafikas parodo koreliaciją tarp dviejų gautosios sekos narių kintant vėlinimui t. y. atstumui tarp narių.

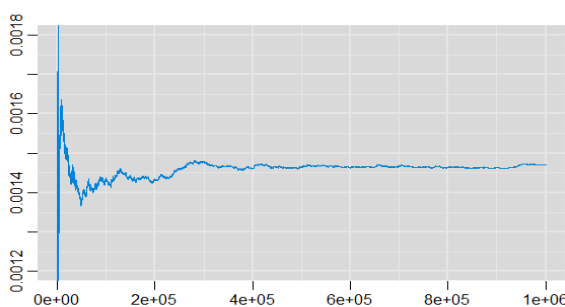
Gautieji atsitiktinių sekų rezultatai yra pateikiami 5 ir 6 paveiksluose.



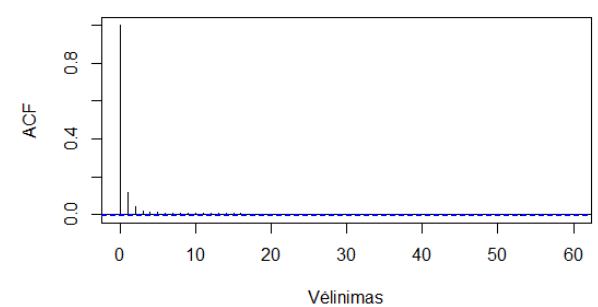
(a)



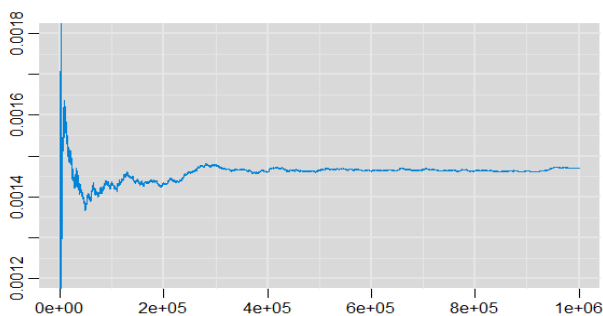
(b)



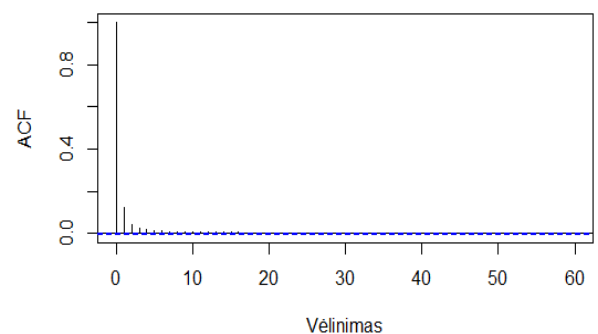
(c)



(d)



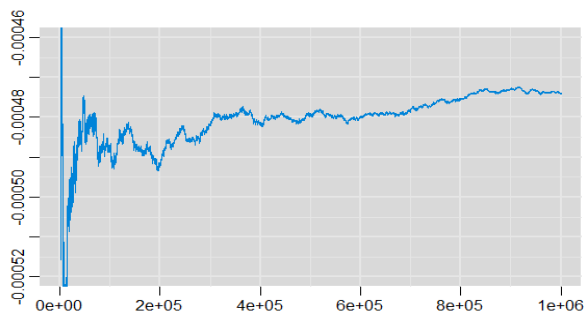
(e)



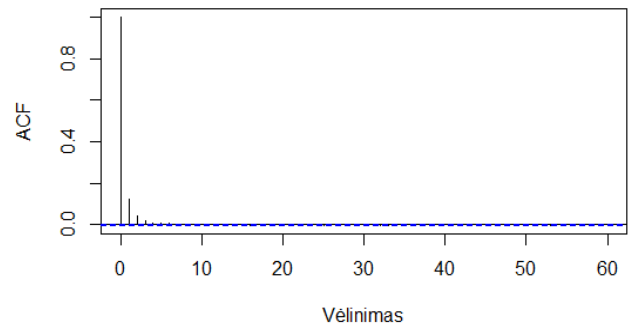
(f)

5 pav. Microsoft Corporation slenkančio vidurkio ir autokoreliacijos funkcijos grafikai, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f)

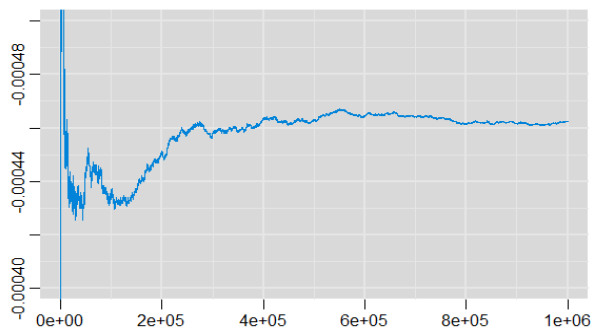
Nagrinėjant Microsoft Corporation akcijų grąžas galima pastebėti, kad slenkančio vidurkio grafikas neturi trendo ir yra pastovus. ACF grafikas gęsta greitai, vadinasi, koreliacija tarp narių yra maža. Kadangi grandinėje mažai pasikartojimų galima spėti, jog siūlomas dalimis tolygus skirstinys yra tinkamas generuoti taškus kandidatus.



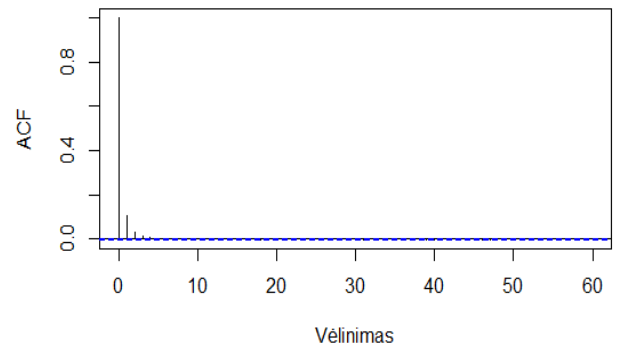
(a)



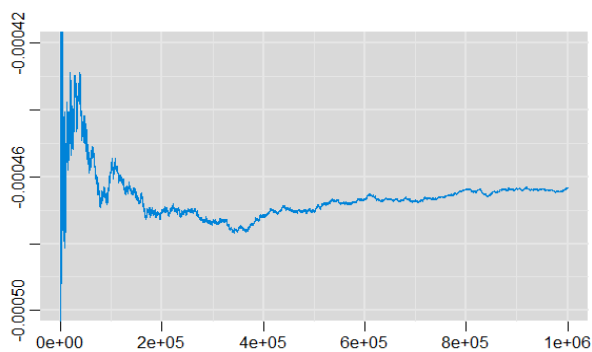
(b)



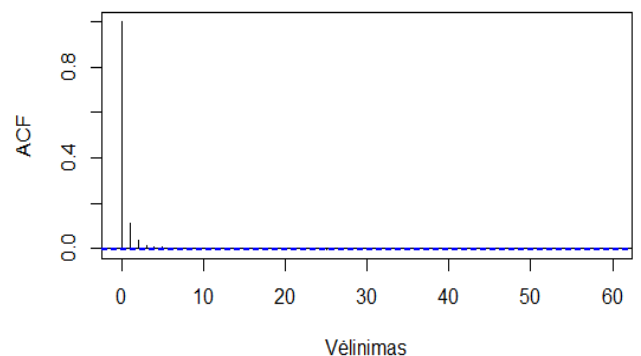
(c)



(d)



(e)

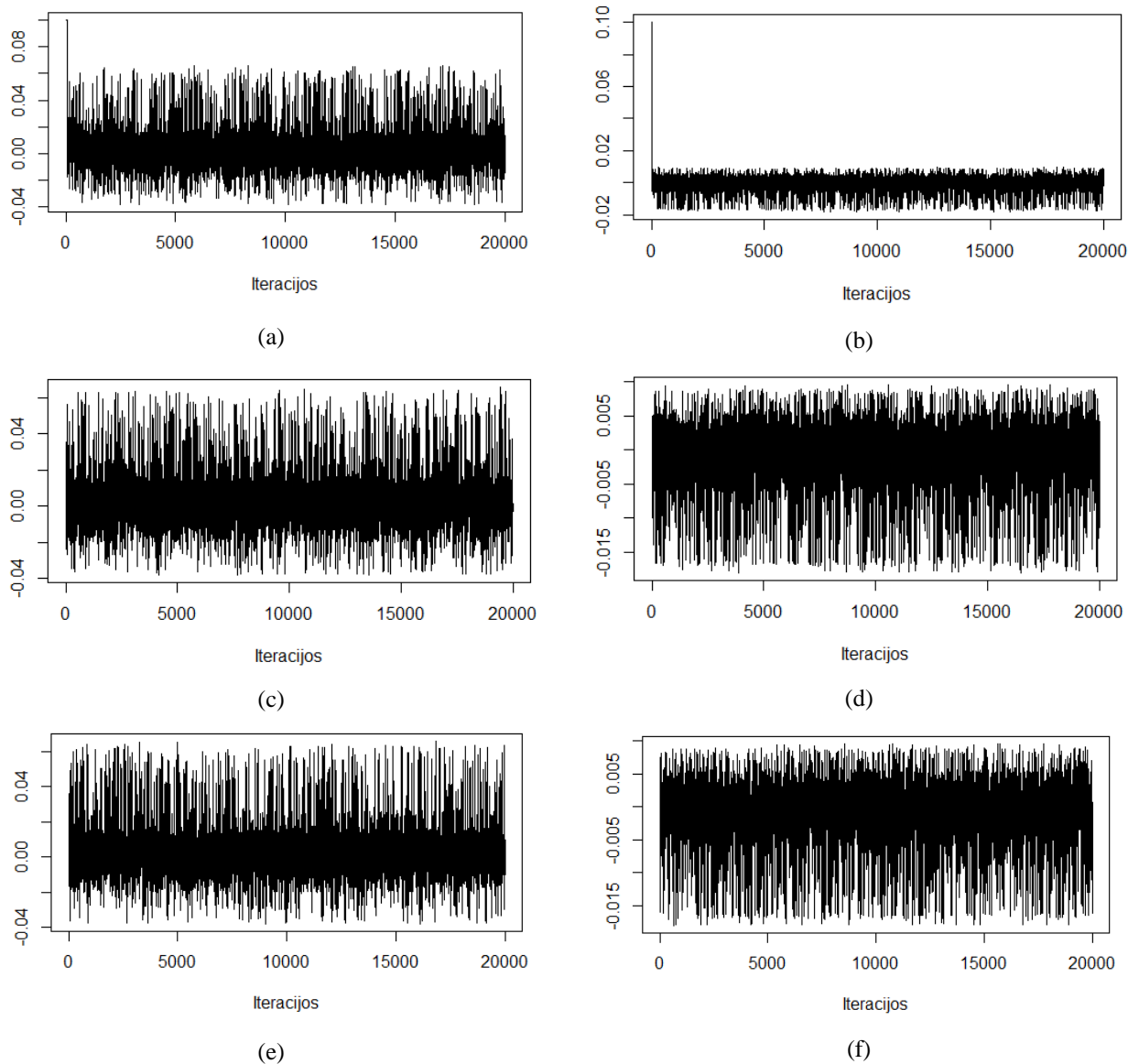


(f)

6 pav. Barclays Bank PLC slenkančio vidurkio ir autokoreliacijos funkcijos grafikai, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f)

Analogiškai kaip 5 paveiksle, stebint Barclays Bank PLC slenkančio vidurkio grafikus (6 paveikslas) nepastebimas stipriai svyruojantis trendas. Taip pat ACF grafikas parodo mažą autokoreliaciją tarp narių, todėl galima numanyti, jog taškai kandidatai dažniausiai yra priimami.

Taip pat konvergavimui nustatyti galima nubrėžti gautųjų reikšmių iš grandinės grafikus (angl. traceplot). 7 paveiksle atvaizduojami visų sugeneruotų eilučių pirmosios 20000 iteracijų.



7 pav. Microsoft Corporation (kairėje) ir Barclays Bank PLC (dešinėje) iteracijų ir gautųjų reikšmių grafikas, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f)

Stebint 7 paveiksle esančius grafikus galima pasakyti, jog grandinė turėtų konverguoti gana greitai, nes abiejuose grafikuose nepastebimas trendas ar staigūs šuoliai. Taip pat gautosios reikšmės kinta adekvačiuose turimų akcijų grąžų normai esančiuose intervaluose.

Toliau kiekvienai grandinei apskaičiuojamas ir efektyvios imties dydis, t. y. įvertiname nepriklausomų gautų stebėjimų skaičių iš koreliuotos grandinės. Efektyvios imties dydis yra apskaičiuojamas įvertinant koreliaciją tarp gautųjų stebėjimų. Kadangi Metropolio-Hastingso algoritmas esant mažai priėmimo tikimybei kandidatą tašką pakartoja, galima apskaičiuoti grandinės atmetimo santykį, parodantį kiek taškų kandidatų buvo atmesti. Visų grandinių rezultatai yra pateikiami 2 lentelėje.

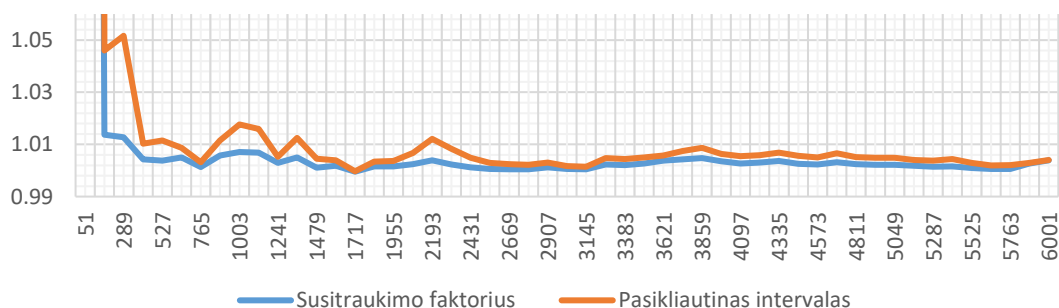
2 lentelė. Markovo grandinių efektyvūs imties dydžiai bei atmetimo santykiai

Akcijos	Grandinės pradinė vertė	Efektyvus imties dydis	Atmetimo santykis
Microsoft Corporation	$X_0 = 0.1$	504428.7	0.1420721
	$X_0 = -0.01$	584317.9	0.1421411
	$X_0 = 0$	617595.7	0.1422031
Barclays Bank PLC	$X_0 = 0.1$	709694.6	0.13465195
	$X_0 = -0.01$	749377.6	0.1345371
	$X_0 = 0$	744029.1	0.1343851

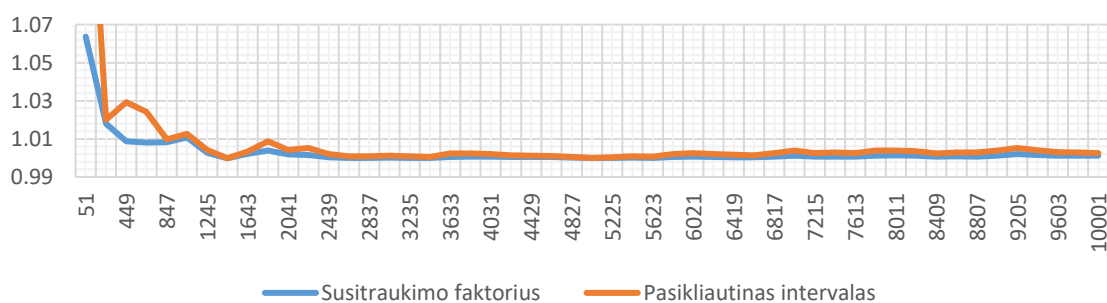
Efektyvus imties dydis visoms grandims yra gana didelis lyginant su bendru Markovo grandinių ilgiu. Atmetimo santykis parodo, jog generuojant Microsoft Corporation akcijų grąžas pasikartojančių elementų dalis grandinėje yra apie 14,21%, tuo tarpu Barclays Bank PLC – apytiksliai 13.45%. Tai gana mažas procentas ir paaiškina mažą autokoreliaciją tarp narių sugeneruotoje grandinėje.

3.2.2. Konvergavimo diagnostikos

Konvergavimui ištirti pasitelkiamos populiariausios konvergavimo diagnostikos. Iš pradžių panaudojant Gelmano bei Rubino diagnostiką įvertiname susitraukimo faktorių.



(a)



(b)

8 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų grąžų konvergavimo tyrimas pasitelkiant Gelmano ir Rubino diagnostiką

Kai grandinė konverguoja, susitraukimo koeficientas mažėja iki 1, tuomet reiškia, kad lygiagrečios Markovo grandinės iš esmės yra sutampančios. Gautieji diagnostikos grafikai yra atvaizduojami 8 paveiksle. Iš jo galima pastebėti, jog abiejoms akcijoms gaunama, jog susitraukimo koeficientas sumažėja iki 1 nepasiekus net 3000 iteracijų.

Kadangi konvergavimo nustatymas yra gana subjektyvus, todėl kiek išmesti pirmųjų iteracijų nebus nustatoma. Panaudojant grafinius metodus bei Gelmano ir Rubino diagnostiką galima manyti, jog grandinė konverguoja gana greitai, todėl tyrime bus išmetami pirmieji 20000 gautųjų stebėjimų iš visų turimų eilučių. Kadangi Geweke ir iš dalies Heidelbergerio ir Welcho diagnostika remiasi vidurkiu ir jo palyginimu grandinėje, diagnostika bus atliekama visoms sugeneruotoms atsitiktinėms imtims, išmetus pirmąsias 20000 iteracijų.

Toliau konvergavimo tyrimui pasitelkiama Geweke diagnostika, kuri yra atliekama visoms šešioms grandinėms. Kadangi sugeneruota pradinė grandinė yra ilga, diagnostika atliekama ir pasirinkus mažesnius fiksuotus grandinės pirmuosius stebėjimus bei išmetus dalį pirmųjų stebėjimų. Grandinės yra skaidomos į dvi dalis, kuriuose yra pirmieji 10% ir paskutiniai 50% iteracijų. Gautosios statistikos reikšmės yra pateikiamos 3 lentelėje.

3 lentelė. Geweke diagnostikos statistikos reikšmės

Akcijos	Grandinės pradinė iteracija	Pirmųjų stebėjimų skaičius					Pašalinus pirmuosius 20000 stebėjimų
		1000000	10000	1000	100	20	
Microsoft Corporation	$X_0 = 0.1$	0.122	0.7262	1.27	98.71	-	0.1353
	$X_0 = -0.01$	-0.9134	0.1153	-1.166	-1.935	-0.0878	-1.117
	$X_0 = 0$	0.2217	0.2093	2.064	-1.648	1.183	0.2748
Barclays Bank PLC	$X_0 = 0.1$	-1.322	0.8884	1.291	1.061	1.003	-0.8894
	$X_0 = -0.01$	1.844	-1.002	1.703	0.07367	-0.1144	1.871
	$X_0 = 0$	-0.0576	0.1907	1.301	-0.2471	0.288	-0.007671

Geweke diagnostika parodo, jog stacionarumas pasiekiamas, kai ji yra panaudojama dideliame imties skaičiui. Jei konvergavimas yra greitas, netinkami pirmieji stebėjimai yra suvidurkinami, todėl šią diagnostiką derėtų taikyti ir mažesnėms imtims. Pašalinus 20000 pirmųjų stebėjimų diagnostika nurodo, jog gautosios grandinės pirmųjų 10% ir paskutiniųjų 50% iteracijų stebėjimų vidurkiai eilutėje yra panašūs.

Toliau vykdoma Heidelbergerio ir Welcho konvergavimo diagnostika abiejų akcijų gražų grandinėms, o rezultatai pateikiami 4 lentelėje.

4 lentelė. Heidelbergerio ir Welcho konvergavimo diagnostikos rezultatai

Akcijos	Pradinė iteracijos reikšmė	Stacionarumo testas	Pusės pločio testas		
			Būsena	Vidurkis	Pusės intervalo plotis
Microsoft Corporation	$X_0 = 0.1$	Įvykdomas	Įvykdomas	0.00149	2.73e-05
	$X_0 = -0.01$	Įvykdomas	Įvykdomas	0.00147	2.81e-05
	$X_0 = 0$	Įvykdomas	Įvykdomas	0.00148	2.79e-05
Barclays Bank PLC	$X_0 = 0.1$	Įvykdomas	Įvykdomas	-0.000473	8.23e-06
	$X_0 = -0.01$	Įvykdomas	Įvykdomas	-0.000463	7.97e-06
	$X_0 = 0$	Įvykdomas	Įvykdomas	-0.000464	8.02e-06

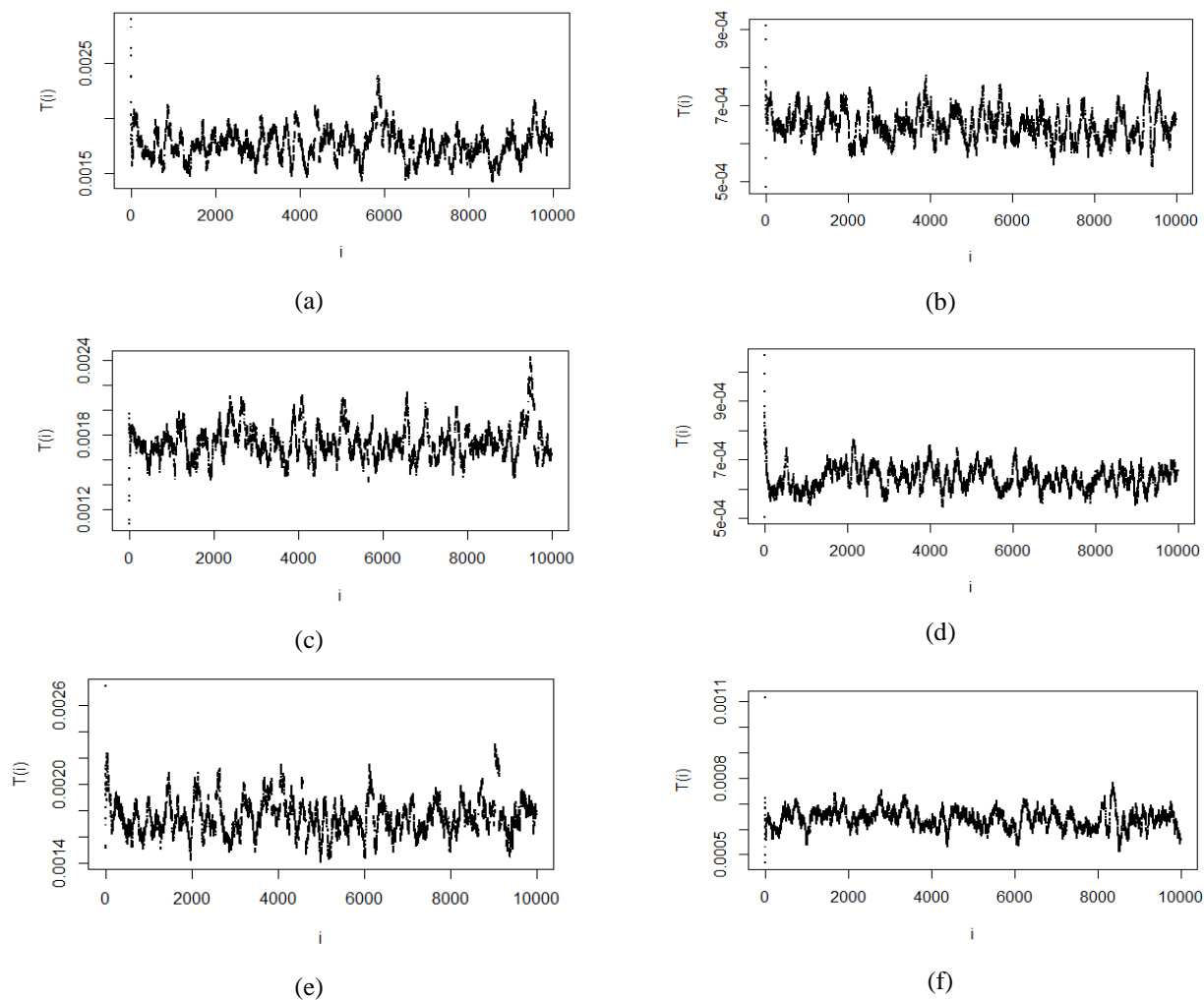
Naudojant Heidelbergerio ir Welcho konvergavimo diagnostiką gaunama, jog visos grandinės įvykdo stacionarumo testą. Kai testas yra įvykdomas, pereinama prie pusės pločio testo, kurio metu pusė šio intervalo pločio yra lyginama su vidurkio įverčiu. Kadangi gautieji santykiai tarp šių dydžių yra labai maži, testai yra išlaikomi.

3.2.3. Konvergavimas panaudojant tikslo funkciją

Toliau nagrinėjama teorinių bei sugeneruotų Metropolio-Hastingso būdu stebėjimų pradinių bei centrinių momentų skirtumai nepašalinant nei vieno pradinio elemento. Momentai suskaičiuojami iš 100 turimų stebėjimų, kas 100 iteracijų jie perskaičiuojami. Funkcija, priklausanti nuo momentų skirtumo, yra apibrėžiama taip:

$$T(i) = \sum_{j=1}^3 |\hat{\mu}_{j; Y_i} - \mu_j| + \sum_{j=1}^3 |\hat{\mu}'_{j; Y_i} - \mu'_j|.$$

Gautosioms reikšmėms suskaičiuojamas paprastas slenkantis vidurkis. Konverguojant eilutei skirtumai tarp momentų turėtų būti kuo panašesni į sugeneruotuosius momentus, todėl jų skirtumai turi mažėti arba kisti tam tikrose ribose. Rezultatai atvaizduojami 9 paveiksle.



9 pav. Microsoft Corporation ir Barclays Bank PLC akcijų gražų tikslo funkcijų reikšmių grafikas pritaikius slenkantį vidurkį, kai pradiniai taškai yra $X_0 = 0.1$ (a, b); $X_0 = -0.01$ (c, d); $X_0 = 0$ (e, f)

Stebint 9 paveiklo grafikus galima pastebėti, kad tikslo funkcijos reikšmės kinta tolygiai. Iš gautųjų rezultatų yra sunku nustatyti, kada konverguoja eilutės, nes gautosios tikslo funkcijos reikšmės yra ganėtinai mažos, todėl tokį konvergavimo tyrimą vertėtų bandyti atlikti vykstant lėtam konvergavimui. Kadangi akcijų vertei prognozuoti bus pasirenkama viena turima eilutė iš turimų trijų, todėl pasirenkamos tos akcijų gražos iš grandinės, turint mažiausią gautųjų tikslo funkcijų verčių vidurkį (sakome, kad mažesni skirtumai tarp momentų geriau apibūdina empirinius duomenis), nevertinant pirmųjų gautųjų 20000 iteracijų. Gaunami tikslo funkcijų reikšmių vidurkiai pateikiami 5 lentelėje.

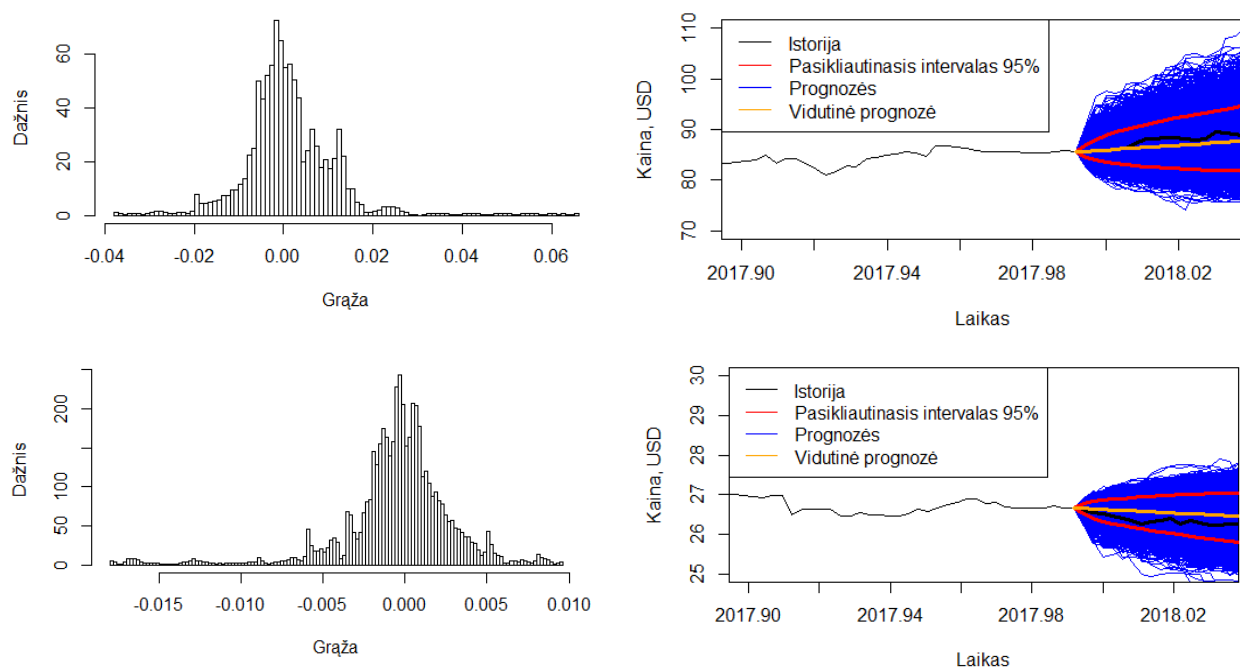
5 lentelė. Tikslo funkcijų reikšmių vidurkiai

Pradinė reikšmė	Vidurkiai	
	Microsoft Corporation	Barclays Bank PLC
$X_0 = 0.1$	0.001759467	0.0006476981
$X_0 = -0.01$	0.001737242	0.0006373095
$X_0 = 0$	0.001748309	0.0006391929

Kadangi mažiausias skirtumas tarp pradinių bei centrinių momentų tiek Microsoft Corporation, tiek Barclays Bank PLC akcijų gražose yra antrosios sekos, todėl šios akcijų gražos ir bus panaudojamos prognozuojant būsimas akcijų vertes.

3.3. Prognozavimas Markovo grandinių Monte Karlo metodu

Siekiant, kad turimos atsitiktinės gražos būtų kuo įvairesnės, todėl iš pasirinktų sekų (prieš tai išmetus 20000 iteracijų) papildomai yra pasirenkamas kas trečias elementas ir visa turima imtis yra sumaišoma. Akcijų vertė prognozuojama iki 2018 m. sausio 16 d. ir 15000 trajektorių bei pasirinktų sugeneruotų akcijų gražų histograma ir atvaizduojame 10 paveiksle.

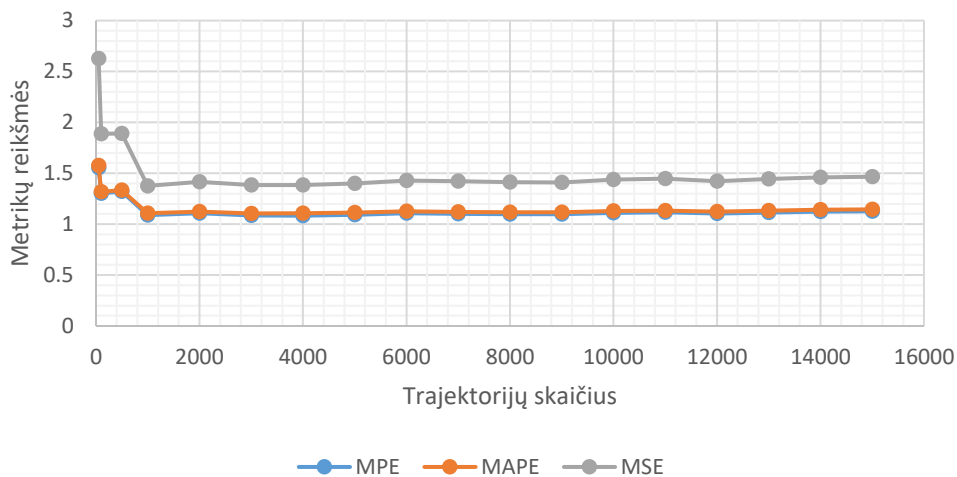


10 pav. Microsoft Corporation (viršuje) ir Barclays Bank PLC (apačioje) akcijų gražų histograma ir jų kainų prognozė MCMC metodu

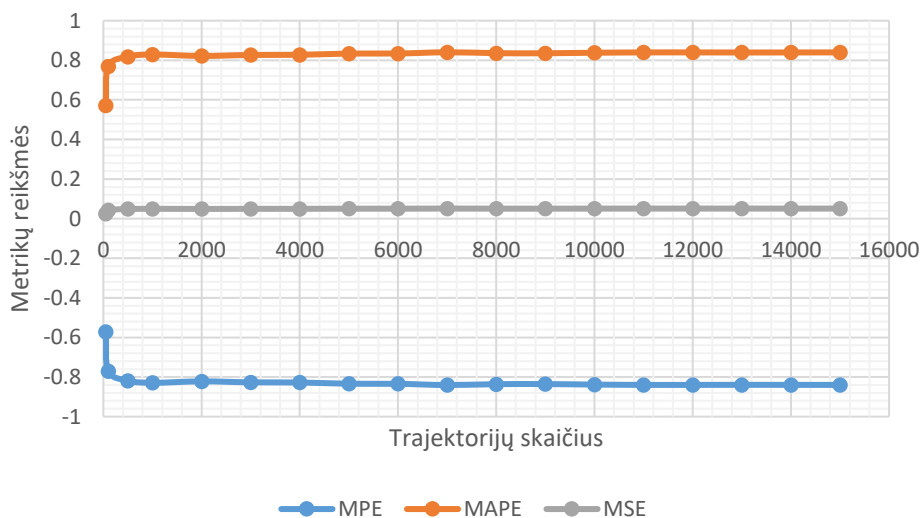
Lyginant turimų akcijų gražų histogramą, pavaizduotą 2 paveiksle, su gautosiomis 10 paveiksle, galima spręsti, kad siūlomas dalimis tolydus skirstinys yra tinkamas generuoti taškus kandidatus, nes gaunamos labai panašios formos histogramos.

Stebint 10 paveikslo vidutines kainų prognozes ir lyginant su buvusiomis 2018 metų istorinėmis kainomis galima pastebėti, kad abiejų akcijų prognozavimas MCMC metodu yra gana tikslingas, nes tikroji Microsoft Corporation akcijų vertė didėjo, kai suprognozuota vidutinė prognozė irgi turi tendą didėti, o tuo tarpu mažėjant Barclays Bank PLC akcijų vertei, MCMC metodu prognozuojama akcijų vertė taip pat mažėja.

Gautos vidutinės prognozės palyginimui su turima akcijų vertės istorija yra panaudojamos prognozės tikslumo metrikos. Šios metrikos yra apskaičiuojamos skirtingam trajektorijų skaičiui, o gautieji rezultatai yra pateikiami 11 paveiksle.



(a)



(b)

11 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų prognozavimo metrikų priklausomybė nuo trajektorijų skaičiaus

Vertinant gautųjų metrikų rezultatus iš 11 paveikslo galima pastebėti, kad prognozuojant Microsoft Corporation akcijų vertes, tikslumo matai nusistovi po maždaug 2000 trajektorijų, o didinant jų skaičių šių parametrų reikšmės nežymiai kinta. Tuo tarpu vertinant Barclays Bank PLC akcijų verčių paklaidų rodiklius, nusistovėjimas atsiranda po 1000 trajektorijų.

3.4. Prognozavimas Monte Karlo metodu

Klasikinis Monte Karlo modelis reikalauja, jog gražos būtų pasiskirsčiusios pagal normalųjį dėsnį. Siekiant patikrinti gražų normalumą, visiems duomenims atliekami šeši normalumo testai. Gautieji rezultatai pateikiami 6 lentelėje.

6 lentelė. Microsoft Corporation ir Barclays Bank PLC akcijų gražų normalumo testai

Normalumo testai	Akcijos			
	Microsoft Corporation		Barclays Bank PLC	
	Statistikos reikšmė	p-reikšmė	Statistikos reikšmė	p-reikšmė
Shapiro-Wilk	0.91855	1.864e-10	0.86537	5.265e-14
Shapiro-Francia	0.90939	7.349e-10	0.86021	1.736e-12
Cramer-von Mises	0.50843	2.253e-06	1.101	7.37e-10
Lilliefors	0.087521	8.314e-05	0.13929	9.303e-13
Anderson-Darling	2.9468	2.013e-07	6.4954	5.638e-16
Pearson chi kadrato	35.304	0.003611	79.232	2.288e-10

Atliekus testus visais atvejais gaunamos labai mažos p reikšmės, todėl galima daryti išvadą, jog duomenys nėra pasiskirstę pagal normalųjį dėsnį, o tai reiškia, kad prognozuoti Monte Karlo metodu būtų netikslinga.

Kadangi turima imtis nėra pasiskirčiusi pagal normalųjį dėsnį, todėl daroma prielaida, jog galbūt dalis duomenų tenkins normalumo sąlygas. Vadinasi, tikrinsime normalumo sąlygas suskaičiuotoms gražoms nuo 2017m. lapkričio 9 d. iki 2017m. gruodžio 29 d. (paskutiniams 34 duomenims). Gautieji rezultatai pateikiami 7 lentelėje.

7 lentelė. Microsoft Corporation ir Barclays Bank PLC akcijų gražų normalumo testai duomenims nuo 2017m. lapkričio 9 d. iki iki 2017m. gruodžio 29 d.

Normalumo testai	Akcijos			
	Microsoft Corporation		Barclays Bank PLC	
	Statistikos reikšmė	p reikšmė	Statistikos reikšmė	p reikšmė
Shapiro-Wilk	0.92886	0.02899	0.86173	0.0005177
Shapiro-Francia	0.9114	0.01143	0.83926	0.0005177
Cramer-von Mises	0.12333	0.05114	0.12111	0.05485
Lilliefors	0.12349	0.2094	0.13304	0.1322
Anderson-Darling	0.73184	0.05109	0.92271	0.01684
Pearson chi kvadrato	8.8824	0.1803	9.4118	0.1517

Iš 7 lentelės galima pamatyti, jog atliekant normalumo testus 34 duomenims, dalis jų parodo, jog duomenys yra pasiskirstę pagal normalųjį dėsnį. Gautsios p reikšmės turi būti didesnės už 0.05, jog hipotezė apie normalumą būtų priimtina.

Tikrinant Microsoft Corporation akcijų gražų duomenis gaunama, jog jos tenkina 4 iš 6 testų (tenkinami Cramer-von Mises, Lilliefors, Anderson-Darling bei Pearson chi kvadrato testai), o tuo tarpu Barclays Bank PLC duomenys tenkina 3 iš 6 testų (tenkinami Cramer-von Mises, Lilliefors bei Pearson chi kvadrato testai). Kadangi dalis testų yra priimtini, laikysime, kad duomenys yra pasiskirstę pagal normalųjį dėsnį.

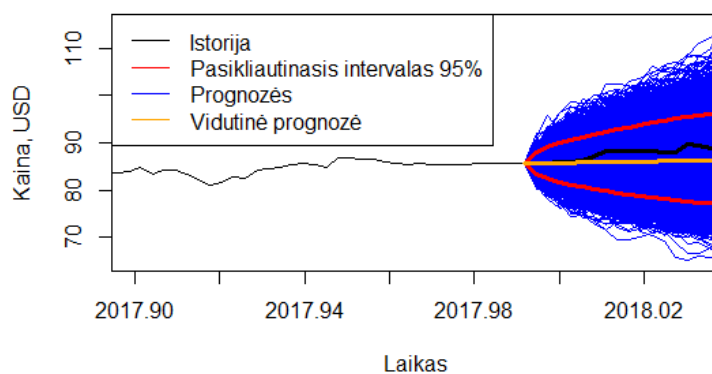
Apskaičiuojame Monte Karlo akcijų vertėms prognozuoti modelio parametrus:

$$\begin{aligned}\sigma_{MC} &= 0.01819553, & \sigma_{BB} &= 0.005036163, \\ \delta_{MC} &= 0.0005028432, & \delta_{BB} &= -0.00001094584.\end{aligned}$$

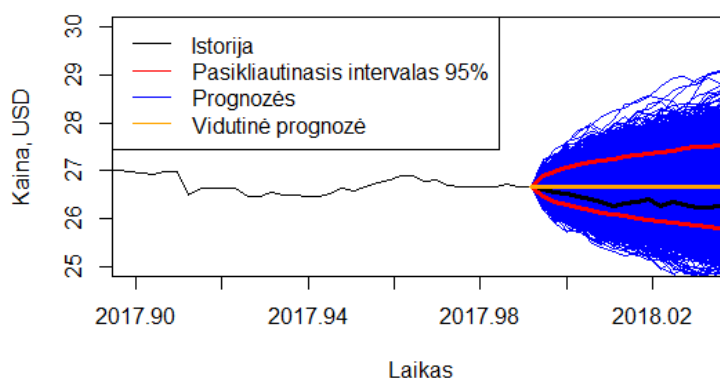
Akcijų vertės prognozuojamos pagal modelį

$$S(t+1) = S(t)e^{(\delta + \frac{1}{2}\sigma^2) + \sigma Z},$$

vietoj σ bei δ įstačius atitinkamas parametrų reikšmes kiekvienai akcijai. Pagal turimus modelius sugeneruojama 200 trajektorijų ir gautieji rezultatai atvaizduojami 12 paveiksle.



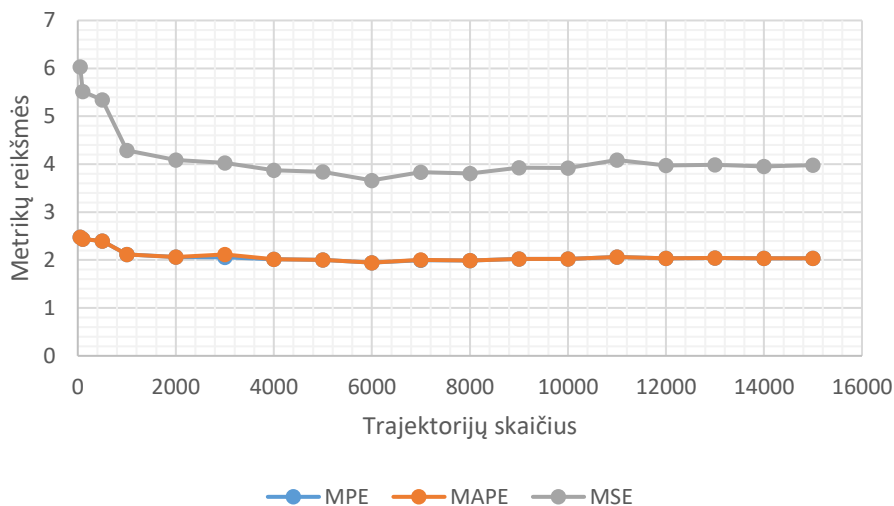
(a)



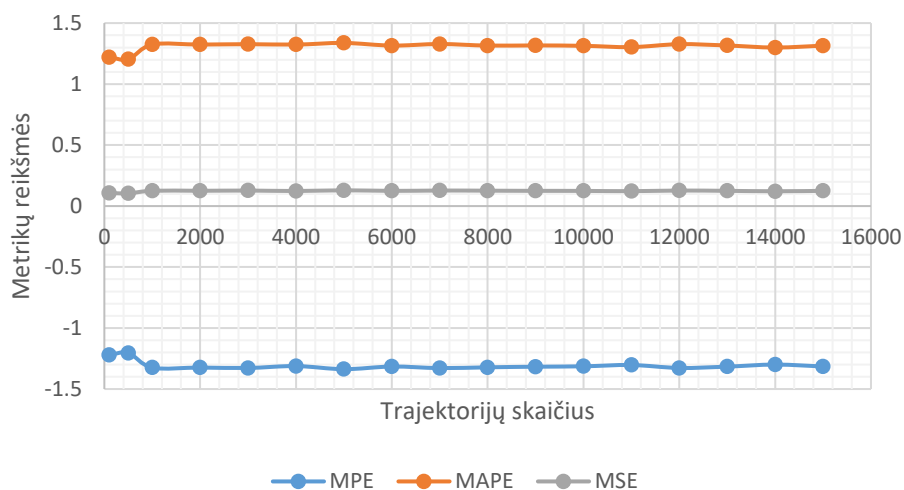
(b)

12 pav. Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų kainų prognozė Monte Karlo modeliu

Kaip ir Microsoft Corporation akcijų kainų su turima istorija palyginime, Barclays Bank akcijoms taip pat apskaičiuojami tikslumo matai, pateikti 13 paveiksle. Gautųjų metrikų rezultatai abejose akcijų vertėse nusistovi po 5000 iteracijų. Lyginant su MCMC modelio gautomis vertėmis (11 paveikslas) galima daryti išvadą, kad MCMC modelio metu gaunamos tikslesnės akcijų vertės.



(a)

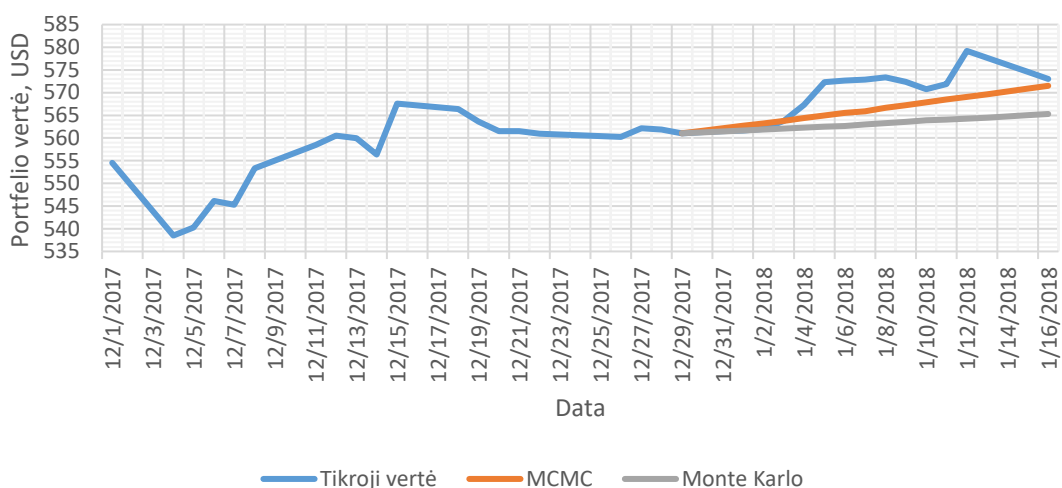


(b)

13 pav. Klasikinio Monte Karlo metodo Microsoft Corporation (a) ir Barclays Bank PLC (b) akcijų prognozavimo tikslumų matų priklausomybė nuo generuojamų trajektorijų skaičiaus

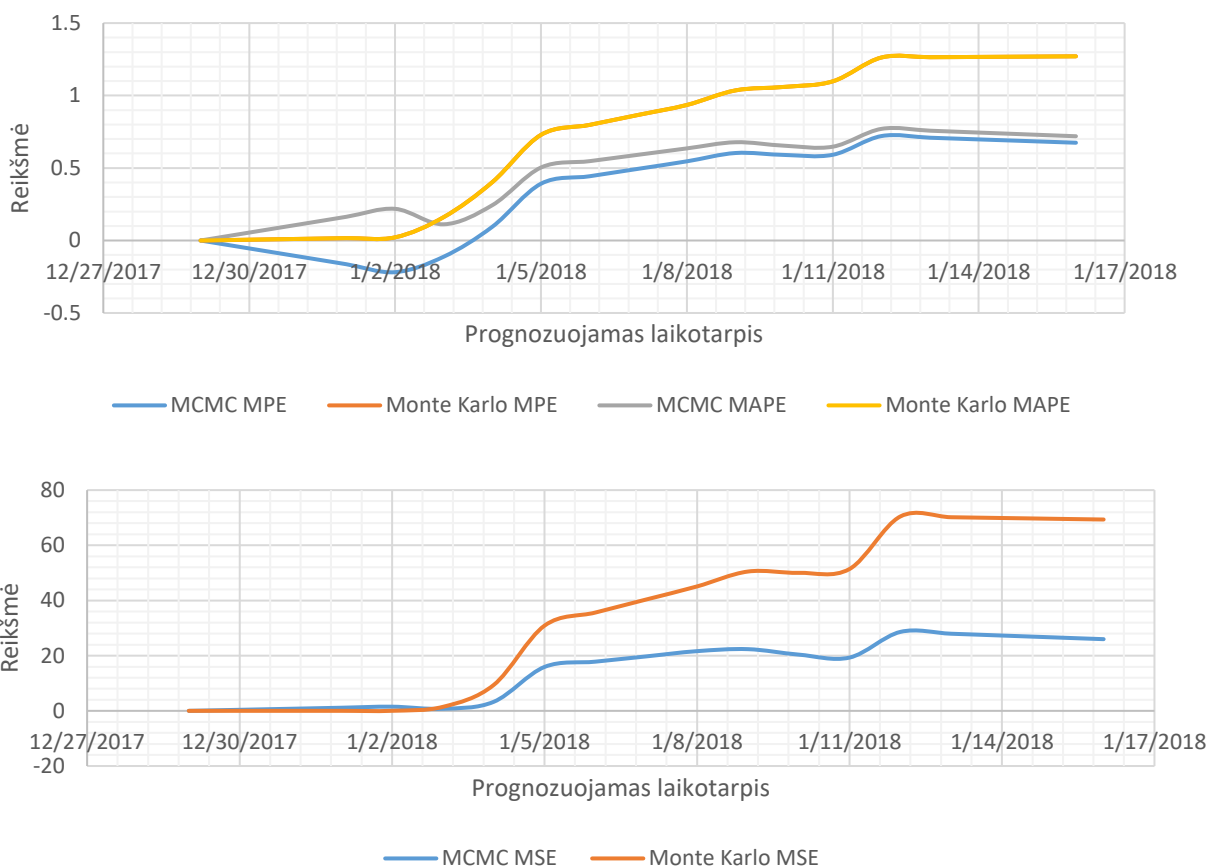
3.5. Investicinio portfelio sudarymas

Kadangi turimą investicinį portfelį sudaro po penkis vienetus tiek Microsoft Corporation, tiek Barclays Bank PLC akcijų, palyginsime gautąsias investicinio portfelio vertes MCMC bei Monte Karlo metodu su faktine verte. Gautojų portfelio vertės istorija ir gautoji vertė panaudojant abu modelius yra atvaizduojama 14 paveiksle, panaudojus vidutines prognozuojamas akcijų vertes sugeneravus 15000 trajektorijų.



14 pav. Investicinio portfelio istorinė ir prognozuojamos vertės

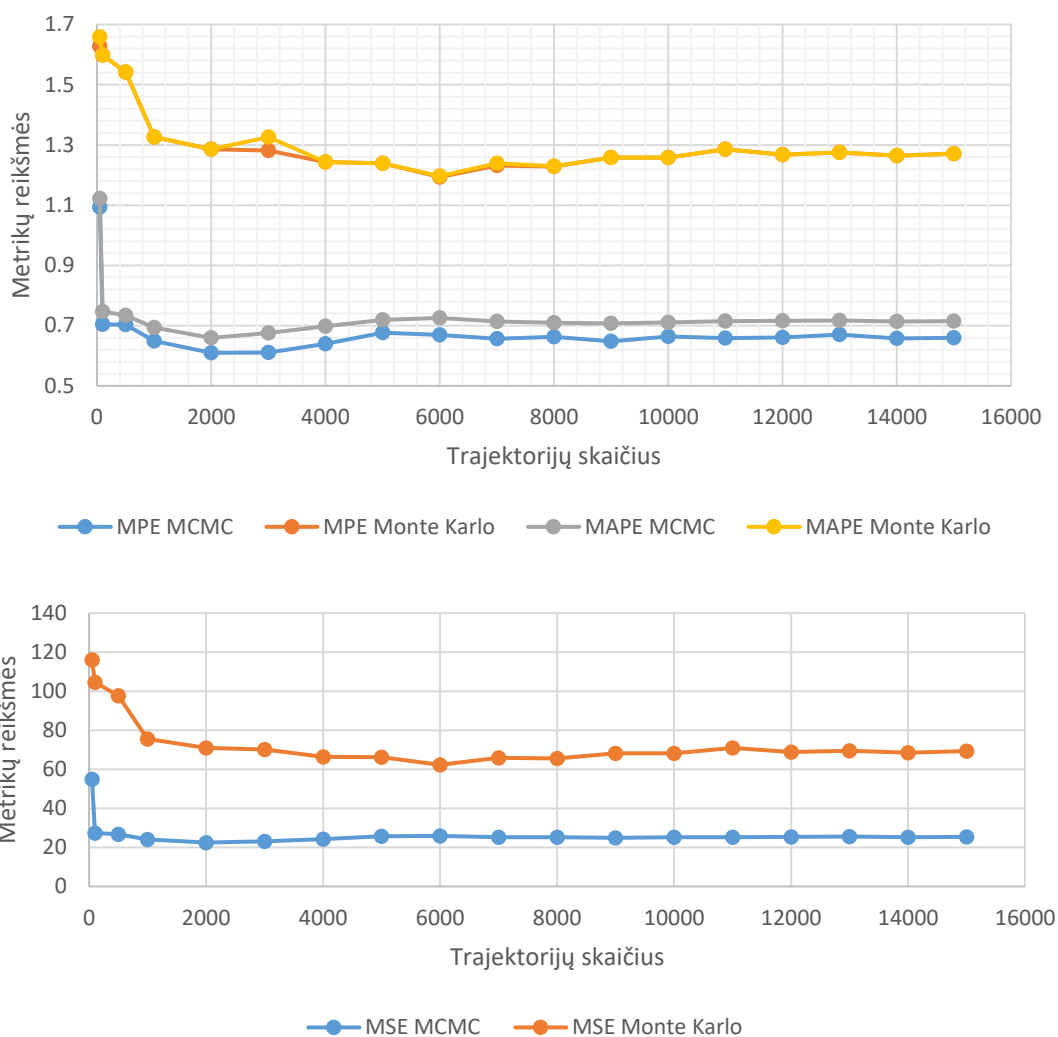
14 paveiksle pavaizduota istorinė bei prognozuojamos Monte Karlo ir MCMC metodu portfelio vertės. Prognozavimas šiais modeliais yra gana tikslingas, nes gautosios vertės prognozės turi istorijai atitinkantį tendą didėti. Kadangi akcijų verčių prognozavimas MCMC buvo tikslesnis, todėl ir investicinio portfelio vertės gautoji trajektorija yra arčiau istorinių portfelio verčių prognozuojant MCMC metodu. Tam įsitikinti bus įvertinama prognozavimo metrikų reikšmių priklausomybė nuo prognozuojamo laikotarpio. Gautieji rezultatai yra atvaizduojami 15 paveiksle.



15 pav. Investicinio portfelio vertės tikslumo priklausomybės nuo prognozuojamo laikotarpio

Nagrinėjant investicinio portfelio vertės tikslumo priklausomybę nuo prognozuojamo laikotarpio galima pastebėti, kad klasikinis Monte Karlo metodas prognozuoja pirmuosius laikotarpius šiek tiek tiksliau, tačiau prognozavimo laikotarpiui didėjant, MCMC metodu prognozuojama investicinio portfelio vertė yra tikslesnė.

Toliau nagrinėjama prognozės tikslumo metrikų priklausomybė nuo generuojamų trajektorijų skaičiaus. Vertinant šią priklausomybę galima apytiksliai nustatyti, kiek trajektorijų pakanka, kad tikslumo matai nusistovėtų. Gautieji rezultatai pavaizduoti 16 paveiksle.



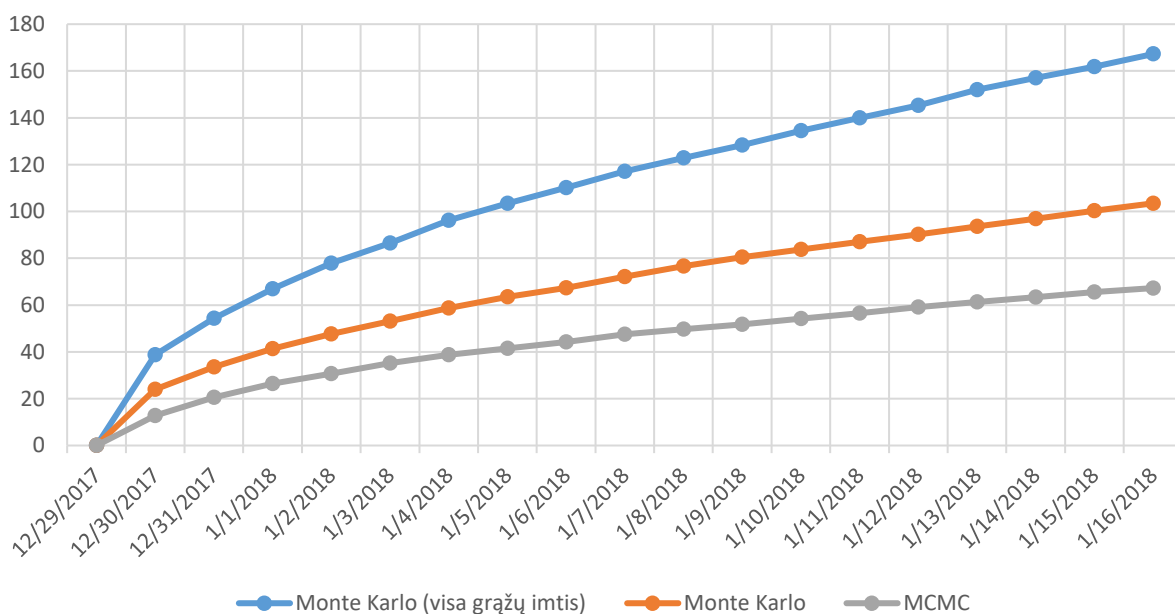
16 pav. Investicinio portfelio vertės tikslumo matų priklausomybė nuo trajektorijų skaičiaus

Visais atvejais yra gaunama, kad MCMC metodu gauto investicinio portfelio vertė yra žymiai tikslesnė nei prognozuojant klasikiniu Monte Karlo metodu, o paklaidos mažesnės net esant labai mažam trajektorijų skaičiui. Tikslumo matų nusistovėjimas abejais metodais pastebimas apytiksliai po 5000 iteracijų.

Reikia pabrėžti, jog klasikinis Monte Karlo modelis reikalauja, kad akcijų grąžos būtų pasiskirsčiusios pagal normalųjį dėsnį. Kadangi šis reikalavimas nėra išpildomas vertinant visą

nagrinėjamą akcijų gražų imtį, todėl gali būti daroma prielaida, kad Monte Karlo metodas blogiau prognozuoja akcijų vertes, nes modelio parametrai buvo įvertinti iš mažesnės gražų imties. Bet kokių atveju taip išryškėja šio modelio trūkumas, o siūlomas MCMC metodas nėra apribojamas tokia sąlyga.

Toliau bus tiriamas gautųjų investicinio portfelio prognozuotos vertės trajektorijų išsibarstymas. Tam tikslui bus panaudojami pasikliautiniai intervalai su 95% pasiklovimo tikimybe, apskaičiuoti iš sugeneruotų 15000 trajektorijų, parodantys tikėtinos vertės svyravimą. Dėl plataus pasikliautino intervalo negalima būti tikriems dėl tikrosios prognozuojamos vertės, kadangi intervalas su pasirinkta tikimybe parodo, kiek ši vertė gali svyruoti. Mažesnis pasikliautino intervalo plotis nusako siauresnę gautojo investicinio portfelio trajektorijų verčių pasiskirtymą bei dispersiją. Šio pločio priklausomybė nuo prognozuojamo laikotarpio abiems modeliams ir klasikiniui modeliui, įvertinant visą turimą akcijų verčių imtį (kuri netenkina normalumo sąlygų ir naudojama MCMC modelyje), yra atvaizduojama 17 paveiksle.



17 pav. Modelių pasikliautinių intervalų pločio priklausomybė nuo prognozuojamo laikotarpio

Lyginant Monte Karlo modelį panaudojant visą ir dalį akcijų gražų galima pastebėti, kad naudojant gražas, netenkinančias normalumo sąlygų, gaunamas platesnis investicinio portfelio vertės trajektorijų išsibarstymas nei panaudojus tik tas gražas, tenkinančias normalumą. Vadinasi, didesnės imties naudojimas klasikiniame modelyje netenkinant normalumo sąlygų, investicinio portfelio tikėtina vertė su 95% pasiklovimo tikimybe svytuos platesnėse ribose. Prognozuojant šią vertę panaudojus akcijų gražas, gautas MCMC metodu, gaunamas siauresnis tikėtinos portfelio vertės intervalas.

Išvados

- racionaliosios Čebyševio aproksimacijos eilė yra pagrindinis faktorius, apibrėžiantis tikslinį skirstinį. Ieškant optimalios šio parametro vertės buvo pastebėta, kad vertinant pradinius bei centrinius skirtumus tarp empirinių ir pradinių momentų, bendruoju atveju, atsitiktinių svorių pasirinkimas nėra tinkamas, nes keičiant šiuos parametrus nėra gaunamas unikalus rezultatas, todėl reikalinga išbandyti įvairius šių parametrų variantus;
- dėl savo paprastumo, Rstudio programinis paketas yra patogus realizuoti Markovo grandinių Monte Karlo metodą, atlikti ortogonalinių eilučių įverčio aproksimaciją ir tirti sugeneruotos grandinės konvergavimą, nes *coda* plėtinyje yra pateikiamos populiariausios konvergavimo diagnostikos. Nors tokiam tyrimui atlikti plėtinyje yra realizuota nemažai lengvai naudojamų funkcijų, tačiau vertinant iš praktinės pusės, siekiant greitai įvertinti dominančias akcijas, didelis atsitiktinių akcijų gražų generavimo kiekis užima daug laiko, ypač jei konvergavimo diagnostikos metu yra pasitelkiama keletas ilgų, lygiagrečiai sugeneruotų, atsitiktinių sekų;
- panaudojant įvairias diagnostikas ir metodus buvo nagrinėjamas gautosios Markovo grandinės konvergavimas. Taikant grafines priemones bei Gelmano ir Rubino diagnostiką pastebėta, kad atsitiktinės gražų normos greitai konverguoja, todėl dalimis tolydus skirstinys yra tinkamas generuoti taškus kandidatus. Taip pat gautasis didelis efektyvios imties dydis bei žemas atmetimo santykis patvirtina siūlomo skirstinio tinkamumą. Papildomai panaudojus Geweke bei Heidelbergerio ir Welcho diagnostiką buvo gauta, jog visos nagrinėtos grandinės pasiekia stacionarumą; panaudojus sudarytą tikslo funkciją, nustatyta, kad grandinės konvergavimą fiksuoti sudėtinga, ypač jei grandinės konvergavimas yra greitas. Teoriškai, konverguojant eilutei, skirtumai tarp momentų turi mažėti, todėl tokią metodiką vertėtų išbandyti esant lėtam konvergavimui. Kita vertus, kiekvienai nagrinėjamai akcijai buvo sugeneruota po tris akcijų gražų grandines, todėl ši funkcija gali pasitarnauti pasirenkant vieną iš grandinių įvertinus jos vidurkį (ar kitą matą, pavyzdžiui, sumą);
- klasikinis Monte Karlo modelis reikalauja, kad akcijų gražos būtų pasiskirsčiusios pagal normalųjį dėsnį. Kadangi ši sąlyga nėra išpildoma, todėl modelis buvo taikomas tik vertinant gražas mažesnei daliai imties, tenkinančias normalumą. Lyginant klaisikinio Monte Karlo bei Markovo grandinių Monte Karlo metodų prognozavimo tikslumo metrikas buvo pastebėta, kad naudojant pastarąjį, gaunama tikslesnė abiejų nagrinėtų akcijų bei investicinio portfelio vertė; lyginant Monte Karlo metodo metu gautus pasikliautinuosius intervalus buvo gauta, kad didesnės imties akcijų gražų naudojimas nebūtinai parodys siauresnį vertės svyravimą, ypač

kai normalumo sąlygos yra netenkinamos. Mažiausias pasiklautinas tikėtinos portfelio vertės plotis buvo gautas prognozuojant akcijų vertę panaudojus grąžas, sugeneruotas Markovo grandinių Monte Karlo metodu.

Diskusija

Šiame skyriuje pateikiami keletas bendrųjų pastebėjimų, atsiradusių darbo eigoje, kuriuos panaudojant investicinio portfelio vertę galėjo būti nustatyta kitaip. Keletas pastebėjimų ir alternatyvų:

- apksimuojant akcijų gražų histogramą buvo panaudoti Čebyševio polinomiali. Tankis gali būti įvertintas išbandant ir kitokius ortogonalius polinomus. Gali būti, kad naudojant kitus polinomus ir tą pačią tikslo funkciją bus gaunama kitokia optimali aproksimacijos eilė;
- šiame darbe ieškant aproksimacijos eilės buvo panaudota tikslo funkcija, vertinanti skirtumus tarp pirmų trijų pradinių bei centrinių momentų. Kaip alternatyva šiai tikslo funkcijai gali būti vertinami dar ir aukštesnės eilės momentų skirtumai ir (arba) jų santykis;
- verta paminėti, kad gautoji aproksimacijos eilė apibūdina akcijų gražas, tačiau būtų naudinga išnagrinėti ir gautojo investicinio portfelio vertės priklausomybę nuo aproksimacijos eilės. Nors aukštesnė aproksimacijos eilė pagal pasirinktą tikslo funkciją tiksliau neapibūdins turimų akcijų gražų, tačiau gali būti, kad prognozuojant investicinio portfelio vertę bus gaunami tikslesni rezultatai. Tokios priklausomybės nagrinėjimas reikalauja didesnių kompiuterinių resursų, nes skaičiavimas vyks ilgai, priklausomai nuo to, kokio ilgio atsitiktinės sekos bus generuojamos. Šiam tikslui įgyvendinti vertėtų rinktis greitesnę ir praktiškesnę programinę įrangą;
- tinkamas siūlomas skirtinys – svarbus MCMC algoritmo elementas. Darbe buvo panaudojamas dalimis tolydus skirstinys, tačiau generuoti taškus kandidatus būtų galima ir iš kito, pavyzdžiui, normaliojo skirstinio, kurio forma yra panaši į turimų gražų histogramą. Kita vertus, naudojant skirstinius, turinčius begalinę apibrėžimo sritį, MCMC algoritmo metu bus generuojami taškai kandidatai ir iš šio skirstinio uodegų. Tokie taškai bus atmetami dėl mažos priėmimo tikimybės, nes ortogonalinių eilučių tankio įvertis apibūdina duomenis baigtiniame intervale. Tokiu būdu gautosios grandinės akcijų gražos bus labiau koreliuotos;
- vertinant praktinę naudojamų modelių pusę verta paminėti, kad dėl akcijų rinkos nepastovumo ir pokyčių, investuotojas gali būti priverstas greitai priimti sprendimus parduodant ar įsigijant akcijas. Taigi, prognozuojant portfelio vertę, klasikinis Monte Karlo metodas yra paprastesnis ir realizuojamas greičiau, tuo tarpu siūlomas akcijų gražų generavimas Markovo grandinių Monte Karlo metodu pareikalaus ilgesnių laiko sąnaudų, nes jo metu atliekama išsamesnė analizė. Šiam algoritmui įgyvendinti reikalinga įvertinti ne tik tikslinį ir siūlomą skirtinius, bet ir pasirinkti pradines grandinės reikšmes, nuo kurios pradedamos generuoti akcijų gražos, bei nustatyti konvergavimą. Toks vykdymas gali būti paspartinamas ir supaprastinamas pasirinkus bet kokį siūlomą skirtinį, kurio tinkamumo įvertinimui užtektų panaudoti atmetimo

santyki, gautųjų reikšmių grafiką ir autokoreliacinę funkciją. Taip pat pasirinkus pradinis iteracinius taškus, adekvačius akcijų gražoms, stacionarumas turėtų būti greitai pasiekiamas, todėl neanalizavus konvergavimo, būtų galima išmesti pirmuosius keletą tūkstančių iteracijų.

Literatūros sąrašas

- [1] S. Dupernex, „Why Might Share Prices Follow a Random Walk?“, *Student Economic Review*, t. 21, pp. 167-179, 2007.
- [2] B. G. Malkiel, „The Efficient Market Hypothesis and Its Critics“, *Journal of Economic Perspectives*, t. 17, nr. 1, pp. 59-82, 2003.
- [3] N. Petrusheva ir I. Jordanoski, „Comparative Analysis Between the Fundamental and Technical Analysis of Stocks“, *Journal of Process Management - New Technologies, International*, t. 4, nr. 2, pp. 26-31, 2016.
- [4] W. K. Hastings, „Monte Carlo Sampling Methods Using Markov Chains and Their Applications“, *Biometrika*, t. 57, nr. 1, pp. 97-109, 1970.
- [5] D. Hatem ir M. Maima, „On the Random Walk Metropolis Algorithm“, Gaza, 2015.
- [6] L. A. Breyer, M. Piccioni ir S. Scarlatti, „Optimal scaling of MALA for nonlinear regression“, *The Annals of Applied Probability*, t. 14, nr. 3, pp. 1479-1505, 2004.
- [7] L. Tierney, „Markov Chains for Exploring Posterior Distributions“, *The Annals of Statistics*, t. 22, nr. 4, pp. 1701-1728, 1994.
- [8] S. F. Jarner ir G. O. Roberts, „Convergence of Heavy-tailed Monte Carlo Markov Chain Algorithms“, *Scandinavian Journal of Statistics*, t. 34, nr. 4, pp. 781-815, 2007.
- [9] F. Mitha, „Perfect Sampling on Continuous State Space“, University of North Carolina, Chapel Hill, 2003.
- [10] S. Geman ir D. Geman, „Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images“, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 6, nr. 6, pp. 721-741, 1984.
- [11] C. Robert ir G. Casella, „Monte, A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data“, *Statistical Science*, t. 26, nr. 1, pp. 102-115, 2011.
- [12] G. O. Roberts ir J. S. Rosenthal, „Examples of Adaptive MCMC“, *Journal of Computational and Graphical Statistics*, t. 18, nr. 2, pp. 349-367, 2009.
- [13] M. Vihola, „Robust adaptive Metropolis algorithm with coerced acceptance rate“, *Statistics and Computing*, t. 22, nr. 5, p. 997-1008, 2012.
- [14] C. P. Robert ir G. Casella, *Introducing Monte Carlo Methods with R*, London: Springer, 2010.
- [15] A. Gelman, W. R. Gilks ir G. O. Roberts, „Weak convergence and optimal scaling of random walk Metropolis algorithms“, *The Annals of Applied Probability*, t. 7, nr. 1, pp. 110-120, 1997.
- [16] G. O. Roberts ir J. S. Rosenthal, „Optimal Scaling for Various Metropolis-Hastings Algorithms“, *Statistical Science*, t. 16, nr. 4, pp. 351-367, 2001.
- [17] A. Gelman ir D. B. Rubin, „Inference from Iterative Simulation Using Multiple Sequences“, *Statist. Sci.*, t. 7, nr. 4, pp. 457-472, 1992.
- [18] S. P. Brooks, „Markov Chain Monte Carlo Method and Its Application“, *Journal of the Royal Statistical Society. Series D (The Statistician)*, t. 47, nr. 1, pp. 69-100, 1998.
- [19] C. J. Geyer, „Practical Markov Chain Monte Carlo“, *Statistical Science*, t. 7, nr. 4, pp. 473-511, 1992.

- [20] S. P. Brooks ir A. Gelman, „General Methods for Monitoring Convergence of Iterative Simulations,“ *Journal of Computational and Graphical Statistics*, t. 7, nr. 4, pp. 434-455, 1998.
- [21] J. Geweke, „Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments,“ *Bayesian Statistics 4*, pp. 169-193, 1992.
- [22] P. Heidelberger ir P. D. Welch, „Simulation Run Length Control in the Presence of an Initial Transient,“ *Operations Research*, pp. 1109-1144, 1989.
- [23] A. E. Raftery ir S. Lewis, „[Practical Markov Chain Monte Carlo]: Comment: One Long Run with Diagnostics: Implementation Strategies for Markov Chain Monte Carlo,“ *Statistical Science*, t. 7, nr. 4, pp. 493-497, 1992.
- [24] S. T. Garren ir R. L. Smith, „Estimating the second largest eigenvalue of a Markov transition matrix,“ *Bernoulli*, t. 6, nr. 2, pp. 215-242, 2000.
- [25] M. K. Cowles ir B. P. Carlin, „Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review,“ *Journal of the American Statistical Association*, t. 91, nr. 434, pp. 883-904, 1994.
- [26] W.-J. Beyn, L. Dieci, N. Guglielmi, E. Hairer, J. M. Sanz-Serna ir M. Zennaro, *Current Challenges in Stability Issues for Numerical Differential Equations*, Cetraro: Springer, 2011.
- [27] B. Ballnus, S. Hug, K. Hatz, L. Görlitz, J. Hasenauer ir F. J. Theis, „Comprehensive benchmarking of Markov chain Monte Carlo methods for dynamical systems,“ *BMC Systems Biology*, t. 11, nr. 1, 2017.
- [28] O. Malve, M. Laine ir H. Haario, „Estimation of winter respiration rates and prediction of oxygen regime in a lake using Bayesian inference,“ *Ecological Modelling*, t. 182, nr. 2, pp. 183-197, 2005.
- [29] W. Li, C. Peng, X. Zhou, J. Sun, Q. Zhu, H. Wu ir B. St-Onge, „Application of the ecosystem model and Markov Chain Monte Carlo for parameter estimation and productivity prediction,“ *Ecosphere. An ESA Open Access Journal*, t. 6, nr. 12, pp. 1-15, 2015.
- [30] A. Solonen, P. Ollinaho, M. Laine, H. Haario, J. Tamminen ir H. Jarvinen, „Efficient MCMC for Climate Model Parameter Estimation: Parallel Adaptive Chains and Early Rejection,“ *Bayesian Analysis*, t. 7, nr. 3, p. 715–736, 2012.
- [31] L. Görlitz, Z. Gao ir W. Schmitt, „Statistical Analysis of Chemical Transformation Kinetics Using Markov-Chain Monte Carlo Methods,“ *Environmental Science & Technology*, t. 45, nr. 10, p. 4429–4437, 2011.
- [32] S. Brooks, A. Gelman, G. L. Jones ir M. Xiao-Li, *MCMC using Hamiltonian dynamics*, Boca Raton: CRC Press, 2011.
- [33] P. Ivanchuk ir M. Ivanchuk, „One Example of Using Markov Chain Monte Carlo Method for Predicting in Medicine,“ *Cardiology and Cardiovascular Research*, t. 1, nr. 4, pp. 113-116, 2017.
- [34] O. O. Apenteng ir N. A. Ismail, „A Markov Chain Monte Carlo Approach to Estimate AIDS after HIV Infection,“ *PLoS One*, t. 10, nr. 7, 2015.
- [35] K. Vaikundamoorthy, „Diagnose of Primary Tumor Cancer using Markov Chain Monte-Carlo Convergence Model,“ *Advances in Applied Mathematical Biosciences*, t. 4, nr. 2, pp. 63-38, 2013.
- [36] K. Vaikundamoorthy, „Diagnosis of blood cancer using Markov chain Monte Carlo trace model,“ *International Journal of Biomathematics*, t. 10, nr. 3, 2017.
- [37] H. G. Claycamp, R. Kon, R. Fahmy ir S. W. Hoag, „Quality-by-Design II: Application of Quantitative Risk Analysis to the Formulation of Ciprofloxacin Tablets,“ *AAPS PharmSciTech*, t. 17, nr. 2, p. 233–244, 2015.

- [38] X. Feng ir D. Xie, „Application of MCMC Algorithm in Interest Rate Modeling,“ Hong Kong, 2011.
- [39] P. Damien ir P. P. N. G. S. D. A. Dellaportas, „Markov chain Monte Carlo methods in corporate finance,“ įtraukta *Bayesian Theory and Applications*, Oxford, Oxford University Press, 2012, pp. 516-545.
- [40] B. Eraker, M. Johannes ir N. Polson, „The Impact of Jumps in Volatility and Returns,“ *The Journal of Finance*, t. 58, nr. 3, pp. 1269-1300, 2003.
- [41] S. Sharma, „Markov Chain Monte Carlo Methods for Bayesian Data Analysis in Astronomy,“ *Annual Review of Astronomy and Astrophysics*, t. 55, p. 213–259, 2017.
- [42] P. Diaconis, „The Markov chain Monte Carlo revolution,“ *Bulletin of the American Mathematical Society*, t. 46, nr. 2, p. 179–205, 2009.
- [43] J. Chen ir J. S. Rosenthal, „Decrypting classical cipher text using Markov chain Monte Carlo,“ *Statistics and Computing*, t. 22, nr. 2, p. 397–413, 2012.
- [44] B. Silverman, *Density Estimation for Statistics and Data Analysis*, London: CRC Press, 1986.
- [45] S. Efromovich, „Orthogonal series density estimation,“ *John Wiley & Sons*, t. 2, pp. 467-476, 2010.
- [46] J. C. Mason ir D. Handscomb, *Chebyshev Polynomials*, New York: A CRC Press Company, 2003.
- [47] P. Wilmott, *Quantitative Finance*, John Wiley & Sons Ltd, 2006.

Pranešimai konferencijose

1. L. Lukašinskas, M. Landauskas – Application of MCMC method for modeling the value of stock prices („Mathematics and Natural Sciences: Theory and Application“, 2018-04-19);
2. L. Lukašinskas, M. Landauskas – Investicinio portfelio vertės prognozavimas Markovo grandinių Monte Karlo metodu („Matematika ir matematikos dėstymas – 2018“, 2018-05-25)

Priedai

1 priedas

Microsoft Corporation akcijų vertės

Data	Vertė	Data	Vertė	Data	Vertė	Data	Vertė
2017-01-03	62,58	2017-03-02	64,01	2017-04-28	68,46	2017-06-26	70,53
2017-01-04	62,30	2017-03-03	64,25	2017-05-01	69,41	2017-06-27	69,21
2017-01-05	62,30	2017-03-06	64,27	2017-05-02	69,30	2017-06-28	69,80
2017-01-06	62,84	2017-03-07	64,40	2017-05-03	69,08	2017-06-29	68,49
2017-01-09	62,64	2017-03-08	64,99	2017-05-04	68,81	2017-06-30	68,93
2017-01-10	62,62	2017-03-09	64,73	2017-05-05	69,00	2017-07-03	68,17
2017-01-11	63,19	2017-03-10	64,93	2017-05-08	68,94	2017-07-05	69,08
2017-01-12	62,61	2017-03-13	64,71	2017-05-09	69,04	2017-07-06	68,57
2017-01-13	62,70	2017-03-14	64,41	2017-05-10	69,31	2017-07-07	69,46
2017-01-17	62,53	2017-03-15	64,75	2017-05-11	68,46	2017-07-10	69,98
2017-01-18	62,50	2017-03-16	64,64	2017-05-12	68,38	2017-07-11	69,99
2017-01-19	62,30	2017-03-17	64,87	2017-05-15	68,43	2017-07-12	71,15
2017-01-20	62,74	2017-03-20	64,93	2017-05-16	69,41	2017-07-13	71,77
2017-01-23	62,96	2017-03-21	64,21	2017-05-17	67,48	2017-07-14	72,78
2017-01-24	63,52	2017-03-22	65,03	2017-05-18	67,71	2017-07-17	73,35
2017-01-25	63,68	2017-03-23	64,87	2017-05-19	67,69	2017-07-18	73,30
2017-01-26	64,27	2017-03-24	64,98	2017-05-22	68,45	2017-07-19	73,86
2017-01-27	65,78	2017-03-27	65,10	2017-05-23	68,68	2017-07-20	74,22
2017-01-30	65,13	2017-03-28	65,29	2017-05-24	68,77	2017-07-21	73,79
2017-01-31	64,65	2017-03-29	65,47	2017-05-25	69,62	2017-07-24	73,60
2017-02-01	63,58	2017-03-30	65,71	2017-05-26	69,96	2017-07-25	74,19
2017-02-02	63,17	2017-03-31	65,86	2017-05-30	70,41	2017-07-26	74,05
2017-02-03	63,68	2017-04-03	65,55	2017-05-31	69,84	2017-07-27	73,16
2017-02-06	63,64	2017-04-04	65,73	2017-06-01	70,10	2017-07-28	73,04
2017-02-07	63,43	2017-04-05	65,56	2017-06-02	71,76	2017-07-31	72,70
2017-02-08	63,34	2017-04-06	65,73	2017-06-05	72,28	2017-08-01	72,58
2017-02-09	64,06	2017-04-07	65,68	2017-06-06	72,52	2017-08-02	72,26
2017-02-10	64,00	2017-04-10	65,53	2017-06-07	72,39	2017-08-03	72,15
2017-02-13	64,72	2017-04-11	65,48	2017-06-08	71,95	2017-08-04	72,68
2017-02-14	64,57	2017-04-12	65,23	2017-06-09	70,32	2017-08-07	72,40
2017-02-15	64,53	2017-04-13	64,95	2017-06-12	69,78	2017-08-08	72,79
2017-02-16	64,52	2017-04-17	65,48	2017-06-13	70,65	2017-08-09	72,47
2017-02-17	64,62	2017-04-18	65,39	2017-06-14	70,27	2017-08-10	71,41
2017-02-21	64,49	2017-04-19	65,04	2017-06-15	69,90	2017-08-11	72,50
2017-02-22	64,36	2017-04-20	65,50	2017-06-16	70,00	2017-08-14	73,59
2017-02-23	64,62	2017-04-21	66,40	2017-06-19	70,87	2017-08-15	73,22
2017-02-24	64,62	2017-04-24	67,53	2017-06-20	69,91	2017-08-16	73,65
2017-02-27	64,23	2017-04-25	67,92	2017-06-21	70,27	2017-08-17	72,40
2017-02-28	63,98	2017-04-26	67,83	2017-06-22	70,26	2017-08-18	72,49
2017-03-01	64,94	2017-04-27	68,27	2017-06-23	71,21	2017-08-21	72,15

Data	Verté	Data	Verté	Data	Verté	Data	Verté
2017-08-22	73,16	2017-10-02	74,61	2017-11-09	84,09	2017-12-20	85,52
2017-08-23	72,72	2017-10-03	74,26	2017-11-10	83,87	2017-12-21	85,50
2017-08-24	72,69	2017-10-04	74,69	2017-11-13	83,93	2017-12-22	85,51
2017-08-25	72,82	2017-10-05	75,97	2017-11-14	84,05	2017-12-26	85,40
2017-08-28	72,83	2017-10-06	76,00	2017-11-15	82,98	2017-12-27	85,71
2017-08-29	73,05	2017-10-09	76,29	2017-11-16	83,20	2017-12-28	85,72
2017-08-30	74,01	2017-10-10	76,29	2017-11-17	82,40	2017-12-29	85,54
2017-08-31	74,77	2017-10-11	76,42	2017-11-20	82,53	2018-01-02	85,95
2017-09-01	73,94	2017-10-12	77,12	2017-11-21	83,72	2018-01-03	86,35
2017-09-05	73,61	2017-10-13	77,49	2017-11-22	83,11	2018-01-04	87,11
2017-09-06	73,40	2017-10-16	77,65	2017-11-24	83,26	2018-01-05	88,19
2017-09-07	74,34	2017-10-17	77,59	2017-11-27	83,87	2018-01-08	88,28
2017-09-08	73,98	2017-10-18	77,61	2017-11-28	84,88	2018-01-09	88,22
2017-09-11	74,76	2017-10-19	77,91	2017-11-29	83,34	2018-01-10	87,82
2017-09-12	74,68	2017-10-20	78,81	2017-11-30	84,17	2018-01-11	88,08
2017-09-13	75,21	2017-10-23	78,83	2017-12-01	84,26	2018-01-12	89,60
2017-09-14	74,77	2017-10-24	78,86	2017-12-04	81,08	2018-01-16	88,35
2017-09-15	75,31	2017-10-25	78,63	2017-12-05	81,59	2018-01-17	90,14
2017-09-18	75,16	2017-10-26	78,76	2017-12-06	82,78	2018-01-18	90,10
2017-09-19	75,44	2017-10-27	83,81	2017-12-07	82,49	2018-01-19	90,00
2017-09-20	74,94	2017-10-30	83,89	2017-12-08	84,16	2018-01-22	91,61
2017-09-21	74,21	2017-10-31	83,18	2017-12-11	85,23	2018-01-23	91,90
2017-09-22	74,41	2017-11-01	83,18	2017-12-12	85,58	2018-01-24	91,82
2017-09-25	73,26	2017-11-02	84,05	2017-12-13	85,35	2018-01-25	92,33
2017-09-26	73,26	2017-11-03	84,14	2017-12-14	84,69	2018-01-26	94,06
2017-09-27	73,85	2017-11-06	84,47	2017-12-15	86,85	2018-01-29	93,92
2017-09-28	73,87	2017-11-07	84,27	2017-12-18	86,38	2018-01-30	92,74
2017-09-29	74,49	2017-11-08	84,56	2017-12-19	85,83	2018-01-31	95,01

Barclays Bank PLC akciju vērtēs

Data	Vertē	Data	Vertē	Data	Vertē	Data	Vertē
2017-01-03	25,73	2017-03-03	25,94	2017-05-02	26,15	2017-06-29	26,61
2017-01-04	25,8	2017-03-06	25,93	2017-05-03	26,22	2017-06-30	26,59
2017-01-05	25,82	2017-03-07	25,94	2017-05-04	26,25	2017-07-03	26,68
2017-01-06	25,68	2017-03-08	25,89	2017-05-05	26,25	2017-07-05	26,71
2017-01-09	25,72	2017-03-09	25,75	2017-05-08	26,28	2017-07-06	26,73
2017-01-10	25,8	2017-03-10	25,78	2017-05-09	26,3	2017-07-07	26,77
2017-01-11	25,93	2017-03-13	25,84	2017-05-10	26,41	2017-07-10	26,73
2017-01-12	25,93	2017-03-14	25,75	2017-05-11	26,37	2017-07-11	26,77
2017-01-13	25,92	2017-03-15	25,9	2017-05-12	26,41	2017-07-12	26,77
2017-01-17	25,83	2017-03-16	25,93	2017-05-15	26,46	2017-07-13	26,77
2017-01-18	25,78	2017-03-17	25,99	2017-05-16	26,6	2017-07-14	26,78
2017-01-19	25,67	2017-03-20	25,99	2017-05-17	26,57	2017-07-17	26,7
2017-01-20	25,67	2017-03-21	25,81	2017-05-18	26,58	2017-07-18	26,77
2017-01-23	25,86	2017-03-22	25,83	2017-05-19	26,58	2017-07-19	26,79
2017-01-24	25,83	2017-03-23	25,97	2017-05-22	26,73	2017-07-20	26,83
2017-01-25	25,82	2017-03-24	25,92	2017-05-23	26,7	2017-07-21	27,08
2017-01-26	25,82	2017-03-27	25,88	2017-05-24	26,74	2017-07-24	26,97
2017-01-27	25,83	2017-03-28	25,98	2017-05-25	26,67	2017-07-25	26,95
2017-01-30	25,81	2017-03-29	25,91	2017-05-26	26,64	2017-07-26	26,92
2017-01-31	25,8	2017-03-30	25,94	2017-05-30	26,2	2017-07-27	26,91
2017-02-01	25,85	2017-03-31	25,86	2017-05-31	26,24	2017-07-28	26,93
2017-02-02	26,04	2017-04-03	26,08	2017-06-01	26,33	2017-07-31	26,97
2017-02-03	26,15	2017-04-04	26,13	2017-06-02	26,32	2017-08-01	26,97
2017-02-06	26,16	2017-04-05	26,13	2017-06-05	26,31	2017-08-02	26,96
2017-02-07	26,22	2017-04-06	26,23	2017-06-06	26,28	2017-08-03	26,95
2017-02-08	26,31	2017-04-07	26,22	2017-06-07	26,29	2017-08-04	26,91
2017-02-09	26,25	2017-04-10	26,25	2017-06-08	26,33	2017-08-07	26,91
2017-02-10	26,19	2017-04-11	26,24	2017-06-09	26,29	2017-08-08	26,95
2017-02-13	26,2	2017-04-12	26,23	2017-06-12	26,34	2017-08-09	26,93
2017-02-14	26,2	2017-04-13	26,24	2017-06-13	26,41	2017-08-10	26,88
2017-02-15	26,23	2017-04-17	26,26	2017-06-14	26,51	2017-08-11	26,89
2017-02-16	26,25	2017-04-18	26,13	2017-06-15	26,59	2017-08-14	26,92
2017-02-17	26,21	2017-04-19	26,16	2017-06-16	26,58	2017-08-15	26,92
2017-02-21	26,25	2017-04-20	26,16	2017-06-19	26,46	2017-08-16	27
2017-02-22	26,31	2017-04-21	26,17	2017-06-20	26,48	2017-08-17	26,91
2017-02-23	26,33	2017-04-24	26,15	2017-06-21	26,43	2017-08-18	26,92
2017-02-24	26,33	2017-04-25	26,13	2017-06-22	26,46	2017-08-21	27,14
2017-02-27	25,9	2017-04-26	26,13	2017-06-23	26,43	2017-08-22	26,98
2017-02-28	25,96	2017-04-27	26,15	2017-06-26	26,48	2017-08-23	26,96
2017-03-01	25,93	2017-04-28	26,13	2017-06-27	26,5	2017-08-24	26,98
2017-03-02	25,93	2017-05-01	26,15	2017-06-28	26,52	2017-08-25	27,05

Data	Vertè	Data	Vertè	Data	Vertè	Data	Vertè
2017-08-28	27,07	2017-10-05	26,28	2017-11-13	26,74	2017-12-21	26,8
2017-08-29	27,1	2017-10-06	26,27	2017-11-14	26,7	2017-12-22	26,68
2017-08-30	26,76	2017-10-09	26,26	2017-11-15	26,75	2017-12-26	26,65
2017-08-31	26,81	2017-10-10	26,41	2017-11-16	26,83	2017-12-27	26,72
2017-09-01	26,76	2017-10-11	26,36	2017-11-17	26,97	2017-12-28	26,65
2017-09-05	26,72	2017-10-12	26,34	2017-11-20	26,93	2017-12-29	26,67
2017-09-06	26,82	2017-10-13	26,4	2017-11-21	26,95	2018-01-02	26,45
2017-09-07	26,81	2017-10-16	26,38	2017-11-22	26,96	2018-01-03	26,4
2017-09-08	26,73	2017-10-17	26,4	2017-11-24	27,02	2018-01-04	26,35
2017-09-11	26,73	2017-10-18	26,5	2017-11-27	26,93	2018-01-05	26,27
2017-09-12	26,72	2017-10-19	26,49	2017-11-28	26,99	2018-01-08	26,39
2017-09-13	26,74	2017-10-20	26,57	2017-11-29	26,98	2018-01-09	26,25
2017-09-14	26,71	2017-10-23	26,54	2017-11-30	26,5	2018-01-10	26,33
2017-09-15	26,8	2017-10-24	26,5	2017-12-01	26,64	2018-01-11	26,3
2017-09-18	26,83	2017-10-25	26,46	2017-12-04	26,62	2018-01-12	26,24
2017-09-19	26,85	2017-10-26	26,42	2017-12-05	26,47	2018-01-16	26,25
2017-09-20	26,81	2017-10-27	26,54	2017-12-06	26,45	2018-01-17	26,22
2017-09-21	26,85	2017-10-30	26,65	2017-12-07	26,56	2018-01-18	26,06
2017-09-22	26,79	2017-10-31	26,68	2017-12-08	26,5	2018-01-19	26,51
2017-09-25	26,76	2017-11-01	26,53	2017-12-11	26,46	2018-01-22	26,45
2017-09-26	26,72	2017-11-02	26,6	2017-12-12	26,52	2018-01-23	26,37
2017-09-27	26,66	2017-11-03	26,65	2017-12-13	26,64	2018-01-24	26,41
2017-09-28	26,58	2017-11-06	26,57	2017-12-14	26,58	2018-01-25	26,45
2017-09-29	26,55	2017-11-07	26,6	2017-12-15	26,66	2018-01-26	26,35
2017-10-02	26,54	2017-11-08	26,66	2017-12-18	26,89	2018-01-29	26,23
2017-10-03	26,54	2017-11-09	26,68	2017-12-19	26,9	2018-01-30	26,12
2017-10-04	26,51	2017-11-10	26,65	2017-12-20	26,78	2018-01-31	26,23

Prognozavimo tikslumo matai

MCMC metodo paklaidos prognozuojant akcijų vertes

Prognozių skaičius	Microsoft Corporation			Barclays Bank PLC		
	MPE	MAPE	MSE	MPE	MAPE	MSE
50	1,55386	1,57741	2,62804	-0,57174	0,57174	0,02516
100	1,30259	1,31753	1,88866	-0,76973	0,76973	0,04342
500	1,32035	1,33267	1,89151	-0,81819	0,81819	0,04861
1000	1,08948	1,10816	1,37656	-0,82903	0,82903	0,04989
2000	1,10681	1,12298	1,41445	-0,8219	0,8219	0,04911
3000	1,08652	1,10563	1,38353	-0,82687	0,82687	0,04968
4000	1,08345	1,10678	1,38447	-0,82759	0,82759	0,04977
5000	1,09168	1,11266	1,39997	-0,83386	0,83386	0,05047
6000	1,10702	1,12772	1,42982	-0,83432	0,83432	0,05051
7000	1,10161	1,12178	1,42155	-0,84043	0,84043	0,05119
8000	1,0978	1,11733	1,41158	-0,83598	0,83598	0,05067
9000	1,09886	1,11717	1,41035	-0,83551	0,83551	0,05061
10000	1,11213	1,13069	1,43779	-0,83809	0,83809	0,05092
11000	1,1163	1,13435	1,44603	-0,83969	0,83969	0,0511
12000	1,10303	1,12332	1,42202	-0,83973	0,83973	0,0511
13000	1,11535	1,13379	1,44426	-0,83932	0,83932	0,05106
14000	1,12437	1,14191	1,46114	-0,83948	0,83948	0,05108
15000	1,12637	1,14427	1,46537	-0,8398	0,8398	0,05113

Monte Karlo modelio paklaidos prognozuojant akcijų vertes

Prognozių skaičius	Microsoft Corporation			Barclays Bank PLC		
	MPE	MAPE	MSE	MPE	MAPE	MSE
50	2,47822	2,47822	6,02972	-1,22111	1,22111	0,10697
100	2,4341	2,4341	5,52024	-1,20416	1,20416	0,10415
500	2,3974	2,3974	5,34701	-1,32434	1,32434	0,12537
1000	2,11649	2,11649	4,28474	-1,32401	1,32401	0,12524
2000	2,06474	2,06474	4,08457	-1,32722	1,32722	0,12712
3000	2,05471	2,11649	4,02465	-1,31177	1,32401	0,12349
4000	2,01327	2,01327	3,87404	-1,33727	1,33727	0,12915
5000	2,00067	2,00067	3,84122	-1,31572	1,31572	0,12448
6000	1,94598	1,94598	3,66261	-1,32824	1,32824	0,12731
7000	1,99403	2,00067	3,83224	-1,32193	1,31572	0,12601
8000	1,98767	1,98767	3,80439	-1,31694	1,31694	0,12474
9000	2,02442	2,02442	3,92604	-1,31372	1,31372	0,12436
10000	2,02176	2,02176	3,92051	-1,30318	1,30318	0,12214
11000	2,06474	2,06474	4,08457	-1,32722	1,32722	0,12712
12000	2,03876	2,03876	3,97336	-1,31623	1,31623	0,12487
13000	2,0431	2,0431	3,98518	-1,29974	1,29974	0,12154
14000	2,03418	2,03418	3,95687	-1,31523	1,31523	0,12465
15000	2,03935	2,03935	3,97959	-1,3052	1,3052	0,12256

MCMC ir Monte Karlo Investicinio portfelio prognozių palyginimas

Prognozių skaičius	MPE		MAPE		MSE	
	MCMC	Monte Karlo	MCMC	Monte Karlo	MCMC	Monte Karlo
50	1,09373	1,62824	1,12181	1,65849	54,98680	116,07804
100	0,70471	1,59774	0,74714	1,59774	27,33338	104,64766
500	0,70353	1,54185	0,73355	1,54185	26,69175	97,77387
1000	0,64914	1,32570	0,69387	1,32570	24,13285	75,61464
2000	0,60976	1,28520	0,65935	1,28520	22,50606	71,00185
3000	0,61090	1,28094	0,67503	1,32570	23,03758	70,21388
4000	0,63944	1,24322	0,69806	1,24322	24,29053	66,34277
5000	0,67715	1,23846	0,71912	1,23846	25,71447	66,29953
6000	0,66957	1,19353	0,72590	1,19660	25,89683	62,30263
7000	0,65667	1,23197	0,71428	1,23846	25,22387	65,89836
8000	0,66293	1,22819	0,70979	1,22936	25,17667	65,52244
9000	0,64841	1,25760	0,70790	1,25760	24,85850	68,18746
10000	0,66365	1,25760	0,71030	1,25760	25,18269	68,18746
11000	0,65846	1,28520	0,71451	1,28520	25,28724	71,00185
12000	0,66113	1,26769	0,71551	1,26769	25,37024	68,88286
13000	0,67054	1,27480	0,71661	1,27480	25,63534	69,55199
14000	0,65783	1,26439	0,71340	1,26439	25,26283	68,57198
15000	0,65954	1,27067	0,71508	1,27067	25,35741	69,34353

MCMC ir Monte Karlo Investicinio portfelio gautosios vertės esant 15000 trajektorijų

Data	Tikroji vertė	MCMC	Monte Karlo metodas
2017-12-29	561,050	561,05	561,05
2018-01-02	562,00	563,22745	561,8772
2018-01-03	563,75	563,77745	562,0536
2018-01-04	567,30	564,41765	562,2747
2018-01-05	572,30	564,9605	562,5312
2018-01-08	573,35	566,6788	563,2564
2018-01-09	572,35	567,2522	563,5474
2018-01-10	570,75	567,8341	563,8794
2018-01-11	571,90	568,4692	564,0806
2018-01-12	579,20	569,0474	564,2746
2018-01-16	573,00	571,50695	565,2946

R programinis kodas

```

#Microoft cooperation
#graza vn = (Sn-Sn-1)/Sn-1
#data <- read.csv(` failas1.txt, sep=` `, dec=`.`)
#Barclays Bank
#graza vn = (Sn-Sn-1)/Sn-1
data <- read.csv(` failas2.txt, sep=` `, dec=`.`)
hist(data, probability = FALSE, breaks = 40, xlab="Graža", ylab = "Dažnis", main="Duomenų histograma")
pos <- (data-min(data))/(max(data)-min(data))
data2 <- pos*2 - 1
hist(data2, probability = TRUE, breaks = 50, xlab="Graža", ylab = "Dažnis", main="Transformuotų duomenų
histograma")
TTT <- function(n,x) { cos(n*acos(x))}
x <- seq(-1, 1, 0.00001)
m <- 100
a <- numeric(length=(m+1))
for (i in 0:m){
  a[i+1] <- sum( TTT(i, data2))/length(data2)}
f.estimate <- function(m, X){
  sum <- 0
  for(i in 0:m){
    sum <- sum + a[i+1]*TTT(i, X) }
  return (sum)}
f.estimate.final <- function(m, x, data, miny, plotas){
  x <- min(max(x, min(data)), max(data))
  x <- 2*(x-min(data))/(max(data)-min(data))-1
  return((f.estimate(m, x)-miny)/plotas)}
library(sfsmisc)
x <- seq(-1, 1, 0.00001)
y <- f.estimate(m, x)
miny <- min(y)
plotas <- integrate.xy(x, y-miny, -1, 1)
y <- (y- miny)/plotas
lines(x, y)
xx <- runif(length(x))*2-1
yy <- runif(length(x))*4
points(xx,yy)
id <- yy < y
xx <- sort(xx)
vec <- xx[id]
hist(vec, probability = TRUE, breaks = 70, xlab="Graža")
lines(x, y)
transformacija <- (vec+1)/2*(max(data)-min(data))+min(data)
hist(transformacija, probability = TRUE, breaks = 70, xlab="Graža")
library(moments)
empirical1 <- moment(data, 1, central = FALSE)
theoretical1 <- moment(transformacija, 1, central = FALSE)
empirical2 <- moment(data, 2, central = FALSE)
theoretical2 <- moment(transformacija, 2, central = FALSE)
empirical3 <- moment(data, 3, central = FALSE)
theoretical3 <- moment(transformacija, 3, central = FALSE)
cempirical1 <- moment(data, 1, central = TRUE)
ctheoretical1 <- moment(transformacija, 1, central = TRUE)
cempirical2 <- moment(data, 2, central = TRUE)
ctheoretical2 <- moment(transformacija, 2, central = TRUE)
cempirical3 <- moment(data, 3, central = FALSE)
ctheoretical3 <- moment(transformacija, 3, central = TRUE)
Tikslas <- abs(empirical1-theoretical1)+ 2*abs(empirical2-theoretical2) + 3*abs(empirical3-theoretical3)+
2*abs(cempirical1-ctheoretical1)+1*abs(cempirical2-ctheoretical2)+2* abs(cempirical3-ctheoretical3)

```

```

matrica<-numeric(length=(m))
matrica1<-numeric(length=(m))
matrica2<-numeric(length=(m))
matrica3<-numeric(length=(m))
for(i in 0:m){
  x <-seq(-1, 1, 0.00001)
  y <-f.estimate(i, x)
  plotas <- integrate.xy(x, y-min(y) , -1, 1 )
  y <- (y- min(y))/plotas
  id <- yy < y
  xx <- sort(xx)
  vec <- xx[id]
  transformacija <-(vec+1)/2*(max(data)-min(data))+min(data)
  theoretical1 <- moment(transformacija, 1, central = FALSE)
  theoretical2 <- moment(transformacija, 2, central = FALSE)
  theoretical3 <- moment(transformacija, 3, central = FALSE)
  ctheoretical1 <- moment(transformacija, 1, central = TRUE)
  ctheoretical2 <- moment(transformacija, 2, central = TRUE)
  ctheoretical3 <- moment(transformacija, 3, central = TRUE)
  Tikslas <- 0.01*abs(empirical1-theoretical1)+ 6*abs(empirical2-theoretical2) + 10*abs(empirical3-
theoretical3)+1*abs(cempirical1-ctheoretical1)+1*abs(cempirical2-ctheoretical2)+1* abs(cempirical3-ctheoretical3)
  Tikslas1 <- 0.01*abs(empirical1-theoretical1)+ 6*abs(empirical2-theoretical2) +10*abs(empirical3-theoretical3)+
1*abs(cempirical1-ctheoretical1)+1*abs(cempirical2-ctheoretical2)+1* abs(cempirical3-ctheoretical3)
  Tikslas2 <-0.05*abs(empirical1-theoretical1)+ 0.04*abs(empirical2-theoretical2) +12*abs(empirical3-theoretical3)+
1*abs(cempirical1-ctheoretical1)+1*abs(cempirical2-ctheoretical2)+1* abs(cempirical3-ctheoretical3)
  Tikslas3 <- 4*abs(empirical1-theoretical1)+ 6*abs(empirical2-theoretical2) + 13*abs(empirical3-theoretical3)+
1*abs(cempirical1-ctheoretical1)+1*abs(cempirical2-ctheoretical2)+1* abs(cempirical3-ctheoretical3)
  matrica[i] <- Tikslas
  matrica1[i] <- Tikslas1
  matrica2[i] <- Tikslas2
  matrica3[i] <- Tikslas3
  plot(matrica, xlab="K", ylab="T(K)", pch=".", cex = 4 ) #kaip piesti linija??}
plot(matrica, xlab="K", ylab="T(K)", pch=".", cex = 4, xaxt='n', yaxt='n', ylim=c(0,0.004))
points(matrica1, type = "p", pch=".", cex = 4, col = "red")
points(matrica, type = "p", pch=".", cex = 4, col = "blue")
title(main = "Tikslo funkcijos priklausomybė nuo aproksimacijos eilės")
axis(side=1, at=seq(from = 0, to = 150, by = 30))
axis(2,at=seq(from = 0, to = 0.004, by = 0.001))
which.min(matrica)
which.min(matrica1)
which.min(matrica2)
min(matrica)
m <-25
a<-numeric(length=(m+1))
for (i in 0:m){
  a[i+1]<-sum( TTT(i, data2))/length(data2)}
hist(data2, probability = TRUE, breaks = 70, xlab="Graža", ylab ="Tankis", main="Histograma ir jos Čebyševio
polinomų aproksimacija")
x <-seq(-1, 1, 0.00001)
y <-f.estimate(m, x)
miny<- min(y)
plotas <-integrate.xy(x, y-miny , -1, 1 )
y <- (y- miny)/plotas
lines(x, y)
histograma <-hist(transformacija, probability = TRUE, breaks =40)
suminis <- cumsum(histograma$counts/sum(histograma$counts))
suminis.x <- histograma$breaks
suminis.y <- c(0, suminis)
n=1000000
x <- 0.1
vec1 <- vector("numeric", n) #grazos v
vec1[1] <- x
aprob <- vector("numeric", n) #grazos v

```

```

piX <- vector("numeric", n)
piY <- vector("numeric", n)
qXY <- vector("numeric", n)
qYX <- vector("numeric", n)
my.approx <- function(x, y, x0){
  x0 <- min(max(x0, x[1]), tail(x, n=1))
  return(approx(x, y, xout=x0)$y)}
argerai <- vector("numeric", n) #grazos v
for (i in 2:n) {
  u <- runif(1)
  n <- length(suminis.x)
  id1 <- max(which (suminis.y[-n] <= u))
  id2 <- min(which (suminis.y [-1] >= u))+1
  inv <- (suminis.x[id2]-suminis.x[id1]) * (u-suminis.y[id1])/(suminis.y[id2]-suminis.y[id1])+ suminis.x[id1]
  y <- inv #y~q(.|Xt)
  piX[i] <- f.estimate.final(m, x, data, miny,plotas)
  piY[i] <- f.estimate.final(m, y, data, miny,plotas)
  qXY[i] <- my.approx(histograma$breaks[-1],histograma$density,x)
  qYX[i] <- my.approx(histograma$breaks[-1],histograma$density,y)
  aprob[i] <- min(1, (piY[i]*qXY[i])/(piX[i]*qYX[i]))
  v <- runif(1)
  if (v < aprob[i]) {x <- y}
  vec1[i] <- x}
rr <- as.mcmc(vec1)
rejectionRate(rr)
plot(vec1)
hist(vec1, probability = TRUE, xlim = c(min(data), max(data)), breaks = 100)
#antras
n=1000000
x <- -0.01
vec2 <- vector("numeric", n) #grazos v
vec2[1] <- x
for (i in 2:n) {
  u <- runif(1)
  n <- length(suminis.x)
  id1 <- max(which (suminis.y[-n] <= u))
  id2 <- min(which (suminis.y [-1] >= u))+1
  inv <- (suminis.x[id2]-suminis.x[id1]) * (u-suminis.y[id1])/(suminis.y[id2]-suminis.y[id1])+ suminis.x[id1]
  y <- inv #y~q(.|Xt)
  piX[i] <- f.estimate.final(m, x, data, miny,plotas)
  piY[i] <- f.estimate.final(m, y, data, miny,plotas)
  qXY[i] <- my.approx(histograma$breaks[-1],histograma$density,x)
  qYX[i] <- my.approx(histograma$breaks[-1],histograma$density,y)
  aprob[i] <- min(1, (piY[i]*qXY[i])/(piX[i]*qYX[i]))
  v <- runif(1)
  if (v < aprob[i]) { x <- y}
  vec2[i] <- x}
hist(vec2, probability = TRUE, xlim = c(min(data), max(data)), breaks = 200)
#trecias
n=1000000
x <- 0
vec3 <- vector("numeric", n)
vec3[1] <- x
histograma <- hist(transformacija, probability = TRUE, breaks = 40)
suminis <- cumsum(histograma$counts/sum(histograma$counts))
suminis.x <- histograma$breaks
suminis.y <- c(0, suminis)
for (i in 2:n) {
  u <- runif(1)
  n <- length(suminis.x)
  id1 <- max(which (suminis.y[-n] <= u))
  id2 <- min(which (suminis.y [-1] >= u))+1
  inv <- (suminis.x[id2]-suminis.x[id1]) * (u-suminis.y[id1])/(suminis.y[id2]-suminis.y[id1])+ suminis.x[id1]

```

```

y <- inv #y~q(.|Xt)
piX[i] <- f.estimate.final(m, x, data, miny,plotas)
piY[i] <- f.estimate.final(m, y, data, miny,plotas)
qXY[i]<- my.approx(histograma$breaks[-1],histograma$density,x)
qYX[i]<- my.approx(histograma$breaks[-1],histograma$density,y)
aprob[i] <- min(1, (piY[i]*qXY[i])/(piX[i]*qYX[i]))
v<- runif(1)
if (v < aprob[i]) {x <- y}
vec3[i] <- x}
hist(vec3, probability = TRUE, xlim = c(min(data), max(data)), breaks = 500)
#trecias
library(coda)
arv11 <- as.mcmc(BarcleyVec1)
arv22<- as.mcmc(BarcleyVec2)
arv33<- as.mcmc(BarcleyVec3)
geweke.diag(tail(arv11, 1000000-20000), frac1=0.1, frac2=0.5)
geweke.plot(tail(arv11, 1000000-20000))
heidel.diag(tail(arv11,1000000-20000), eps=0.1, pvalue=0.05)
mh.list1 <- mcmc.list(head(arv11, 10000), head(arv22, 10000), head(arv33, 10000))
gelman2 <- gelman.plot(mh.list1)
rmeanplot(arv11, style = "gray", ylim=c(-0.00052, -0.00046))
rmeanplot(arv22, style = "gray", ylim=c(-0.0004, -0.0005))
rmeanplot(arv33, style = "gray", ylim=c(-0.0005, -0.0004))
arv1 <- as.mcmc(MicrosoftVec1)
arv2 <- as.mcmc(MicrosoftVec2)
arv3 <- as.mcmc(MicrosoftVec3)
mh.list <- mcmc.list(head(arv1, 6000), head(arv2, 6000), head(arv3, 6000))
abcd <-gelman.diag(mh.list)
abcd <-gelman.plot(mh.list)
geweke.diag(arv1, frac1=0.1, frac2=0.5)
geweke.plot(head(arv1, 14950))
raftery.diag(arv1, q = 0.025, r = 0.005, s = 0.95)
heidel.diag(arv1, eps=0.1, pvalue=0.05)
library(mcmcplots)
rmeanplot(arv1, style = "gray", ylim=c(0.001, 0.002))
rmeanplot(arv2, style = "plain", ylim=c(0, 0.002))
rmeanplot(arv3, style = "plain", ylim=c(0, 0.003))
rmeanplot(c(arv1, arv2, arv3), style = "plain")
traceplot(arv3)
kiek<- 100
kaskiek <- 100
aaa <- vec1[seq(1, kiek, 1)]
momentas1 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas2 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas3 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas1 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas2 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas3 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
Tikslofvec1 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
for(i in 0:(length(vec2)-kiek)/kaskiek){
  aaa <- arv11[seq(kaskiek*i+1, kiek+kaskiek*i, 1)]
  momentas1[i+1] <- abs(moment(aaa, 1, central = FALSE) - empirical1)
  momentas2[i+1] <- abs(moment(aaa, 2, central = FALSE) - empirical2)
  momentas3[i+1] <- abs(moment(aaa, 3, central = FALSE) - empirical3)
  cmomentas1[i+1] <- abs(moment(aaa, 1, central = TRUE) - cempirical1)
  cmomentas2[i+1] <- abs(moment(aaa, 2, central = TRUE) - cempirical2)
  cmomentas3[i+1] <- abs(moment(aaa, 3, central = TRUE) - cempirical3) }
bbb <- vec2[seq(1, kiek, 1)]
momentas11 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas22 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas33 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas11 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas22<- vector("numeric", (length(vec1)-kiek)/kaskiek)

```

```

cmomentas33 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
Tikslofvec2 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
for(i in 0:(length(vec2)-kiek)/kaskiek){
  bbb <- arv2[seq(kaskiek*i+1, kiek+kaskiek*i, 1)]
  momentas11[i+1] <- abs(moment(bbb, 1, central = FALSE) - empirical1)
  momentas22[i+1] <- abs(moment(bbb, 2, central = FALSE) - empirical2)
  momentas33[i+1] <- abs(moment(bbb, 3, central = FALSE) - empirical3)
  cmomentas11[i+1] <- abs(moment(bbb, 1, central = TRUE) - cempirical1)
  cmomentas22[i+1] <- abs(moment(bbb, 2, central = TRUE) - cempirical2)
  cmomentas33[i+1] <- abs(moment(bbb, 3, central = TRUE) - cempirical3) }
ccc <- vec3[seq(1, kiek, 1)]
momentas111 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas222 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
momentas333 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas111 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas222 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
cmomentas333 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
Tikslofvec3 <- vector("numeric", (length(vec1)-kiek)/kaskiek)
for(i in 0:(length(vec2)-kiek)/kaskiek){
  ccc <- arv3[seq(kaskiek*i+1, kiek+kaskiek*i, 1)]
  momentas111[i+1] <- abs(moment(ccc, 1, central = FALSE) - empirical1)
  momentas222[i+1] <- abs(moment(ccc, 2, central = FALSE) - empirical2)
  momentas333[i+1] <- abs(moment(ccc, 3, central = FALSE) - empirical3)
  cmomentas111[i+1] <- abs(moment(ccc, 1, central = TRUE) - cempirical1)
  cmomentas222[i+1] <- abs(moment(ccc, 2, central = TRUE) - cempirical2)
  cmomentas333[i+1] <- abs(moment(ccc, 3, central = TRUE) - cempirical3) }
for(i in 0:(length(vec1)-kiek)/kaskiek){
  Tikslofvec1[i+1] <- 1*momentas1[i+1]+ 1*momentas2[i+1] + 1*momentas3[i+1]+ 1*cmomentas1[i+1]
+1*cmomentas2[i+1]+1*cmomentas3[i+1]
  Tikslofvec2[i+1] <- 1*momentas11[i+1]+ 1*momentas22[i+1] + 1*momentas33[i+1]+ 1*cmomentas11[i+1]
+1*cmomentas22[i+1]+1*cmomentas33[i+1]
  Tikslofvec3[i+1] <- 1*momentas111[i+1]+ 1*momentas222[i+1] + 1*momentas333[i+1]+ 1*cmomentas111[i+1]
+1*cmomentas222[i+1]+1*cmomentas333[i+1]}
plot(Tikslofvec1, pch=".", cex = 2)
points(Tikslofvec2, type = "p", pch=".", cex = 2, col = "red")
points(Tikslofvec3, type = "p", pch=".", cex = 2, col = "blue")
rmeanplot(Tikslofvec1)
rmeanplot(Tikslofvec2)
rmeanplot(Tikslofvec3)
library(pracma)
xyz1 <- movavg(Tikslofvec1, 100, type="s")
xyz2 <- movavg(Tikslofvec2, 100, type="s")
xyz <- movavg(Tikslofvec3, 100, type="s")
plot(xyz, pch=".", cex = 2)
plot(head(xyz, 500), pch=".", cex = 2)

reiksmes <- tail(vec2, length(vec2)-20000)
ilgis <- length(reiksmes)
isrinkta <- reiksmes[seq(1, ilgis2, 3)]
grazos <- sample(isrinkta)

#Microsoft corporation istorija 2017.01.01-2018.01.02
istorija1 <- read.csv(`istorija1.txt, sep=` ` , dec=` `.`)
#Microsoft corporation istorija 2017.12.29-2018.01.16
istorija11 <- read.csv(`istorija11.txt, sep=` ` , dec=` `.`)
#Microsoft corporation istorijos 2017.01.03-2017.12.31 grazos (Sn/Sn-1)
grazos1 <- read.csv(`grazos1.txt, sep=` ` , dec=` `.`)
#Microoft corporation normalumo testai
loggrazos1 <- log(grazos1)
shapiro.test(loggrazos1)
sf.test(loggrazos1)
cvm.test(loggrazos1)
lillie.test(loggrazos1)

```

```

ad.test(loggrazos1)
pearson.test(loggrazos1)
shapiro.test(tail(loggrazos1, 34))
sf.test(tail(loggrazos1, 34))
cvm.test(tail(loggrazos1, 34))
lillie.test(tail(loggrazos1, 34))
ad.test(tail(loggrazos1, 34))
pearson.test(tail(loggrazos1, 34))
ZZ1 <- rnorm(15000, mean = 0, sd = 1)
Z1 <- rnorm(15000*prognoze, mean = 0, sd = 1)
ZZ1 <- as.vector(ZZ1)
Z1 <- as.vector(Z1)
normloggrazos1 <- tail(loggrazos1, 34)
prognoze <- 18
vidloggrazos1 <- mean(normloggrazos1)
sigma1 <- sd(tail(log(istorija1), 34), na.rm = TRUE)
#Monte karlo modelis
S <- 85.54
#Kokiam laikotarpiui prognozuojama
prognoze <- 18
#Prognozuojamu trajektoriju skaicius
kiekprognoziu <- 15000
#Microoft cooperation
mcakcijuvertes1 <- matrix(nrow = prognoze, ncol = kiekprognoziu)
for (j in 1:kiekprognoziu) {
  mcakcijuvertes1[1, j] <- S * exp((vidloggrazos1 -0.5*sigma1*sigma1)+sigma1*ZZ1[j,])
  for (i in 2:prognoze) {
    mcakcijuvertes1[i, j] <- mcakcijuvertes1[i-1, j] * exp((vidloggrazos1 -0.5*sigma1*sigma1)+sigma1*Z1[(j-1)*prognoze+i,])} }
mcpradineverte1 <- matrix(S, nrow = 1, ncol = kiekprognoziu )
mcforecast1 <- rbind(mcpradineverte1, mcakcijuvertes1)
mcvidurkis1 <- rowMeans(mcforecast1, na.rm = FALSE, dims = 1)
mcPCP21 <- ts(mcforecast1, frequency = 365, start = 2018-3/365)
PCP41 <- ts(na.approx(istorija1), frequency = 365, start = 2018-3/365)
PCP1 <- ts(na.approx(istorija1), frequency = 365, start = 2017)
ts.plot(mcPCP21, xlim= c(2017, 2018.033), ylim= c(60, 110), col="blue", ylab="Kaina, USD", xlab ="Laikas")
title(main = "Microsoft Corporation")
lines(PCP1, type = "l", col="black")
lines(PCP41, type = "l", col="black", lwd = 3)
CL1 <- apply(mcforecast1, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcCL1 <- apply(mcforecast1, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcPCP31 <- ts(t(CL1), frequency = 365, start = 2018-3/365)
lines(mcPCP31[, 1], col=c("red"), lwd = 3)
lines(mcPCP31[, 2], col=c("red"), lwd = 3)
mcvidurkiolinija1 <- ts(mcvidurkis1, frequency = 365, start = 2018-3/365)
lines(mcvidurkiolinija1, type = "l", col="orange", lwd = 3)
legend("topleft", legend = c("Istorija", "Pasikliautinasis intervalas 95%", "Prognozės", "Vidutinė prognozė"), col =c("black", "red", "blue", "orange"), lty = 1)
ZZ2 <- rnorm(kiekprognoziu, mean = 0, sd = 1)
Z2 <- rnorm(kiekprognoziu*prognoze, mean = 0, sd = 1)
#Barclays Bank istorija 2017.01.03-2018.01.02
istorija2 <- read.csv("istorija2.txt", sep="`", dec=".`")
#Barclays Bank istorija 2017.12.29-2018.01.16
istorija22 <- read.csv("istorija22.txt", sep="`", dec=".`")
#Barclays Bank istorijos 2017.01.03-2017.12.31 grazos (Sn/Sn-1)
grazos2 <- read.csv("grazos2.txt", sep="`", dec=".`")
#Barclays Bank normalumo testai
loggrazos2 <- log(grazos2)
shapiro.test(loggrazos2)
sf.test(loggrazos2)
cvm.test(loggrazos2)
lillie.test(loggrazos2)
ad.test(loggrazos2)

```

```

pearson.test(loggrazos2)
shapiro.test(tail(loggrazos2, 34))
sf.test(tail(loggrazos2, 34))
cvm.test(tail(loggrazos2, 34))
lillie.test(tail(loggrazos2, 34))
ad.test(tail(loggrazos2,34))
pearson.test(tail(loggrazos2, 34))
#Barclays Bank
normloggrazos2 <- tail(loggrazos2, 34)
vidloggrazos2 <- mean(normloggrazos2)
prognoze <- 18
ZZ2 <- rnorm(15000, mean = 0, sd = 1)
Z2 <- rnorm(15000*prognoze, mean = 0, sd = 1)
ZZ2 <- as.vector(ZZ2)
Z2 <- as.vector(Z2)
sigma2 <- sd(tail(log(istorija2), 34), na.rm = TRUE)
#Monte karlo modelis
S <- 26.67
#Kokiam laikotarpiui prognozuojama
prognoze <-18
#Prognozuojamu trajektoriju skaičius
kiekprognoziu <- 15000
mcakcijuvertes2 <- matrix(nrow = prognoze, ncol = kiekprognoziu)
for (j in 1:kiekprognoziu) {
  #Z <- rnorm(1, mean = 0, sd = 1)
  mcakcijuvertes2[1, j] <- S * exp((vidloggrazos2 -0.5*sigma2*sigma2)+sigma2*ZZ2[j, ])
  for (i in 2:prognoze) {
    #Z <- rnorm(1, mean = 0, sd = 1)
    mcakcijuvertes2[i, j] <- mcakcijuvertes2[i-1, j] * exp((vidloggrazos2 -0.5*sigma2*sigma2)+sigma2*Z2[(j-1)*prognoze+i, ]) }
}
mcpradineverte2 <- matrix(S, nrow = 1, ncol = kiekprognoziu )
mcforecast2 <- rbind(mcpradineverte2, mcakcijuvertes2)
mcvidurkis2 <- rowMeans(mcfcast2, na.rm = FALSE, dims = 1)
mcPCP22<- ts(mcfcast2, frequency = 365, start = 2018-3/365)
PCP42 <- ts(na.approx(istorija22),frequency = 365, start = 2018-4/365)
PCP2<- ts(na.approx(istorija2), frequency = 365, start = 2017+2/365)
title(main = "Barclays Bank PLC")
ts.plot(mcPCP22, xlim= c(2017, 2018.033), ylim= c(25, 30), col="blue", ylab="Kaina, USD")
lines(PCP2, type = "l", col="black")
lines(PCP42, type = "l", col="black", lwd = 3)
mcCL2 <- apply(mcfcast2, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcPCP32 <- ts(t(mcCL2), frequency = 365, start = 2018-3/365)
lines(mcPCP32[, 1], col=c("red"), lwd = 3)
lines(mcPCP32[, 2], col=c("red"), lwd = 3)
mcvidurkiolinija2 <- ts(mcvidurkis2, frequency = 365, start = 2018-3/365)
lines(mcvidurkiolinija2, type = "l", col="orange", lwd = 3)
legend("topleft", legend = c("Istorija", "Pasikliautinas intervalas 95%", "Prognozės", "Vidutinė prognozė"), col
=c("black", "red", "blue", "orange"), lty = 1)
mcinvestportf <- 5*( mcakcijuvertes1 + mcakcijuvertes2)
MCCLinv <- apply(mcinvestportf, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
MCCLinv[2, ]-MCCLinv[1, ]
#MCMC modelis
reiksmes1 <- tail(arv2, 1000000-20000)
ilgis1 <- length(reiksmes1)
isrinkta1 <- reiksmes1[seq(1, ilgis2, 3)]
mcgrazos1 <- sample(isrinkta1)
hist(grazos1, probability = TRUE, breaks = 100, xlab="Graža", ylab ="Dažnis", main="Sugeneruotų gražų
histograma")
S <- 85.54
prognoze <- 18
stnr <- 15000
duomenys <- matrix(head(mcgrazos1, prognoze*stnr), nrow = prognoze, ncol = stnr)
akcijuvertes1 <- matrix(nrow = prognoze, ncol = stnr)

```

```

for (j in 1:stnr) {
  akcijuvertes1[1, j] <- S * duomenys[1, j] + S
  for (i in 2:prognose) {
    akcijuvertes1[i, j] <- akcijuvertes1[i-1, j] * duomenys[i, j] + akcijuvertes1[i-1, j]}
mcmcpadineverte1 <- matrix(S, nrow = 1, ncol = stnr)
mcmcfcast1 <- rbind(mcmcpadineverte1, akcijuvertes1)
mcmcfvidurkis1 <- rowMeans(mcmcfcast1, na.rm = FALSE, dims = 1)
mcmfPCP21 <- ts(mcmcfcast1, frequency = 365, start = 2018-3/365)
PCP41 <- ts(na.approx(istorija1), frequency = 365, start = 2018-3/365)
PCP1 <- ts(na.approx(istorija1), frequency = 365, start = 2017)
ts.plot(mcmfPCP21, xlim= c(2017, 2018.033), ylim= c(60, 110), col="blue", ylab="Kaina, USD", xlab="Laikas")
title(main = "Microsoft Corporation")
lines(PCP1, type = "l", col="black")
lines(PCP41, type = "l", col="black", lwd = 3)
mcCL1 <- apply(mcmcfcast1, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcmfPCP31 <- ts(t(CL1), frequency = 365, start = 2018-3/365)
lines(mcmfPCP31[, 1], col=c("red"), lwd = 3)
lines(mcmfPCP31[, 2], col=c("red"), lwd = 3)
mcmcfvidurkiolinija1 <- ts(mcmcfvidurkis1, frequency = 365, start = 2018-3/365)
lines(mcmcfvidurkiolinija1, type = "l", col="orange", lwd = 3)
legend("topleft", legend = c("Istorija", "Pasikliautinasis intervalas 95%", "Prognozės", "Vidutinė prognozė"), col
=c("black", "red", "blue", "orange"), lty = 1)
mcmcfvidurkiolinija1
rowVars(akcijuvertes1)
reiksmes2 <- tail(arv22, 1000000-20000)
ilgis2 <- length(reiksmes2)
isrinkta2 <- reiksmes2[seq(1, ilgis2, 3)]
mcgrazos2 <- sample(isrinkta2)
hist(mcgrazos2, probability = TRUE, breaks = 100, xlab="Grąža", ylab="Dažnis", main="Sugeneruotų grąžų
histograma")
S <- 26.67
prognose <- 18
stnr <- 15000
duomenys2 <- matrix(head(mcgrazos2, prognoze*stnr), nrow = prognoze, ncol = stnr)
akcijuvertes2 <- matrix(nrow = prognoze, ncol = stnr)
for (j in 1:stnr) {
  akcijuvertes2[1, j] <- S * duomenys2[1, j] + S
  for (i in 2:prognose) {
    akcijuvertes2[i, j] <- akcijuvertes2[i-1, j] * duomenys2[i, j] + akcijuvertes2[i-1, j]}
mcmcpadineverte2 <- matrix(S, nrow = 1, ncol = stnr)
mcmcfcast2 <- rbind(mcmcpadineverte2, akcijuvertes2)
mcmcfvidurkis2 <- rowMeans(mcmcfcast2, na.rm = FALSE, dims = 1)
mcmfPCP22 <- ts(mcmcfcast2, frequency = 365, start = 2018-3/365)
PCP42 <- ts(na.approx(istorija2), frequency = 365, start = 2018-3/365)
PCP2 <- ts(na.approx(istorija2), frequency = 365, start = 2017)
ts.plot(mcmfPCP22, xlim= c(2017, 2018.033), ylim= c(25, 30), col="blue", ylab="Kaina, USD", xlab="Laikas")
title(main = "Barclays Bank PLC")
lines(PCP2, type = "l", col="black")
lines(PCP42, type = "l", col="black", lwd = 3)
CL2 <- apply(mcmcfcast2, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcCL1 <- apply(mcmcfcast2, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)
mcmfPCP31 <- ts(t(CL2), frequency = 365, start = 2018-3/365)
lines(mcmfPCP31[, 1], col=c("red"), lwd = 3)
lines(mcmfPCP31[, 2], col=c("red"), lwd = 3)
mcmcfvidurkiolinija2 <- ts(mcmcfvidurkis2, frequency = 365, start = 2018-3/365)
lines(mcmcfvidurkiolinija2, type = "l", col="orange", lwd = 3)
legend("topleft", legend = c("Istorija", "Pasikliautinasis intervalas 95%", "Prognozės", "Vidutinė prognozė"), col
=c("black", "red", "blue", "orange"), lty = 1)
mcmcfvidurkiolinija2
mcmcinvestportfelio <- 5*(akcijuvertes1+akcijuvertes2)
mcmcinvport <- ts(mcmcinvestportfelio, frequency = 365, start = 2018-3/365)
ts.plot(mcmcinvport, xlim= c(2017, 2018.033), ylim= c(450, 750), col="blue", ylab="Verte, USD", xlab="Laikas")
CLinv <- apply(mcmcinvport, 1, quantile, probs = c(0.05, 0.95), na.rm = TRUE)

```


CLinv[2,]-CLinv[1,]