

<https://doi.org/10.15388/vu.thesis.831>
<https://orcid.org/0000-0003-3828-1592>

VILNIAUS UNIVERSITETAS

Vytautas Dulskis

Stochastinių dinaminių sistemų modelių parametrų efektyvių vertinimo algoritmų sudarymas ir taikymas

DAKTARO DISERTACIJA

Gamtos mokslai,
Informatika (N 009)

VILNIUS 2025

Disertacija rengta 2020–2024 metais Vilniaus universitete.

Mokslinis vadovas:

prof. habil. dr. Leonidas Sakalauskas (Vilniaus universitetas, gamtos mokslai, informatika – N 009).

Gynimo taryba:

Pirmininkė – prof. dr. Audronė Jakaitienė (Vilniaus universitetas, gamtos mokslai, informatika – N 009).

Nariai:

prof. dr. Romas Baronas (Vilniaus universitetas, gamtos mokslai, informatika – N 009),

prof. dr. Igoris Belovas (Vilniaus universitetas, gamtos mokslai, informatika – N 009),

prof. dr. Audrius Kabašinskas (Kauno technologijos universitetas, gamtos mokslai, informatika – N 009),

doc. dr. Tatiana Tchemisova (Aveiro universitetas, Portugalija, gamtos mokslai, matematika – N 001).

Disertacija ginama viešame Gynimo tarybos posėdyje 2025 m. rugsėjo 30 d. 12 val. Vilniaus universiteto Duomenų mokslo ir skaitemeninių technologijų instituto 203 auditorijoje. Adresas: Akademijos g. 4, LT-08412, Vilnius, Lietuva; tel.: +370 5 210 9300; el. paštas: info@mii.vu.lt.

Disertaciją galima peržiūrėti Vilniaus universiteto bibliotekoje ir Vilniaus universiteto interneto svetainėje adresu <https://www.vu.lt>.

<https://doi.org/10.15388/vu.thesis.831>
<https://orcid.org/0000-0003-3828-1592>

VILNIUS UNIVERSITY

Vytautas Dulskis

Development and Application of Efficient Algorithms for Parameter Estimation in Stochastic Dynamic System Models

DOCTORAL DISSERTATION

Natural Sciences,
Informatics (N 009)

VILNIUS 2025

The dissertation was prepared between 2020 and 2024 at Vilnius University.

Academic supervisor – Prof. Habil. Dr. Leonidas Sakalauskas (Vilnius University, Natural Sciences, Informatics – N 009).

This doctoral dissertation will be defended in a public meeting of the Dissertation Defense Panel:

Chair – Prof. Dr. Audronė Jakaitienė (Vilnius University, Natural Sciences, Informatics – N 009).

Members:

Prof. Dr. Romas Baronas (Vilnius University, Natural Sciences, Informatics – N 009),

Prof. Dr. Igoris Belovas (Vilnius University, Natural Sciences, Informatics – N 009),

Prof. Dr. Audrius Kabašinskas (Kaunas University of Technology, Natural Sciences, Informatics – N 009),

Assoc. Prof. Dr. Tatiana Tchemisova (University of Aveiro, Portugal, Natural Sciences, Mathematics – N 001).

The dissertation shall be defended at a public meeting of the Dissertation Defense Panel at noon on 30th September 2025 in room 203 of the Institute of Data Science and Digital Technologies of Vilnius University. Address: Akademijos g. 4, LT-08412, Vilnius, Lithuania;
Tel.: +370 5 210 9300; E-mail: info@mii.vu.lt.

The text of this dissertation can be accessed at the Library of Vilnius University, as well as on the website of Vilnius University: <https://www.vu.lt>.

PADĖKA

Nuoširdžiai dėkoju savo darbo vadovui prof. habil. dr. Leonidui Sakalauskui už profesionalų vadovavimą, vertingas mokslines išvalgas ir nuolatinę pagalbą rengiant šį darbą. Esu dėkingas disertacijos recenzentams prof. dr. Igoriui Belovui ir prof. dr. Romui Baronui už konstruktyvią kritiką ir naudingas pastabas, padėjusias tobulinti disertaciją. Taip pat reiškiu padėką Vilniaus universiteto Duomenų mokslo ir skaitmeninių technologijų instituto kolegoms už bendradarbiavimą, dalijimąsi patirtimi ir suteiktą pagalbą visų studijų metu.

Dėkoju savo artimiesiems ir draugams už supratingumą, paramą ir kantrybę.

Taip pat dėkoju visiems, kurie tiesiogiai ar netiesiogiai prisidėjo prie šio darbo.

Vytautas Dulskis

SANTRAUKA

Disertacijoje akcentuojamas didžiųjų duomenų eroje kylantis poreikis kurti tokius stochastinių dinaminių sistemų modelių parametrų vertinimo algoritmus, kurie ne tik teiktų optimalius parametrų įverčius sistemos stebėjimo duomenų atžvilgiu, bet ir pasižymėtų efektyviu skaičiavimu, leidžiančiu juos taikyti beveik bet kokio dydžio sistemos stebėjimo duomenų imtims. Šie algoritmai suprantami kaip pagrįsti didžiausio tikėtimumo metodu ir rekursiniais skaičiavimais, kurie metodologiškai apibendrinami kaip palaipsninio vertinimo technika (PVT). Atsižvelgiant į tai, kad tokių algoritmų sudarymo sudėtingumas didėja kartu su modelio kompleksiskumu, PVT taikymas grindžiamas principu „iš apačios į viršų“, kuris numato algoritmų kūrimą atskiriems bendrojo dinaminių sistemų modelio atvejams ir palaipsninį judėjimą link bendresnių modelių.

Šiuo pagrindu disertacijoje plačiau nagrinėjamas dinaminis struktūrinių lygčių modelis, kurio santykinai paprastu, tačiau fundamentaliu atskiru atveju inicijuojamas PVT taikymas. Šio atvejo tyrimas atskleidžia PVT duomenų transformavimo, kovariacijų matricos blokinio skaidymo, tikėtimumo funkcijos dimensijos sumažinimo ir įverčių rekursinio apskaičiavimo etapus. Sėkmingas šių etapų įgyvendinimas leidžia sudaryti tiesinio sudėtingumo algoritmą, skirtą nagrinėjamo modelio parametrų įvertinimui didžiausio tikėtimumo metodu, kai duomenų rinkinys yra fiksuotas. Lyginant su alternatyvomis, šio algoritmo potencialas ypač atsiskleidžia esant intensyviai modelio išvesties stebėjimo triukšmui. Atlikus PVT papildančius veiksmus, sudarytas algoritmas transformuojamas į analogiškų savybių algoritmą, tinkamą parametrų vertinti realiuoju laiku.

Nors sudarytus algoritmus nesudėtinga įvykdyti naudojant kompiuterinę įrangą ir taip leisti juos taikyti, modelio parametrų vertinimas nėra tapatus visam dinaminių sistemų modeliavimo procesui. Šiuo požiūriu per COVID-19 pandemijos ir socialinio kapitalo modeliavimo prizmę disertacijoje atskleidžiami praktiniai taikymo aspektai, susiję su adekvataus tiriamam reiškiniui modelio, suderinamo su prieinamais sistemos stebėjimo duomenimis, parinkimu.

SANTRUMPOS

SLM	struktūrinių lygčių modeliavimas
DSLM	dinaminis struktūrinių lygčių modelis
KSLM	kumuliacinis struktūrinių lygčių modelis
PVT	palaipsninio vertinimo technika
DTM	didžiausio tikėtinumo metodas
VAP	vidutinė absoliutinė paklaida
min.	minimumas
vid.	vidurkis
med.	mediana
maks.	maksimumas
sn	standartinis nuokrypis
SIRD	segmentinis modelis, susidedantis iš imliųjų (angl. <i>Susceptible</i>), užkrėstųjų (angl. <i>Infected</i>), pasveikusiųjų (angl. <i>Recovered</i>) ir mirusiųjų (angl. <i>Deceased</i>) segmentų
EBPO	Ekonominio bendradarbiavimo ir plėtros organizacija
PSK	potencialus socialinis kapitalas
SKVS	socialinio kapitalo vystymosi santykis

ŽYMĖJIMAI

I	vienetinė matrica
$M_{i \times j}$	išreikštinis nurodymas, kad matrica M susideda iš i eilučių ir j stulpelių
$a_{i \times j}$	i eilučių ir j stulpelių matrica, kurios visi elementai yra lygūs skaliarui a
$[M]_{i,j}$	matricos M elementas, esantis i eilutėje ir j stulpelyje
$[M]_{(i_a:i_b),(j_a:j_b)}$	matricos M elementai, esantys nuo i_a eilutės iki i_b eilutės (įskaitant) ir nuo j_a stulpelio iki j_b stulpelio (įskaitant)
$[M],$	matricos M elementai, esantys nuo pirmos iki paskutinės eilutės (įskaitant) ir nuo pirmo iki paskutinio stulpelio (įskaitant)
M'	matricos M transpozicija
$\text{vec}(M)$	matricos M vektorizacija
$\text{diag}(v)$	įstrižaininė matrica, sudaryta iš vektoriaus v elementų
$\text{diag}(M)$	vektorius, sudarytas iš matricos M įstrižaininių elementų
$\text{tr}(M)$	matricos M pėdsakas
$\lambda[M]$	matricos M tikrinių reikšmių vektorius
$v[M]$	matricos M tikrinių vektorių matrica
$\otimes$	Kronekerio sandauga
$\textcircled{\otimes}$	kėlimas eksponentiniu laipsniu paelemenčiui
$\odot$	daugyba paelemenčiui
$\oslash$	dalyba paelemenčiui
E	matematinė viltis
C	kovariacijų matrica
τ	laiko momentų skirtumas
$\mathbb{Z}$	sveikųjų skaičių aibė
$\mathbb{N}_0$	natūraliųjų skaičių aibė, įskaitant 0
$\sum_{k=1}^K$	sumavimas pagal indeksą k nuo 1 iki K
$\mathcal{N}(v, M)$	normalusis (Gauso) skirstinys su vidurkių vektoriumi v ir kovariacijų matrica M
$\frac{\partial f}{\partial x}$	funkcijos f išvestinė pagal kintamąjį x

TURINYS

PADĖKA	5
SANTRAUKA	6
SANTRUMPOS	7
ŽYMĖJIMAI	8
IVADAS	18
Tyrimų sritis	18
Problemos aktualumas	18
Tyrimų objektas	21
Darbo tikslas ir uždaviniai	21
Tyrimų metodai	22
Mokslinis darbo naujumas	22
Praktinė darbo reikšmė	23
Ginamieji teiginiai	23
Tyrimų aprobavimas ir publikavimas	24
Disertacijos struktūra	25
1. STOCHASTINIŲ DINAMINIŲ SISTEMŲ MODELIŲ PARAMETRŲ EFEKTYVIŲ VERTINIMO ALGORITMŲ SUDARYMO TECHNIKA	26
1.1. Struktūrinių lygčių modeliavimas	26
1.2. Dinaminis struktūrinių lygčių modelis	27
1.3. Efektyvūs parametrų vertinimo algoritmai	29
1.4. Parametrų vertinimas realiuoju laiku	33
1.5. Skyriaus išvados	35
2. PALAIPSNINIO VERTINIMO TECHNIKOS TAIKYMAS KU- MULIACINIO STRUKTŪRINIŲ LYGČIŲ MODELIO ATVEJU	37
2.1. Problemos formuluotė	39
2.2. Problemos sprendimas	42
2.2.1. Tikėtinumo funkcija	43

2.2.2.	Dalinės išvestinės	46
2.2.3.	Didžiausio tikėtinumo įvertiniai	48
2.2.4.	Rekursiniai skaičiavimai	50
2.2.5.	Vertinimo algoritmas	52
2.3.	Empiriniai rezultatai	52
2.4.	Skyriaus išvados	61
3.	ALGORITMO ADAPTAVIMAS PARAMETRAMS VERTINTI REALIUOJU LAIKU	63
3.1.	Bendrasis modelis	65
3.1.1.	Tiesioginė tikėtinumo funkcija	65
3.1.2.	Pertvarkyta tikėtinumo funkcija	65
3.1.3.	Išskaidyta tikėtinumo funkcija	67
3.2.	Vienmatės išvesties atvejis	68
3.2.1.	Tikėtinumo funkcija	68
3.2.2.	Dalinė išvestinė	69
3.2.3.	Rekursiniai skaičiavimai	71
3.2.4.	Asimptotinės aproksimacijos	74
3.2.5.	Didžiausio tikėtinumo įvertiniai	75
3.2.6.	Vertinimo algoritmas	76
3.3.	Empiriniai rezultatai	78
3.4.	Diskusija	82
3.5.	Skyriaus išvados	84
4.	PRAKTINIAI ASPEKTAI	86
4.1.	COVID-19 pandemijos modeliavimas	86
4.1.1.	Duomenys ir modelis	93
4.1.2.	Rezultatai	96
4.1.3.	Diskusija	100
4.2.	Socialinio kapitalo modeliavimas	101

4.3. Skyriaus išvados	105
BENDROSIOŠ IŠVADOS	106
LITERATŪROS SĄRAŠAS	108
PRIEDAS	122
R paketas EMLI	122
SUMMARY IN ENGLISH	129
Research Area	129
Research Aim and Objectives	129
Research Methods	130
Scientific Novelty	130
Defended Statements	131
Research Dissemination and Publications	131
Structure of the Dissertation	132
S.1. Technique for Constructing Efficient Parameter Estima- tion Algorithms for Stochastic Dynamic System Models .	133
S.1.1. Dynamic Structural Equation Model	133
S.1.2. Efficient Parameter Estimation Algorithms	135
S.1.3. Real-Time Parameter Estimation	137
S.1.4. Section Conclusions	138
S.2. Application of the Stepwise Estimation Technique to the Cumulative Structural Equation Model	138
S.2.1. Model	139
S.2.2. Likelihood Function	140
S.2.3. Partial Derivatives	142
S.2.4. Maximum Likelihood Estimators	143
S.2.5. Recursive Computations	145
S.2.6. Estimation Algorithm	146
S.2.7. Empirical Results	147
S.2.8. Section Conclusions	154

S.3. Adapting the Algorithm for Real-Time Parameter Estimation	154
S.3.1. Likelihood Function	154
S.3.2. Partial Derivative	155
S.3.3. Recursive Computations	157
S.3.4. Asymptotic Approximations	159
S.3.5. Maximum Likelihood Estimators	161
S.3.6. Estimation Algorithm	162
S.3.7. Empirical Results	163
S.3.8. Section Conclusions	167
S.4. Practical Aspects	167
S.4.1. Modeling the COVID-19 Pandemic	168
S.4.2. Data and Model	170
S.4.3. Results	172
S.4.4. Discussion	176
S.4.5. Section Conclusions	176
General Conclusions	177

LENTELIŲ SĄRAŠAS

2.1	2.1 algoritmo konvergavimo patikrinimas esant skirtingiems modelio parametrams.	55
-----	---	----

PAVEIKSLĖLIŲ SĄRAŠAS

1	Mokslinės produkcijos <i>Web of Science (Clarivate Analytics)</i> duomenų bazėje dinamika pagal užklausą <i>dynamic model (Topic) and parameter estimation (Topic) and efficient algorithm (Topic)</i>	21
1.1	Supaprastinta abstrakti SLM diagrama, kurioje x (stebima įvestis) ir y (stebima išvestis) yra stebimi kintamieji, o ξ (latentinė įvestis) ir η (latentinė išvestis) yra latentiniai kintamieji. Rodyklės rodo poveikio kryptį. Tiek x kartu su ξ , tiek y kartu su η sudaro matavimo modelius, o η kartu su ξ sudaro struktūrinį modelį. Daugiau informacijos apie SLM diagramų braižymą galima rasti [62].	27
2.1	2.1 algoritmo konvergavimo patikrinimas esant skirtingiems modelio parametrų rinkiniams: (a) parametrų rinkinys Nr. 1; (b) parametrų rinkinys Nr. 2; (c) parametrų rinkinys Nr. 3; (d) parametrų rinkinys Nr. 4; (e) parametrų rinkinys Nr. 5; (f) parametrų rinkinys Nr. 6.	54
2.2	2.1 algoritmo lyginimas su 2.2 algoritmu esant skirtingiems stebėjimų imties dydžiams: (a) $T = 10$; (b) $T = 20$; (c) $T = 50$; (d) $T = 100$; (e) $T = 1000$; (f) $T = 10000$	59
2.3	Algoritmų vidutinių vykdymo trukmių lyginimas esant skirtingiems stebėjimų imties dydžiams: (a) 2.1 algoritmas (A2.1) ir tiesioginis (2.6) tikėtinumo funkcijos minimavimas (A0); (b) 2.1 algoritmas (A2.1) ir 2.2 algoritmas (A2.2).	61

3.1	Empirinis 3.1 algoritmo (A3.1) patikrinimas (kreivės brėžiamos nuo $T = 100$): (a) konvergavimas parametru poaibio Nr. 1 atžvilgiu; (b) lyginimas su 3.2 algoritmu (A3.2) parametru poaibio Nr. 1 atžvilgiu; (c) konvergavimas parametru poaibio Nr. 2 atžvilgiu; (d) lyginimas su 3.2 algoritmu (A3.2) parametru poaibio Nr. 2 atžvilgiu; (e) konvergavimas parametru poaibio Nr. 3 atžvilgiu; (f) lyginimas su 3.2 algoritmu (A3.2) parametru poaibio Nr. 3 atžvilgiu.	81
3.2	Skirtumas tarp vidutinių (3.14) tikėtimumo funkcijos reikšmių, apskaičiuotų naudojant atitinkamai 3.1 ir 2.1 algoritmų įverčius (pagal vidutinio išvesties stebėjimo triukšmo scenarijų ir 100 duomenų imčių).	82
4.1	Patvirtinti COVID-19 susirgimai (nauji per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.	89
4.2	COVID-19 testai (nauji per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d. . . .	90
4.3	Teigiamų COVID-19 testų dalis Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.: (a) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 0 dienų į ateitį COVID-19 testų (naujų per dieną) atžvilgiu; (b) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 1 dieną į ateitį COVID-19 testų (naujų per dieną) atžvilgiu; (c) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 2 dienas į ateitį COVID-19 testų (naujų per dieną) atžvilgiu.	90
4.4	Patvirtinti COVID-19 susirgimai (7 dienų slenkantis vidurkis) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.	91
4.5	Patvirtintos COVID-19 mirtys (naujos per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.	92

4.6	Patvirtintos COVID-19 mirtys (naujos per dieną) milijonui gyventojų Lietuvoje (padalinta iš milijono) laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.	94
4.7	COVID-19 galimybių paso įvedimo įtakos delta bangai plisti Lietuvoje modeliavimas: (a) įvertintas modelis; (b) hipotetinis modelis; (c) įvertinto ir hipotetinio modelių lyginimas.	98
4.8	COVID-19 galimybių paso nutraukimo įtakos omikrono bangai plisti Lietuvoje modeliavimas: (a) įvertintas modelis; (b) hipotetinis modelis; (c) įvertinto ir hipotetinio modelių lyginimas.	100
4.9	Pavyzdinis (4.12) ir (4.13) lygtimis apibrėžiamo modelio įgyvendinimas suprantamas kaip individui būdinga kokio nors socialinio kapitalo indekso dinamika laike, kuris apskaitomas įvykusiais kultūriniais įvykiais.	105
A.1	Realių duomenų ir pagal įvertintą modelį sugeneruotų duomenų panašumas (matuojamas remiantis tikėtumo funkcija).	127
S.1	Verification of the convergence of Algorithm S.1 under different model parameters: (a) parameter set No. 1; (b) parameter set No. 2; (c) parameter set No. 3; (d) parameter set No. 4; (e) parameter set No. 5; (f) parameter set No. 6.	149
S.2	Comparison of Algorithm S.1 with Algorithm S.2 for different sample sizes: (a) $T = 10$; (b) $T = 20$; (c) $T = 50$; (d) $T = 100$; (e) $T = 1,000$; (f) $T = 10,000$	152
S.3	Comparison of average algorithm runtimes for different sample sizes: (a) Algorithm S.1 (AS.1) versus direct minimization of the likelihood function defined by Equation (S.9) (A0); (b) Algorithm S.1 (AS.1) versus Algorithm S.2 (AS.2).	153

S.4	Empirical verification of Algorithm S.3 (AS.3) (curves plotted from $T = 100$): (a) convergence for parameter subset No. 1; (b) comparison with Algorithm S.4 (AS.4) for parameter subset No. 1; (c) convergence for parameter subset No. 2; (d) comparison with Algorithm S.4 (AS.4) for parameter subset No. 2; (e) convergence for parameter subset No. 3; (f) comparison with Algorithm S.4 (AS.4) for parameter subset No. 3.	166
S.5	Daily new COVID-19 deaths per million population in Lithuania (divided by one million) from August 7, 2021, to May 8, 2022.	171
S.6	Modeling the impact of the COVID-19 passport introduction on the spread of the Delta wave in Lithuania: (a) estimated model; (b) hypothetical model; (c) comparison of estimated and hypothetical models.	174
S.7	Modeling the impact of the COVID-19 passport expiration on the spread of the Omicron wave in Lithuania: (a) estimated model; (b) hypothetical model; (c) comparison of estimated and hypothetical models.	175

IVADAS

Tyrimų sritis

Didžiųjų duomenų era atveria naujų galimybių modeliuoti sudėtingų gamtinių, inžinerinių ir socialinių sistemų dinamiką [91]. Vis dėlto, siekiant maksimaliai išnaudoti šias galimybes, būtina įveikti iššūkius, susijusius su gebėjimu efektyviai apdoroti vis didėjantį informacijos kiekį [8].

Formaliai dinaminė sistema gali būti apibrėžta taisyklių rinkiniu, nusakančiu, kaip laikui bėgant kinta sistemos būsena. Šios taisyklės gali būti išreiškiamos skirtuminėmis lygtimis diskrečioms sistemoms arba diferencialinėmis lygtimis tolydžioms sistemoms. Tokiu atveju dinaminės sistemos būsena bet kuriuo momentu yra taškas galimų būsenų erdvėje, o jos raida – trajektorija šioje erdvėje [34].

Modelis, kuriuo aprašoma dinaminė sistema, apibrėžia galimas trajektorijas, o jo parametrų reikšmės jas konkretizuoja (deterministinis modelis su nustatytomis parametrų reikšmėmis aprašo vieną konkrečią trajektoriją, o stochastinis – tikimybinį trajektorijų pasiskirstymą [77]). Taigi tiriamo reiškinių raidos modeliavimas yra neatsiejamas nuo tinkamo modelio parinkimo ir tikslaus jo parametrų įvertinimo. Šis savo ruožtu reikalauja tiek gausių sistemos stebėjimo duomenų, tiek efektyvių algoritmų, leidžiančių per praktiškai priimtina laiką maksimaliai išnaudoti sukauptų duomenų informaciją. Šių algoritmų kūrimo problematika tikslingai nagrinėtina remiantis teorinėmis dinaminių sistemų modelių formuluotėmis, o modelių parinkimo bei su duomenimis susiję klausimai geriausiai atsiskleidžia konkrečių praktinių taikymų kontekste.

Problemos aktualumas

Remiantis [20], diskretauso laiko dinaminių sistemų modeliai bendruoju (netiesiniu) atveju gali būti formuluojami taip:

$$\eta_t = f(\eta_{t-1}, \dots, \eta_{t-1-p}, \xi_t, \dots, \xi_{t-q}), \quad (1)$$

$$y_t = g(\eta_t), \quad (2)$$

$$x_t = h(\xi_t), \quad (3)$$

čia η yra latentinis (nestebimas) modelio išvesties kintamasis, ξ – latentinis (nestebimas) modelio įvesties kintamasis, y – stebimas modelio išvesties kintamasis (t. y. η matavimas), x – stebimas modelio įvesties kintamasis (t. y. ξ matavimas); f, g, h yra funkcijos, parametrizuojamos parametru rinkiniais $\theta_f, \theta_g, \theta_h$ ir nusakančios ryšius tarp modelio kintamųjų; indeksas $t \in \mathbb{Z}$ žymi laiko momentą ($p \in \mathbb{N}_0, q \in \mathbb{N}_0$).

(1)–(3) lygtimis aprašytas modelis per diskretaus laiko vienetą priima fiksuotą informacijos kiekį ir apibrėžia tiesiogiai nestebimų reiškinių arba abstrakčių konstruktų, įgyjančių praktinį pavidalą per išmatuojamus rodiklius [127], dinamikos, kuri gali pasižymėti netiesiniu elgesiu, ilga atmintimi ir stochastiškumu, modeliavimo struktūrą. Tiriamo reiškinio raidai analizuoti ir prognozuoti reikalingos jo stebėjimo duomenis atitinkančios funkcijos f, g, h su parametru rinkinių $\theta_f, \theta_g, \theta_h$ reikšmėmis. Netinkamai parinkus modelį, gali nepavykti atskleisti nagrinėjamo reiškinio subtilybių arba galima klaidingai interpretuoti objektyvias aplinkybes, o netikslūs parametru įverčiai neišvengiamai apriboja net ir korektiško, modeliavimo tikslus atitinkančio, modelio potencialą [24, 70].

Pasirinkto modelio parametru įverčių tikslumas priklauso nuo turimų stebėjimo duomenų apimties ir naudojamo parametru vertinimo metodo galimybių išnaudoti jų informaciją. Didesni duomenų masyvai saugo daugiau informacijos, taip leisdami tiksliau įvertinti parametrus, tačiau jų apdorojimas reikalauja daugiau skaičiavimų elementariųjų operacijų pavidalu. Savo ruožtu maksimalios įmanomos informacijos iš duomenų gavimas yra sietinas su santykinai sudėtingais parametru vertinimo metodais, kurių įgyvendinimas taip pat yra imlus skaičiuojamųjų požiūriu [20]. Taigi dinaminių sistemų modelių taikymams didžiųjų duomenų eros kontekste reikalingi efektyvūs algoritmai, leidžiantys įvykdyti optimalius parametru vertinimo metodus esant kuo didesniam stebėjimo momentų skaičiui, o tai praktiškai reiškia, kad skaičiavimams atlikti reikalingas laikas neturėtų ilgėti sparčiau nei tiesiškai duomenų kiekio atžvilgiu (tai yra algoritmo sudėtingumo reikalavimas; du vieno sudėtingumo algoritmai gali skirtis savo konstantomis, o tai taip pat sudaro minimizavimo objektą, tačiau konkrečius reikalavimus tam numatyti nėra prasminga). O esamais algoritmais dažnai negalima vienu metu suderinti maksimalios parametru įverčių kokybės su reikalinga greitimeika, nes jie paremti arba parametru vertinimo atžvilgiu (asimp-

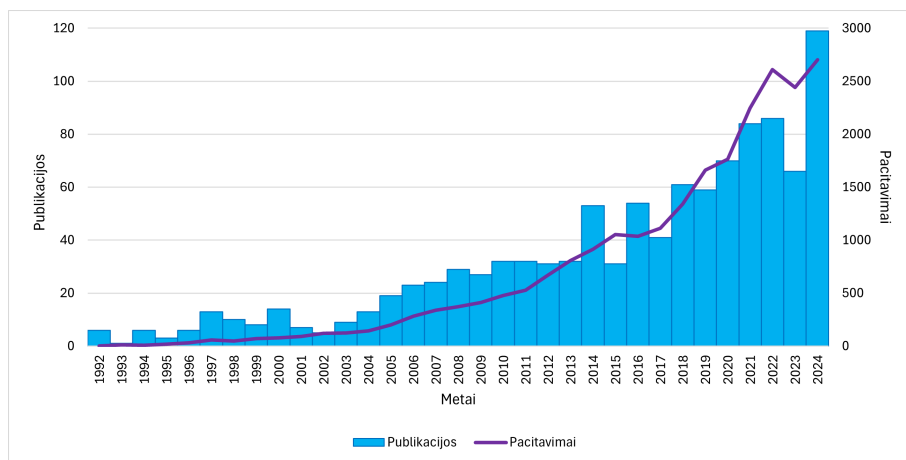
totiškai) optimaliais, tačiau skaičiavimų atžvilgiu reikliais metodais (tokiais kaip didžiausio tikėtimumo metodas [20]), arba lankstesniais, tačiau mažiau tiksliais metodais (tokiais kaip momentų metodas [46] ar stochastiniai metodai [71]).

Idealiu atveju taip suprantamas efektyvus parametru vertinimo algoritmas yra reikalingas bendrajam dinaminių sistemų (1)–(3) lygtimis apibrėžtam modeliui, nes leistų modeliuoti dinamines sistemas ypač plačiu modelių spektru, patenkinančiu daugumos praktikoje sutinkamų reiškinių dinamikos modeliavimo poreikius (svarbiausia tada būtų parinkti sistemos stebėjimo duomenis paaiškinančią modelį arba sudaryti modeliavimo tikslus atitinkančią teorinį modelį ir surinkti jį atitinkančius duomenis – modelių ir duomenų suderinimas yra svarbus praktinis uždavinys [53]). Tačiau tenka konstatuoti, kad tokio algoritmo sukūrimas yra veikiau siekiamybė, o ne reali galimybė, nes funkcijos f , g , h gali kardinaliai skirtis savo sudėtingumu. Be to, ne viskas, kas pavyksta su paprastesniais modeliais, pavyks ir su sudėtingesniais. Pats parametru vertinimo procesas gali skirtis savo dizainu – jis gali būti grindžiamas iš anksto surinktu duomenų rinkiniu arba vykdomas realiuoju laiku, su kiekvienu nauju stebėjimu nuolat atnaujinant įverčius. Šis parametru vertinimo būdas dažnai yra sudėtingesnis, nes realaus laiko algoritmai dėl savo specifikos negali naudoti visos augančios duomenų aibės, todėl jų kūrimas susijęs su didesniais iššūkiais [116, 120]. Taigi, atsižvelgiant į šiuos pastebėjimus, efektyvius parametru vertinimo algoritmus tenka kurti atskiroms (1)–(3) lygtimis nusakomo modelio atveju ir parametru vertinimo dizainų kombinacijoms, taip siekiant palaipsniui kuo labiau išplėsti kokybiškų modeliavimo galimybių spektrą.

Vienas fundamentalią atskirą klasę sudarantis bendrojo modelio atvejis, reikalaujantis efektyvių parametru vertinimo algoritmų, yra gautas funkcijas f , g , h laikant tiesinėmis. Šis atvejis dar žinomas kaip dinaminis struktūrinių lygčių modeliavimas [20] ir yra plačiai naudojamas įvairiose srityse, tarp kurių daug dėmesio skiriama socialiniams mokslams [23, 41, 49, 112]. Iš esmės struktūrinių lygčių modelių poreikis pasireiškia analizuojant sudėtingus ryšius tarp stebimų kintamųjų, kurie yra tiesiogiai matuojami, ir latentinių kintamųjų, tai yra konstrukto, kurie negali būti išmatuoti tiesiogiai ir todėl turi būti išvedami iš kitų kintamųjų [15, 93]. Savo ruožtu dinamikos įvedimas leidžia modeliuoti kintamųjų ryšius laiko dimensijos atžvilgiu, taip išplečiant šių modelių

galimybes išilginiams duomenims [132].

Efektyvaus dinaminių modelių parametų vertinimo problemos aktualumas vis plačiau pripažįstamas tarptautinėje mokslinėje bendruomenėje – susidomėjimas šia tematika akivaizdžiai auga (žr. 1 pav.). Prie šios srities plėtros aktyviai prisideda ir Lietuvos mokslininkai, kurių darbas sėkmingai atneša naujų rezultatų [12, 116, 117].



1 pav.: Mokslinės produkcijos *Web of Science* (*Clarivate Analytics*) duomenų bazėje dinamika pagal užklausą *dynamic model (Topic) and parameter estimation (Topic) and efficient algorithm (Topic)*.

Tyrimų objektas

Disertacijos tyrimų objektas – stochastinių dinaminių sistemų modeliai, o pagrindinis dėmesys skiriamas jų efektyviems parametų vertinimo algoritmams. Taip pat nagrinėjamas praktinis modelių, suderinamų su tiriamu reiškiniu ir prieinamais duomenimis, parinkimas.

Darbo tikslas ir uždaviniai

Darbo tikslas – nusakyti ir pademonstruoti efektyvų stochastinių dinaminių sistemų modelių parametų vertinimą bei išryškinti šių modelių praktinio taikymo aspektus.

Siekiant šio tikslo, paeiliui sprendžiami šie uždaviniai:

1. Stochastinių dinaminių sistemų modelių parametų efektyvių vertinimo algoritmų sąlygų ir sudarymo žingsnių nustatymas.

2. Parametrų vertinimo, kai duomenų rinkinys yra fiksuotas, efektyvaus algoritmo sudarymas dinaminio struktūrinių lygčių modelio atveju.
3. Sudaryto parametrų vertinimo, kai duomenų rinkinys yra fiksuotas, algoritmo transformavimas į parametrų vertinimo realiuoju laiku algoritmą.
4. Stochastinių dinaminių sistemų modelių pritaikymas realių procesų dinamikai modeliuoti.

Tyrimų metodai

Disertacijos uždaviniai sprendžiami taikant daugiamatės statistikos, tiesinės algebros ir kompiuterinio modeliavimo metodus. Efektyvūs stochastinių dinaminių sistemų modelių parametrų vertinimo algoritmai kuriami naudojant didžiausio tikėtimumo metodą ir rekursiją per duomenų transformavimą, matricų skaidymą bei funkcijų pertvarkymą. Sukurti algoritmai ištiriami eksperimentiškai, naudojant Monte Karlo metodą.

Mokslinis darbo naujumas

Disertacijoje gauti šie rezultatai:

1. Pasiūlyta palaipsninio vertinimo technika, nusakanti stochastinių dinaminių sistemų modelių parametrų efektyvių vertinimo algoritmų sudarymo metodologiją.
2. Apibrėžtas kumuliacinis struktūrinių lygčių modelis ir jam sudarytas didžiausio tikėtimumo parametrų vertinimo, kai duomenų rinkinys yra fiksuotas, algoritmas, kurio sudėtingumas stebėjimo momentų skaičiaus atžvilgiu yra tiesinis.
3. Kumuliaciniam struktūrinių lygčių modeliui sudarytas didžiausio tikėtimumo parametrų vertinimo realiuoju laiku algoritmas.
4. Remiantis COVID-19 pandemijos ir socialinio kapitalo dinamikos modeliavimo pavyzdžiais, pademonstruotas stochastinių dinaminių sistemų modelių, suderinamų su tiriamu reiškiniu ir prieinamais duomenimis, parinkimas.

Praktinė darbo reikšmė

Disertacijoje sudaryti efektyvūs kumuliacinio struktūrinių lygčių modelio parametrų vertinimo algoritmai, kai duomenų rinkinys yra fiksuotas ir pateikiamas realiuoju laiku, leidžia šį modelį taikyti grynųjų laiko eilučių duomenims visomis parametrų vertinimo situacijomis ir gauti konkurencingus parametrų įverčius, paremtus asimptotiškai optimaliu parametrų vertinimo metodu, kurio pranašumas ryškėja didėjant santykiniam modelio išvesties stebėjimo triukšmui. Šių algoritmų sudarymas metodologiškai apibrėžto proceso rėmuose leidžia ieškoti analogiškai efektyvių algoritmų palaipsniui vis bendresniems modeliams, o tai sudaro pagrindą formuluoti ir bandyti įgyvendinti spėjimus apie dinaminių struktūrinių lygčių bei platesnių stochastinių dinaminių sistemų modelių optimalų parametrų vertinimą skaitiškai efektyviais algoritmais.

Nors efektyvių parametrų vertinimo algoritmų sudarymas nėra nei paprastas, nei akivaizdus, jų pritaikymas turimiems įvesties duomenims paprastai yra nesudėtingas. Tačiau tiriant konkrečius praktinius fenomenus, ne visada aišku, kokie tai duomenys. Todėl disertacijoje nagrinėjami COVID-19 pandemijos bei socialinio kapitalo dinamikos modeliavimo pavyzdžiai atskleidžia praktinius dinaminių sistemų modelių taikymo aspektus, susijusius su adekvataus tiriamam reiškiniui modelio parinkimu ir galimu modelio atotrūkiu nuo prieinamų sistemos stebėjimo duomenų.

Ginamieji teiginiai

- Palaipsninio vertinimo technika nusako efektyvių algoritmų, skirtų stochastinių dinaminių sistemų modelių parametrams vertinti, kai duomenų rinkinys yra fiksuotas, sudarymo gaires.
- Darbe sudaryti kumuliacinio struktūrinių lygčių modelio parametrų vertinimo, kai duomenų rinkinys yra fiksuotas ir pateikiamas realiuoju laiku, algoritmai sujungia nesudėtingų, tačiau mažiau tikslių, ir tikslesnių, tačiau sudėtingesnių klasikinių parametrų vertinimo metodų privalumus.
- Oficialiai prieinami COVID-19 duomenys nesuteikia pakankamai informacijos, susiejančios testavimo, užsikrėtimo ir mirtingumo

duomenis individo lygmeniu, todėl sudėtinga vienareikšmiškai įvertinti galimybių paso poveikį pandemijos dinamikai Lietuvoje.

Tyrimų aprobavimas ir publikavimas

Pranešimai tarptautinėse konferencijose:

1. Efficient Online Estimation of Dynamic Structural Equation Models: Conjecturing on Feasibility. The 3rd International Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 18–20 September 2024, Almaty (Kazakhstan).
2. On the Gap Between Data and Models in COVID-19 Analysis. The 33rd European Conference on Operational Research, 30 June – 3 July 2024, Copenhagen (Denmark).
3. Efficient Maximum Likelihood Batch Estimation With Pure Time Series Data of a One-Dimensional Cumulative Structural Equation Model. The 2nd International Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 21–23 September 2022, Vilnius (Lithuania).
4. Incremental Maximum Likelihood Estimation of Noisy Gaussian Random Walk. The 31st European Conference on Operational Research, 11–14 July 2021, Athens (Greece).
5. Probabilistic Model of Cultural Participation Impact on Social Capital. The 1st International & EURO Mini Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 18–20 September 2019, Vilnius (Lithuania).

Publikacijos leidiniuose su cituojamumo rodikliu *Web of Science (Clarivate Analytics)* duomenų bazėje:

1. Vytautas Dulskis & Leonidas Sakalauskas (2025). Toward Efficient Online Estimation of Dynamic Structural Equation Models: A Case Study. *Journal of Statistical Computation and Simulation*, 1–24. DOI: 10.1080/00949655.2025.2515955.
2. Leonidas Sakalauskas, Vytautas Dulskis, & Rimas Jonas Jankūnas (2024). Compartmental Modeling for Pandemic Data Analysis:

The Gap Between Statistics and Models. *Heliyon*, 10(11):e31410. DOI: 10.1016/j.heliyon.2024.e31410.

3. Leonidas Sakalauskas, Vytautas Dulskis, & Darius Plikynas (2024). A Technique for Efficient Estimation of Dynamic Structural Equation Models: A Case Study. *Structural Equation Modeling: A Multidisciplinary Journal*, 31(4), 635–650. DOI: 10.1080/10705511.2023.2282378.
4. Leonidas Sakalauskas, Vytautas Dulskis, Rimvydas Laužikas, Arūnas Miliauskas, & Darius Plikynas (2021). A Probabilistic Model of the Impact of Cultural Participation on Social Capital. *The Journal of Mathematical Sociology*, 45(2), 65–78. DOI: 10.1080/0022250X.2020.1725002.

Disertacijos autoriaus indėlis į kiekvieną iš nurodytų publikacijų apima kertinių rezultatų (modelių, algoritmų) sukūrimą, kompiuterinių skaičiavimų atlikimą bei teksto parengimą.

Disertacijos struktūra

Disertaciją sudaro įvadas, keturi pagrindiniai skyriai, bendrosios išvados, literatūros sąrašas ir priedas.

Pagrindinėje disertacijos dalyje, susidedančioje iš keturių skyrių, nuosekliai pateikiami disertacijos rezultatai, kurie derinami su pastabomis, išryškinančiomis esminius teiginius ir sudarančiomis platesnį kontekstą. Pirmajame skyriuje per dinaminio struktūrinių lygčių modelio prizmę apžvelgiama aktuali literatūra, išgryninama efektyvių parametrų vertinimo algoritmų samprata ir pristatoma jų kūrimo technika. Antrajame skyriuje apibrėžiamas kumuliacinis struktūrinių lygčių modelis ir pritaikoma pirmajame skyriuje pristatyta technika. Trečiajame skyriuje aprašomi veiksmas, kuriais iš antrajame skyriuje sudaryto algoritmo parametrus vertinti esant fiksuotam duomenų rinkiniui sukuriamas algoritmas parametrus vertinti realiuoju laiku. Ketvirtajame skyriuje akcentuojami praktiniai dinaminių sistemų modelių taikymo aspektai, analizuojant atotrūkį tarp modelių ir duomenų COVID-19 pandemijos plitimo modeliavimo atveju bei konstruojant socialinio kapitalo dinamikos modelį.

1. STOCHASTINIŲ DINAMINIŲ SISTEMŲ MODELIŲ PARAMETRŲ EFEKTYVIŲ VERTINIMO ALGORITMŲ SUDARYMO TECHNIKA

Šis skyrius parengtas remiantis [28, 97] straipsniais. Čia (1)–(3) lygtimis apibrėžtam bendrajam dinaminių sistemų modeliui suteikiamas konkretus dinaminių struktūrinių lygčių modelio pavidalas, kurio pagrindu toliau nagrinėjama efektyvaus parametrų vertinimo tematika.

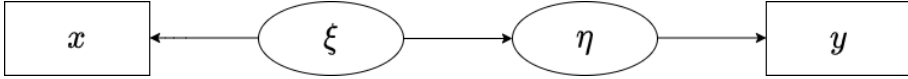
1.1. Struktūrinių lygčių modeliavimas

Struktūrinių lygčių modeliavimas (SLM) [112] sudaro veiksmingas galimybes modeliuoti daugelį realaus pasaulio reiškinių, kurie negali būti tiesiogiai stebimi ir todėl turi būti išreikšti išmatuojamais stebėjimais [3]. SLM dažniau taikomas srityse, kuriose intensyviai naudojamos abstrakčios sąvokos, pavyzdžiui, elgsenos ar socialiniuose moksluose [87], tačiau plačiai taikomas ir daugelyje kitų sričių, tokių kaip medicinos bei sveikatos mokslai [13], verslas ir rinkodara [73] ir gamtinės sistemos [42].

Savo esme SLM yra išsami metodika, skirta teoriniam (dažniausiai) tiesinių ryšių tarp kintamųjų tinklui atvaizduoti, įvertinti ir patikrinti [93]. SLM pripažinimą iš esmės lemia tai, kad pagal jį: 1) aiškiai įvertinamos matavimo paklaidos; 2) latentiniai (nestebimi) kintamieji įvertinami per stebimus kintamuosius (vienu metu sprendžiamos kelios susijusios lygtys, kad būtų nustatyti parametrų įverčiai); ir 3) atliekamas modelio testavimas, leidžiantis duomenis atitinkančios struktūros parinkimą [15]. Atsižvelgiant į tai, kad SLM atveju stebimi įvesties duomenys yra susiejami su stebimais išvesties duomenimis per nestebimus konstruktus, teoriškai SLM galima laikyti juodosios dėžės modeliavimu [37], tačiau dėl specifinės struktūros įvedimo praktiškai tai atitinka pilkosios dėžės modeliavimą [64].

Kaip statistinis metodas, SLM sujungia regresinę [101] ir faktorinę [18] analizes per matavimo modelį, kuris apibrėžia su vienu ar daugiau stebimų kintamųjų susijusius latentinius kintamuosius, ir struktūrinį modelį, kuris nurodo ryšius tarp latentinių kintamųjų (žr. 1.1 pav.). Įprastinė SLM procedūra apima hipotezių apie nagrinėjamą reiškinį formulavimą, kurios vėliau paverčiamos iš tiesinių lygčių su neapibrėžtais parametrais sudarytu modeliu. Norint naudoti modelį pri-

imant kiekybinius sprendimus, modelio parametrus reikia įvertinti pagal stebimus duomenis, kurie yra latentiniais kintamaisiais laikomų teorinių konstrukto vertimo išmatuojamais rodikliais rezultatas. Galiausiai įvertinamas modelio tinkamumas, kuriuo remiantis modelis yra priimamas arba atmetamas [62].



1.1 pav.: Supaprastinta abstrakti SLM diagrama, kurioje x (stebima įvestis) ir y (stebima išvestis) yra stebimi kintamieji, o ξ (latentinė įvestis) ir η (latentinė išvestis) yra latentiniai kintamieji. Rodyklės rodo poveikio kryptį. Tiek x kartu su ξ , tiek y kartu su η sudaro matavimo modelius, o η kartu su ξ sudaro struktūrinį modelį. Daugiau informacijos apie SLM diagramų braižymą galima rasti [62].

1.2. Dinaminis struktūrinių lygčių modelis

Bendrasis dinaminis (panelinis) struktūrinių lygčių modelis (DSLMM), apimantis daugumą įprastų tiesinių dinaminių (ir statinių) modelių, įskaitant ir klasikinį SLM (LISREL) [58], subjektui $i = 1, \dots, N$, laiku $t = 1, \dots, T$, yra formuluojamas taip [20]:

$$\eta_{i,t} = \mu_{\eta,i} + \sum_{j=0}^p B_j \eta_{i,t-j} + \sum_{j=0}^q \Gamma_j \xi_{i,t-j} + \zeta_{i,t}, \quad (1.1)$$

$$y_{i,t} = \mu_{y,i} + \Lambda_y \eta_{i,t} + \epsilon_{i,t}, \quad (1.2)$$

$$x_{i,t} = \mu_{x,i} + \Lambda_x \xi_{i,t} + \delta_{i,t}, \quad (1.3)$$

kur $\eta_{i,t} = (\eta_{i,t}^{(1)}, \dots, \eta_{i,t}^{(l)})'$ ir $\xi_{i,t} = (\xi_{i,t}^{(1)}, \dots, \xi_{i,t}^{(k)})'$ yra nestebimų (latentinių) kintamųjų vektoriai; $y_{i,t} = (y_{i,t}^{(1)}, \dots, y_{i,t}^{(n)})'$ ir $x_{i,t} = (x_{i,t}^{(1)}, \dots, x_{i,t}^{(m)})'$ – stebimų kintamųjų vektoriai; $\mu_{\eta,i}$ ($l \times 1$), $\mu_{y,i}$ ($n \times 1$) ir $\mu_{x,i}$ ($m \times 1$) – fiksuotų efektų vektoriai; B_j ($l \times l$), Γ_j ($l \times k$), Λ_y ($n \times l$) ir Λ_x ($m \times k$) – regresijos koeficientų matricos; $\zeta_{i,t}$ ($l \times 1$), $\epsilon_{i,t}$ ($n \times 1$) ir $\delta_{i,t}$ ($m \times 1$) – matavimo paklaidų (triukšmo) vektoriai.

Viena vertus, (1.1)–(1.3) lygtimis apibrėžto tiesinio modelio taikymas yra nesudėtingas ir nereikalauja ypatingos kompiuterio spartos ar atminties išteklių. Kita vertus, norint juo pasinaudoti, pirmiausia reikia

įvertinti jo parametrus – pageidautina gana tiksliai, nes nuo parametrų įverčių kokybės priklauso tolesnių modeliavimo uždavinių, tokių kaip filtravimas ir prognozavimas, kokybė. Tai dažnai yra sudėtingas uždavinys, ypač esant tokiam grįžtamojo ryšio mechanizmui kaip (1.1) lygties. Šios dalis pasirodo ir garsiajame Kalmano filtro modelyje, kurio praktinis įgyvendinimas sudėtingas dėl sunkumų gauti gerus triukšmo kovariacijų matricų įverčius [61].

Nagrinėjant DSLM parametrų vertinimą, svarbus skirtumas yra tarp atvejų, kai $N > 1$ ir kai $N = 1$. Pirmuoju atveju tai vadinama vertinimu su paneliniais duomenimis, o antruoju – vertinimu su grynaisiais laiko eilučių duomenimis [20]. Įprastai daugiau dėmesio skiriama paneliniams duomenims, tačiau susidomėjimas grynaisiais laiko eilučių duomenimis taip pat yra reikšmingas [20, 47, 104]. Skirtumas tarp šių dviejų atvejų yra tas, kad, kitaip nei panelinių duomenų atveju, grynujų laiko eilučių duomenų negalima pakeisti pakankama statistika, tai yra empiriniu kovariacijų matricos įverčiu, kuris yra kovariacinės struktūrinės analizės – plačiai paplitusios SLM srityje – pagrindas [20].

Nors DSLM pritaikomumas tikriausiai yra akivaizdesnis, kai $N > 1$, o ne kai $N = 1$ [45], šiame darbe išimtinai susitelkiama į atvejį $N = 1$ (dėl šios priežasties nuo šiol subjekto indeksas i bus praleidžiamas). Ši pasirinkimą motyvuoja tai, kad $N = 1$ sudaro fundamentaliai atskirą problemų klasę [20], kuri, nors ir nėra visa apimanti, vis dėlto turi reikšmingą taikymo sritį. $N = 1$ duomenys įprastai yra populiarūs tokiose srityse kaip ekonometrija, seismologija, meteorologija, valdymo inžinerija ir signalų apdorojimas; be to, jie taip pat perspektyvūs SLM artimesnėse srityse, tokiose kaip socialiniai ir medicinos mokslai [19, 48, 99]. Nepaisant ilgamečių kolektyvinių tyrimų, šiai sričiai vis dar gali būti siūlomi nauji sprendimai – kaip ir šio darbo atveju. Be to, atsižvelgiant į tai, kad šis darbas apskritai yra labiau orientuotas į metodologinę ekspoziciją, o ne į visapusiškai praktiškų rezultatų sukūrimą, atvejis $N = 1$ yra patrauklus dėl santykinai paprastesnio techninio pateikimo.

DSLM parametrų vertinimą $N = 1$ atveju galima atlikti įprastoje kovariacinės struktūrinės analizės aplinkoje, sudarant dirbtinę kovariacijų matricą, kurioje paslinktos laiko eilučių vektorių reikšmės laikomos papildomais kintamaisiais (vadinamasis Tioplico metodas) [46, 130]. Nors šis metodas nėra sudėtingas, jis pažeidžia nepriklausomų atvejų (eilučių)

prielaidą, todėl juo gaunami ne didžiausio tikėtumo, o momentų metodo įverčiai.

Kaip alternatyvą galima nagrinėti galimybę modelį užrašyti būsenų erdvės pavidalu ir naudoti Kalmano filtrą rekursiniam modelio tikėtumo funkcijos vertinimui, kai kiekviename žingsnyje gaunami nauji parametru įverčiai iki konvergavimo [104, 113, 130]. Nors vertinti naudojant Kalmano filtrą gaunami didžiausio tikėtumo įverčiai, tam reikalingos analitinės išvestinės ir geros pradinės reikšmės. Be to, rekursinis tokio parametru vertinimo pobūdis prisideda prie lėto konvergavimo problemų, ypač didelių modelių atveju, nes kiekviename vertinimo etape reikia naudoti Kalmano filtro rekursijas.

Klasikinis metodas didžiausio tikėtumo įverčiams gauti – tai tiesioginis didžiausio tikėtumo vertinimas, kai tikėtumo funkcija užrašoma visai duomenų imčiai [20, 25, 47, 118]. Tačiau kadangi tikėtumo funkcijos apskaičiavimas reikalauja ne mažesnės nei $T \times T$ matricos apvertimo, kuris yra sudėtingas skaičiuojamuoju požiūriu (tokios operacijos algoritminis sudėtingumas yra kubinis T atžvilgiu) [85], įprastų procedūrų naudojimas tikėtumo funkcijai optimizuoti yra praktiškas tik mažiems duomenų rinkiniams. Kita galima problema – aukšta modelio dimensija $n + m$, tačiau šiame darbe dėmesys skiriamas išimtinai duomenų apimties keliamiems iššūkiams.

Susidūrus su skaičiavimo problemomis, ypač taikant sudėtingesnius modelius, dažnai pasitelkiamas dar vienas metodas – Bajeso vertinimas su Gibbso (ar kita) atranka [75, 130]. Nors Bajeso metodais pagrįstą vertinimą palyginti lengva įgyvendinti sudėtingiems pasiskirstymams didelių matmenų erdvėse taikant Markovo grandinės Monte Karlo metodą, kuris leidžia dirbti su sudėtingomis modeliavimo situacijomis, šis metodas taip pat turi trūkumų – tarp jų sunkumų nustatant tikslumą ir konvergavimą (netgi empiriškai) [106].

1.3. Efektyvūs parametru vertinimo algoritmai

Atsižvelgiant į esamas DSLM parametru vertinimo galimybes, kai naudojami gryniesi laiko eilučių duomenys, galima apibendrinti, kad esami metodai arba duoda neoptimalius įverčius, arba yra neveiksmingi skaičiavimo požiūriu. Vieni metodai grindžiami įvairiais supaprastinimais, todėl paprastai yra lengvai apskaičiuojami, tačiau neoptimalūs – ta prasme, kad kiti įverčiai gali labiau susitelkti į tikrąją parametro vertę.

Kiti metodai paremti tvirta teorine baze, kaip, pavyzdžiui, didžiausio tikėtino vertinimas [80], todėl gali būti bent jau asimptotiškai optimalūs, bet reikalauja daug skaičiavimo išteklių.

Atsižvelgiant į pastarojo meto technologinę pažangą, inicijavusią didžiųjų duomenų erą [68], šiandien kaip niekada anksčiau paplitę dideli duomenų kiekiai. Norint išnaudoti jų potencialą, būtinas skaičiavimo efektyvumas, o parametrų įverčių optimalumas visada buvo pageidaujama savybė.

Todėl siekiant užtikrinti, kad DSLM išliktų konkurencinga priemonė, galinčia spręsti šiuolaikinius uždavinius, vienas iš tolesnio jo tobulinimo aspektų turėtų būti skaičiavimo požiūriu efektyvių algoritmų, pagrįstų optimaliais parametrų vertinimo metodais, kūrimas.

1.1 pastaba. *Pažymėtina, kad, atsižvelgiant į šio darbo tikslus, skaičiavimo efektyvumas suprantamas kaip tiesinio laiko algoritminis sudėtingumas stebėjimų imties dydžio T atžvilgiu.*

1.2 pastaba. *Įverčių optimalumas šiame darbe suprantamas kaip vertinimo taisyklių, pagal kurias gaunami parametrų įverčiai, norimų savybių (asimptotinis) patenkinimas, tai yra: 1) nepaslinktumas, 2) pagrįstumas ir 3) efektyvumas – savybė, susijusi su mažiausia įmanoma paklaida [36].*

1.3 pastaba. *Nors šiame darbe nagrinėjamas būtent DSLM, tiek įverčių optimalumu, tiek skaičiavimo efektyvumu pasižymintys algoritmai yra siektini ir kituose kontekstuose (žr., pvz., [116]). Apskritai neabejotinai vertingas algoritmas turėtų pasižymėti 1.1 ir 1.2 pastabose aptartomis savybėmis – toks algoritmas laikytinas efektyviu.*

Šiuo požiūriu (plačiąja prasme) didžiausio tikėtino vertinimas užtikrina įverčių optimalumo dalį (1.2 pastaba), bet nepajėgia užtikrinti skaičiavimo efektyvumo (1.1 pastaba). O įvairius supaprastinimus naudojančios metodai geba patenkinti skaičiavimo efektyvumo reikalavimus (1.1 pastaba), tačiau negali užtikrinti įverčių optimalumo (1.2 pastaba) dėl nepasiekiamo efektyvumo – trečiosios savybės, nurodytos 1.2 pastaboje.

Taigi reikalingi moksliniai tyrimai, kuriais būtų siekiama panaikinti kompromiso poreikį.

Nors efektyvaus algoritmo sukūrimas duotam modeliui dažnu atveju nėra lengva užduotis, galima formuluoti bendruosius tokio proceso principus. Dėl paprastumo bendrai vadinkime juos *palaipsninio vertinimo technika* (PVT).

PVT yra susijusi su skaičiavimo požiūriu efektyviu didžiausio tikėtinumo metodo (DTM) taikymu – priešingai nei su skaičiavimo požiūriu efektyvus, bet vertinimo požiūriu neoptimalaus metodo pavertimu optimaliu arba fundamentaliai naujo metodo kūrimu.

Kadangi parametrams vertinti naudojamas DTM, PVT objektas yra tikėtinumo funkcija, formuluojama galimai transformuotiems duomenims (pagal iniciatyvinę PVT dalį). Vėliau jos apdorojimas vykdomas trimis etapais (pagal operatyvinę PVT dalį), kurių sėkmingas įgyvendinimas leidžia sukurti skaičiavimo požiūriu efektyvius didžiausio tikėtinumo parametrų vertinimo algoritmus.

Iniciatyvinė PVT dalis. Pradinė DSLM tikėtinumo funkcija – vidurkių vektoriui ir kovariacijų matricai, kuriai būdinga blokinė struktūra [20], priklausančių modelio parametrų funkcija. Iniciatyvinės PVT dalies tikslas – transformuoti modelio duomenis taip, kad būtų išretinti kovariacijų matricos blokai, paverčiant juos daugiaįstrižainėmis matricomis. To siekiama todėl, kad retosios matricos yra lengviau apdorojamos nei tankiosios, o daugiaįstrižainės (arba plačiąja prasme juostinės) matricos sudaro svarbų ir santykinai gerai ištirtą retųjų matricų poaibį [27].

Operatyvinė PVT dalis. Operatyvinę PVT dalį sudaro trys nuoseklūs etapai, nusakantys reikiamus atlikti veiksmus:

1. Iniciatyvinėje PVT dalyje gautos kovariacijų matricos blokinė dekompozicija. Šiuo veiksmu pradinė nuo modelio parametrų priklausanti tikėtinumo funkcija pakeičiama ekvivalentine, reparametrizuota tikėtinumo funkcija. Ši nauja funkcija priklauso nuo unikalių vidurkių vektorių ir kovariacijų matricos elementų (galimai transformuotų, jei tai palengvina tolesnį apdorojimą), kurie toliau vadinami tiesiog parametrais. Taip gaunama tikėtinumo funkcija, kurios kovariacijų matricos pakaitalas – tai yra ne mažesnė nei $T \times T$ matrica, kurią reikia apversti ir kurios determinantą reikia apskaičiuoti, – priklauso nuo mažiausio skaičiaus parametrų.
2. Gautos ekvivalentinės tikėtinumo funkcijos diferencijavimas pagal parametrus, kurie nepriklauso įvestam kovariacijų matricos

pakaitalui, ir paskui atliekamas manipuliavimas gautais pradiniais įvertiniais, kuriuo siekiama gauti išreikštinius įvertinius, priklausančius tik nuo kovariacijų matricos pakaitalą sudarančių parametru. Šiuos išreikštinius įvertinius ištačius į ekvivalentinę tikėtinumo funkciją, gaunama tik nuo kovariacijų matricos pakaitalui priklausančių parametru priklausanti funkcija, taip sumažinant sprendimo optimizavimo uždavinio dimensiškumą.

3. Veiksmingų rekursinių formuliu, kurių kiekvienas žingsnis reikalauja pastovaus skaičiaus elementariųjų operacijų, išvedimas. Šios formulės yra būtinos sudėtingiems nariams, susijusiems su kovariacijų matricos pakaitalu ir sudarantiems gautus įvertinius, apskaičiuoti per laiką, tiesiogiai proporcingą stebėjimų imties dydžiui T .

1.4 pastaba. Įvertinus visus ekvivalentinės tikėtinumo funkcijos, įvestos pirmajame operatyvinės PVT dalies etape, parametrus, iš esmės įvertinami unikalūs vidurkių vektoriaus ir kovariacijų matricos elementai. Tai leidžia grįžti prie originalių modelio parametru, pritaikant klasikinius ir gerai žinomus šios srities rezultatus, jei modelis yra identifikuojamas.

Pagal savo specifiką PVT geriausiai gali būti plėtojama ir taikoma remiantis požiūriu „iš apačios į viršų“, tai yra pradedant nuo specifškumo ir judant link bendrumo [111]. Iš esmės taip yra todėl, kad kuo sudėtingesnis modelis, tuo sudėtingesnė jo kovariacijų struktūra, todėl PVT etapai tampa sudėtingesni tiek reikalingo matematinio aparato, tiek išvalgų gilumo požiūriu.

Vis dėlto sėkmingas PVT pritaikymas konkrečiam modeliui leidžia ekstrapoliuoti įgytą patirtį pasirinktinai mažam žingsniui link galutinio bendrojo modelio (pvz., aprašyto (1.1)–(1.3) lygtimis). Šis procesas gali būti kartojamas, taip realizuojant veikimą „iš apačios į viršų“.

1.5 pastaba. PVT galima laikyti duokle DTM kaip įvairių euristinių alternatyvų (pvz., simuliaciniu pagrįstų vertinimo metodų) priešingybei. DTM paprastai yra labiau pageidaujamas už kitus metodus, nes juo gaunami parametru įverčiai pasižymi optimalumu, o tai reiškia, kad asimptotiniu požiūriu jo negali pranokti joks kitas metodas. Tačiau autoriai, bandantys taikyti DTM sudėtingų modelių parametrus vertinti, neretai susiduria su skaičiavimo problemomis, tad įverčių kokybę iškeičia į skaičiavimo supaprastinimus [75] (šiua laikinė su DSLM susijusi programinė įranga pasižymi

įdiegtais bendrais modeliais ir stochastiniais metodais, tokiais kaip jiems spręsti skirti Markovo grandinės Monte Karlo metodai [6]). PVT siūlymas iš esmės yra siūlymas permąstyti DTM, nes PVT taikymas gali palengvinti DTM įgyvendinančių algoritmų sukūrimą, taip ilgiau išliekant prie DTM naudojimo. Ir nors vargu ar galima išvengti visų sprendžiamumo problemų, galima teigti, kad kuo didesnis modelių, kurių parametrus galima įvertinti su DTM, rinkinys, tuo didesnė bendra teorinė ir praktinė nauda, o PVT gali prisidėti prie šio rinkinio išplėtimo.

1.4. Parametrų vertinimas realiuoju laiku

Įprastai (D)SLM tyrimai yra susiję su parametrų vertinimu esant fiksuotam duomenų rinkiniui [20, 44]. Tai reiškia, kad pirmiausia surenkami visi aktualūs duomenys, o tada įvertinami parametrai; tai paprastai atliekama iš karto viso duomenų rinkinio atžvilgiu. Pagal apibrėžimą toks parametrų vertinimo būdas duomenų rinkimą ir parametrų vertinimą išskaido į du atskirus tyrimo etapus. Mažiau akivaizdu, tačiau tai taip pat gali reikšti, kad ryšiai tarp kintamųjų yra stabilūs laikui bėgant, o tai dažnai yra pagrįsta prielaida.

Tačiau gali būti ir taip, kad ryšiai tarp kintamųjų laikui bėgant kinta arba duomenų srautas yra nenutrūkstamas (galimai be jokios apibrėžtos pabaigos), o sprendimai, kuriems įtaką daro modelio parametrų įverčiai, turi būti priimami skirtingais laiko momentais, remiantis iki jų surinktais duomenimis [120]. Tokiais atvejais reikalingas parametrų vertinimas realiuoju laiku, apimantis parametrų įverčių atnaujinimą iki naujo duomenų pasirodymo [35]. Parametrų vertinimas realiuoju laiku leidžia išnaudoti laipsniškai didėjančių duomenų kiekį parametrų įverčiams tikslinti bei prisitaikyti prie sistemos dinamikos pokyčių, aktualių nestacionariose aplinkose, kuriose fundamentalus duomenų generavimo procesas laikui bėgant gali keistis.

1.6 pastaba. *PVT atliepia bendruosius efektyvių algoritmų, skirtų parametrams vertinti esant fiksuotam duomenų rinkiniui, sudarymo principus. Savo ruožtu parametrų vertinimas realiuoju laiku reikalauja papildomų veiksmų, susijusių su antrajame operatyvinės PVT dalies etape gauto sumažinto dimensiskumo optimizavimo uždavinio specifiniu sprendimu.*

1.7 pastaba. *Techniškai parametrų vertinimas esant fiksuotam duomenų rinkiniui gali būti naudojamas parametrų vertinimo realiuoju laiku tikslais. Tačiau*

su kiekvienu paskesniu parametru atnaujinimu tai didina skaičiavimo našta, nes vertinimas atliekamas panaudojant visą duomenų aibę, kuri laikui bėgant didėja. O praktiškai priimtinam parametru vertinimui realiuoju laiku ši skaičiavimo našta turi išlikti stabili.

Parametru vertinimo realiuoju laiku atveju modelio parametrai turi būti nedelsiant atnaujinami pagal naujus turimus duomenis, taip atspindint naują informaciją. Kad neatsiliktų nuo gaunamo duomenų srauto, būtina, jog naudojami parametru vertinimo algoritmai pakankamai sparčiai apdorotų naujus duomenis ir atnaujintų parametru įverčius. Įprastai tradicinius, kokybiškus parametru vertinimo metodus įgyvendinantys algoritmai nėra tinkami veikti realiuoju laiku, nes dažnai apima skaičiavimo požiūriu intensyvias iteracines procedūras. Pavyzdžiui, tokie metodai kaip didžiausio tikėtinumo ar mažiausių kvadratų vertinimas, kurie įprastai naudojami parametrams vertinti esant fiksuotai duomenų imčiai, apima skaičiavimo požiūriu reiklų optimizavimo problemų sprendimą, kurios, siekiant patenkinti parametru vertinimo realiuoju laiku keliamus reikalavimus, tampa dar sudėtingesnės. Taigi reikalingi algoritmai, kurie būtų specialiai sukurti parametrams vertinti realiuoju laiku. Šie algoritmai turi pasižymėti greitaveika, teikti patikimus parametru įverčius naudojant mažiau duomenų ir būti atsparūs iš to galimai kylančiam nestabilumui.

Šiuos algoritmus, skirtus parametrams vertinti realiuoju laiku, sudarymo iššūkius galima bandyti įveikti naudojant stochastiniu aproksimavimu pagrįstus metodus, pavyzdžiui, stochastinio gradientinio nusileidimo algoritmą – iteracinį metodą, skirtą optimizuoti tikslo funkciją su tinkamomis glotnumo savybėmis [114]. Jis gali būti laikomas gradientinio nusileidimo optimizavimo metodo (plačiai naudojamo mašininiame mokyme ir statistikoje, siekiant rasti funkciją minimizuojančius parametrus [78]) stochastiniu aproksimavimu, nes pakeičia tikrąjį gradientą (apskaičiuojamą iš visos duomenų aibės) tam tikru jo įverčiu, apskaičiuojamu iš atsitiktinai parinkto duomenų aibės poaibio.

O alternatyvus būdas – naudoti rekursiniu vertinimu pagrįstus metodus, pavyzdžiui, rekursinį matematinės vilties maksimizavimo algoritmą, specialiai sukurtą realaus laiko užduotims, susijusioms su paslėpto Markovo modelio parametru įvertinimu [116], spręsti. Pastarieji metodai gali atnaujinti parametru įverčius pagal jų einamąsias reikšmes ir naujus duomenis, taip išvengiant visos duomenų istorijos pakartotinio

apdorojimo.

1.8 pastaba. Iš esmės parametrų vertinimo realiuoju laiku algoritmai gali būti sudaromi naudojant stochastinį arba rekursinį (palaipsninį) metodą. Stochastinis metodas yra pagrįstas fiksuoto dydžio atsitiktiniais viso duomenų rinkinio, prieinamo duotuoju laiko momentu, poaibiais, kurie naudojami atliekant parametrų vertinimą sudarančius procedūrinius veiksmus. O štai rekursinis (palaipsninis) metodas yra susijęs su parametrų įverčių atnaujinimu, naudojant tik naujai gautus duomenis, tai yra nauji parametrų įverčiai priklauso nuo esamų parametrų įverčių ir duomenų, gautų laikotarpiu tarp dviejų parametrų vertinimo momentų.

Nors parametrų vertinimo realiuoju laiku koncepcija egzistuoja jau ne vieną dešimtmetį, per kuri įvairiose srityse pasiūlyta nemažai ją įgyvendinančių algoritmų, DSLM pažanga šiuo aspektu yra santykinai ribota. Tai iš dalies gali būti siejama su vystymo sunkumais, kylančiais bandant kurti tokius parametrų vertinimo realiuoju laiku algoritmus, kurie greitaveikos vardan neaukotų parametrų įverčių kokybės. Kita vertus, galimai vis dar stokojama geresnio parametrų vertinimo realiuoju laiku potencialo ir svarbos supratimo didžiųjų duomenų eros kontekste, todėl šiai sričiai plėtoti nėra skiriama daugiausia įmanomų pastangų. Nepaisant šių aplinkybių, DSLM visavertiškumui šiuolaikiniame kontekste būtini efektyvūs parametrų vertinimo realiuoju laiku algoritmai, kurių sudarymas, pagal PVT, turėtų vykti DTM ir rekursijos pagrindais.

1.5. Skyriaus išvados

- DSLM yra svarbus tiesinis stochastinių dinaminių sistemų modelis, leidžiantis kontekstualiai nagrinėti efektyvaus parametrų vertinimo tematiką.
- Esami DSLM parametrų vertinimo algoritmai pasižymi arba įverčių kokybe, arba veikimo greitaveika, tačiau norint visapusiškai išnaudoti šiuolaikines aplinkybes reikalingi abiem savybėmis pasižymintys algoritmai.
- PVT apibrėžia veiksmus, kurių sėkmingas atlikimas leidžia sukurti efektyvius algoritmus DSLM ir bendresnių stochastinių

dinaminių sistemų modelių parametrų vertinti, kai duomenų rinkinys yra fiksuotas.

- DSLM parametrų vertinimas realiuoju laiku stokoja pelnyto dėmesio didžiųjų duomenų eros kontekste, tačiau efektyvių tam skirtų algoritmų vystymas gali būti vykdomas kaip PVT sąsaja.

2. PALAIPSNINIO VERTINIMO TECHNIKOS TAIKYMAS KUMULIACINIO STRUKTŪRINIŲ LYGČIŲ MODELIO ATVEJU

Šiame skyriuje, parengtame [97] straipsnio pagrindu, palaipsninio vertinimo technika (PVT) pademonstruojama arba, turint omenyje jos taikymą pagal požiūrį „iš apačios į viršų“, inicijuojama santykinai paprastu, tačiau fundamentaliu (1.1)–(1.3) lygtimis apibūžiamo modelio atveju, nustatant $B_j = 0_{l \times l}$ ($j \neq 1$), $B_1 = I_{l \times l}$ ir $\Gamma_j = 0_{l \times k}$ ($j > 0$). Gaunamas modelis, vadinamas *kumuliaciniu struktūrinių lygčių modeliu* (KSLM). KSLM – atskiras dinaminio struktūrinių lygčių modelio (DSLMM) atvejis, gaunamas apribojant tik struktūrinę lygtį, tačiau kitus atskirus KSLM (taigi ir DSLMM) atvejus galima gauti apribojant matavimo lygtis. Galima teigti, kad praktinių taikymų atžvilgiu KSLM pagrįstas Okamo skustuvo principu [107], nes derinamas santūrumas, tikslumas ir adekvatumas. Šiame modelyje yra latentinių kintamųjų, matuojamų faktoriniu išvesties modeliu, o jų prieaugiai regresuojami pagal kitus modeliuojamus latentinius kintamuosius, matuojamus faktoriniu įvesties modeliu [18]. Atsižvelgiant į tai, kad $B_1 = I_{l \times l}$, KSLM modeliuojamas procesas yra panašus į daugiamatį, su triukšmu stebimą Vynerio procesą, kuris, kaip žinoma, nėra stacionarus ir yra fundamentalus tiek teorinėje, tiek taikomojoje matematikoje [52, 128].

KSLM, pavyzdžiui, galėtų būti taikomas kultūrinio dalyvavimo poveikiui socialiniam kapitalui modeliuoti [95]. Šiuo atveju kultūrinis dalyvavimas būtų latentinis kintamasis ξ , kuris pasireikštų per asmens polinkį lankytis įvairaus pobūdžio kultūros renginiuose (kine, teatre, koncertuose ir kt.). Šį polinkį būtų galima įvertinti naudojant su atitinkamais renginiais susijusią statistiką, tokią kaip bilietų pardavimas arba faktinis lankomumas (stebimas kintamasis x). Socialinis kapitalas būtų dar vienas latentinis kintamasis (η), kuris priklausytų nuo ankstesnės savo vertės ir dabartinės kultūrinio dalyvavimo vertės ξ (tai reikštų, kad socialinio kapitalo prieaugį lemia bendras kultūrinio dalyvavimo lygis tarp dviejų gretimų matavimo momentų) ir kuris galėtų būti vertinamas pagal atsakymus į specialiai jam matuoti sudarytą klausimyną (stebimas kintamasis y).

Iš esmės KSLM taikymo diapazonas yra susijęs su tokiais atvejais, kai endogeninių latentinių kintamųjų verčių vienkartinį pokytį veikia atitinkamos dabartinės egzogeninių latentinių kintamųjų reikšmės (kaip nagrinėtame atvejuje arba modeliuojant, tarkime, sveikatos pokyčius

priklausomai nuo gyvenimo būdo pasirinkimų, nuotaikų kaitą priklausomai nuo reakcijos į išorinius poveikius ir pan.).

2.1 pastaba. *Iš principo KSLM yra naudingas vertinant tam tikrų veiksmų poveikį tam tikrų atributų vertės pokyčiams. Tačiau, palyginti su bendruoju (1.1)–(1.3) lygtimis apibrėžtu DSLM, jis pasižymi ribota atmintimi ir neapima nei galimų tiesioginių tarpusavio priklausomybių, nei multiplikatyvinių atributų efektų.*

Šiame skyriuje išimtinai nagrinėjama skaičiavimo požiūriu efektyvaus didžiausio tikėtino vienmatės išvesties KSLM (t. y. $n = l = 1$), su $\mu_y = 0$ ir $\Lambda_y = 1$, parametrų vertinimo iš grynųjų laiko eilučių duomenų problema, kai duomenų rinkinys yra fiksuotas, laikant, kad modelio kintamieji ir matavimo paklaidos turi normalųjį pasiskirstymą.

Viena vertus, efektyvūs skaičiavimai yra būtini, siekiant susitvarkyti su didžiųjų duomenų erai būdingais dideliais duomenų kiekiais. Kita vertus, didžiausio tikėtino metodo (DTM) dėl savo (asimptotiškai) optimalių įvertinių savybių yra prioritetinis metodas, kuriam verta ieškoti skaičiavimo atžvilgiu efektyvių algoritmų.

Nagrinėjamas modelis (arba bet kuris KSLM) yra tik atskiras bendrojo DSLM, nagrinėjamo, pavyzdžiui, [20] arba [6], atvejis. Tačiau sukurti efektyvius parametrų vertinimo algoritmus bendrajam DSLM yra sudėtinga, nes jo DTM apima tankios kovariacijų matricos apvertimą.

O čia nagrinėjamo KSLM modelio atveju kovariacijų matricos analizė gali būti supaprastinta iki blokinės kovariacijų matricos, kurios blokai yra daugiausia tridiagonalinės matricos, žinomos dėl savo veiksmingų savybių. Pavyzdžiui, simetrinės tridiagonalinės matricos atvirkštinė matrica turi analitiškai išreikšiamus elementus [51], o tiesinei tridiagonalinei lygčių sistemai išspręsti reikalingas operacijų skaičius auga tiesiškai su lygčių skaičiumi, o ne kubiškai, kaip kad Gauso eliminacijos atveju [38].

Tokiu būdu darbe nagrinėjamas atskiras atvejis leidžia spręsti efektyvaus DSLM įvertinimo problemą pagal principą „iš apačios į viršų“ [111].

2.2 pastaba. *Atrodo natūralu iš karto nagrinėti bendrąjį DSLM, nes tai susiję su maksimalia praktine nauda. Tačiau taip darant kyla pavojus nepaisyti sudėtingų, bet optimalių metodų (pvz., DTM) ir rinktis nesudėtingas, tačiau*

neoptimalias alternatyvas (pvz., Markovo grandinės Monte Karlo metodus [81]), nes bendrasis DSLM yra sudėtingas ir todėl sunkiai sprendžiamas [75].

Ir atvirkščiai – atskiri DSLM atvejai labiau pasiduoda efektyviems sprendimams, o dėl to jų tyrimas yra vertas dėmesio, nepaisant mažesnio praktinio pritaikomumo.

Likusi skyriaus dalis sudaryta taip, kaip nurodyta toliau. Pirmajame poskyryje konkrečiai suformuluojamas nagrinėti pasirinktas modelis ir su juo susijusi parametrų vertinimo problema, kurią siekiama spręsti per PVT prizmę. Antrajame poskyryje ši problema sprendžiama taikant PVT principus, tokiu būdu klasikinių kubinio sudėtingumo algoritmą, įgyvendinantį DTM, paverčiant tiesinio sudėtingumo algoritmu. Trečiajame poskyryje atliekami sudaryto algoritmo empiriniai eksperimentai, siekiant patikrinti jo konvergavimą, įverčių kokybę bei vykdymo trukmę. Galiausiai pateikiamos skyriaus išvados.

2.1. Problemos formuluotė

Nagrinėjamas modelis apibūrinamas trimis susijusiomis lygtimis.

Pirmoji lygtis gali būti vadinama perėjimo (struktūrine) lygtimi:

$$\eta_t = \eta_{t-1} + \mu_\eta + \Gamma_0 \xi_t + \zeta_t, \quad \eta_0 = c, \quad (2.1)$$

kur $\{\eta_0\} \cup \{\eta_t\}$ yra latentinių būsenų skaliarų seka (c – šios sekos skaliarinė pradžia), μ_η – fiksuotų efektų skaliaras, Γ_0 – latentinės įvesties į latentinį procesą svorių $1 \times k$ vektorius, $\{\xi_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių latentinės įvesties į latentinį procesą $k \times 1$ vektorių seka (bendrieji faktoriai), kur $\xi_t \sim \mathcal{N}(0_{k \times 1}, I_{k \times k})$, o $\{\zeta_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių latentinio proceso triukšmo skaliarų $\mathcal{N}(0, \sigma_\eta^2)$ seka, $t = 1, \dots, T$.

Antroji lygtis gali būti vadinama išvesties matavimo lygtimi:

$$y_t = \eta_t + \epsilon_t, \quad (2.2)$$

kur $\{y_t\}$ yra stebimų išvesties skaliarų seka, o $\{\epsilon_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių stebimo išvesties proceso triukšmo skaliarų $\mathcal{N}(0, \sigma_y^2)$ seka, $t = 0, \dots, T$.

Trečioji lygtis gali būti vadinama įvesties matavimo lygtimi:

$$x_t = \mu_x + \Lambda_x \xi_t + \delta_t, \quad (2.3)$$

kur $\{x_t\}$ yra stebimų įvesties $m \times 1$ vektorių seka, μ_x – fiksuotų efektų $m \times 1$ vektorius, Λ_x – bendrųjų faktorių svorių $m \times k$ matrica, o $\{\delta_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių specifinių faktorių $m \times 1$ vektorių seka, kur $\delta_t \sim \mathcal{N}(0_{m \times 1}, \text{diag}(\sigma_{x_1}^2, \dots, \sigma_{x_m}^2))$, $t = 1, \dots, T$.

Daroma prielaida, kad $\{\xi_t\}$, $\{\zeta_t\}$, $\{\delta_t\}$ ir $\{\epsilon_0\} \cup \{\epsilon_t\}$, $t = 1, \dots, T$, yra tarpusavyje nepriklausomos.

2.3 pastaba. (2.1)–(2.3) lygtimis apibrėžtas modelis yra atskiras (1.1)–(1.3) lygtimis apibrėžto modelio atvejis su išvesties matavimo lygtimi, taip pat apibrėžta laiko momentui $t = 0$.

2.4 pastaba. Sakoma, kad (2.1)–(2.3) lygtys apibūdina vienmatės išvesties grynųjų laiko eilučių duomenų KSLM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis.

Užuot naudojus tiesiogiai stebimą išvestį y_t , $t = 0, \dots, T$, bus naudojami jos gretimų laiko momentų skirtumai, nes tokiu būdu į kovariacijų matricą bus įvestas praretinimas, suteikiant jai tridiagonalinės matricos struktūrą:

$$\begin{aligned} \Delta y_t &= y_t - y_{t-1} \\ &= \mu_\eta + \Gamma_0 \xi_t + \zeta_t + (\epsilon_t - \epsilon_{t-1}), \end{aligned} \quad (2.4)$$

$t = 1, \dots, T$.

2.5 pastaba. (2.4) lygtis yra laikytina iniciatyvinės PVT dalies įgyvendinimu.

Prie μ_x pabaigos prijungus μ_η , o prie Λ_x pabaigos prijungus Γ_0 , gautas vektorius ir matrica atitinkamai pažymimi kaip μ ir L (taip pat dėl vėlesnio poreikio papildomai pažymima $\sigma = (\sigma_{x_1}, \dots, \sigma_{x_m}, \sigma_\eta, \sigma_y)'$).

Taip pat prie x_t pabaigos prijungus Δy_t , $t = 1, \dots, T$, gautas vektorius pažymimas kaip z_t , papildomai pažymint $z = (z_1, \dots, z_T)'$. Tada

$$Ez_t = \mu,$$

$t = 1, \dots, T$, ir

$$C_{zz}(\tau) = \begin{cases} \Sigma, & \tau = 0 \\ -\Psi_1, & \tau = 1 \\ 0_{(m+1) \times (m+1)}, & \text{kitu atveju} \end{cases},$$

kur $\Sigma = LL' + \Psi_0$, o

$$\Psi_0 = \begin{bmatrix} \sigma_{x_1}^2 & 0 & \dots & 0 & 0 \\ 0 & \sigma_{x_2}^2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & \sigma_{x_m}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\eta^2 + 2\sigma_y^2 \end{bmatrix}, \quad \Psi_1 = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_y^2 \end{bmatrix}.$$

Nagrinėjamas efektyvus didžiausio tikėtimumo μ , Σ ir σ_y^2 (t. y. (2.3)–(2.4) lygtimis modeliuojamų duomenų teorinių vidurkių vektoriaus ir nenulinių kovariacijų matricos elementų) vertinimas esant fiksuotam duomenų rinkiniui.

Siekiama sukurti DTM pagrįstą algoritmą, kuris teiktų šių dydžių didžiausio tikėtimumo įverčius ir būtų tiesinis stebėjimų imties dydžio T atžvilgiu.

Gavus įverčius, galima gauti matricos

$$\tilde{\Sigma} = LL' + \text{diag}(\sigma_{x_1}^2, \dots, \sigma_{x_m}^2, \sigma_\eta^2),$$

kuri struktūriškai atitinka faktorinės analizės modelio kovariacijų matricą [129], įvertį ir panaudoti esamą faktorinės analizės teoriją L bei $\sigma_{x_1}^2, \dots, \sigma_{x_m}^2, \sigma_\eta^2$ įverčiams atskirti, taip gaunant pradinio (2.1)–(2.3) lygtimis apibrėžiamo modelio parametrų įverčius. Tačiau ši operacija nepatenka į šio darbo apimtį, nes juo nesiekiama pateikti naujų šios srities rezultatų (dėl bendrojo požiūrio žr. tolesnę pastraipą arba, pvz., [59]).

$\hat{L}$ ir $\hat{\sigma}_{x_1}^2, \dots, \hat{\sigma}_{x_m}^2, \hat{\sigma}_\eta^2$ atstatymas iš matricos $\hat{\tilde{\Sigma}}$, be kitų būdų, gali būti atliekamas taikant principinių komponentų metodą. Pagal šį metodą matricą $\hat{\tilde{\Sigma}}$ reikia išreikšti jos tikrinių reikšmių ir tikrinių vektorių poromis $(\hat{\lambda}_1, \hat{v}_1), (\hat{\lambda}_2, \hat{v}_2), \dots, (\hat{\lambda}_{m+1}, \hat{v}_{m+1})$, kur $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_{m+1}$.

Tada $\hat{L}$ apskaičiuojamas taip:

$$[\hat{L}]_{i,j} = \sqrt{\hat{\lambda}_j} [\hat{v}_j]_i,$$

kur $i = 1, \dots, m+1, j = 1, \dots, k$.

O $\hat{\sigma}_{x_1}^2, \dots, \hat{\sigma}_{x_m}^2$ ir $\hat{\sigma}_\eta^2$ atitinka įstrižinės matricos $\hat{\tilde{\Sigma}} - \hat{L}\hat{L}'$ elementus, todėl:

$$\hat{\sigma}_{x_i}^2 = \left[\hat{\tilde{\Sigma}} \right]_{i,i} - \sum_{j=1}^k \left(\left[\hat{\tilde{L}} \right]_{i,j} \right)^2,$$

$$\hat{\sigma}_{\eta}^2 = \left[\hat{\tilde{\Sigma}} \right]_{m+1,m+1} - \sum_{j=1}^k \left(\left[\hat{\tilde{L}} \right]_{m+1,j} \right)^2,$$

kur $i = 1, \dots, m$.

2.6 pastaba. *Pabrėžtina, kad nors ir siekiami didžiausio tikėtino įverčiai, šiame skyriuje užduotis apribojama tikėtino funkcijos stebimiems duomenims užrašymu ir paskesniu atitinkamų įvertinių bei veiksmingų jų skaičiavimo formulių išvedimu.*

Kitaip tariant, galima užtikrintai teigti, kad sudarytas algoritmas paremtas DTM, tačiau teiginiai apie faktinius didžiausio tikėtino įverčius turėtų turėti teoriškai griežtą pagrindimą.

2.7 pastaba. *Pasirinkimas iš pradžių įvertinti Σ , užuot išliekant ties originaliais modelio parametrais, palengvina operatyvinės PVT dalies įgyvendinimą.*

2.8 pastaba. *Nagrinėjant L ir $\sigma_{x_1}^2, \dots, \sigma_{x_m}^2, \sigma_{\eta}^2$ gavimą iš $\tilde{\Sigma}$, reikia pažymėti, kad iš esmės tai reiškia $(m+1)k + (m+1)$ parametrų reikšmių gavimą iš $\frac{(m+1)(m+2)}{2}$ unikalių $\tilde{\Sigma}$ elementų (lygčių).*

Jei šie dydžiai sutampa, (paprastai) egzistuoja unikalūs sprendinys, duodantis L ir $\sigma_{x_1}^2, \dots, \sigma_{x_m}^2, \sigma_{\eta}^2$ – tokiu atveju sakoma, kad modelis yra tiesiog identifikuojamas.

Jei parametrų yra daugiau nei lygčių, egzistuoja įvairių parametrų nustatymų, kurie visi puikiai atitinka kovariacijų matricą, kontinuumas – tokiu atveju teigiama, kad modelis yra nepakankamai identifikuojamas.

Jei lygčių yra daugiau nei parametrų, sakoma, kad modelis yra pernelyg identifikuojamas, o tai reiškia, kad L ir $\sigma_{x_1}^2, \dots, \sigma_{x_m}^2, \sigma_{\eta}^2$ galima gauti tik tuo atveju, jei $\tilde{\Sigma}$ yra tinkamos, apribotos formos – tai yra, jei modelis yra suderintas.

2.2. Problemos sprendimas

Šiame poskyryje naudojama PVT, siekiant sudaryti tiesinio sudėtingumo DTM algoritmą, skirtą praeitame poskyryje aprašytam modeliui įvertinti pagal fiksuotą duomenų rinkinį.

Skirtinguose skirsniuose demonstruojami skirtingi algoritmo sudarymo žingsniai, o paskutiniame skirsnyje rezultatai apibendrinami sudaryto algoritmo pseudokodu.

Išdėstymas papildytas pastabomis, akcentuojančiomis skirtingus operatyvinės PVT dalies etapus.

2.2.1. Tikėtinumo funkcija

Pažymima:

$$I = I_{T \times T}, \quad J = J_{T \times T} = \begin{cases} -1, & |i - j| = 1 \\ 0, & \text{kitu atveju} \end{cases},$$

kur $i = 1, \dots, T, j = 1, \dots, T$.

Visą informaciją apimanti kovariacijų matrica yra sudaryta iš $(m + 1) \times (m + 1)$ blokų, kurių kiekviename yra $T \times T$ elementų:

$$C_{zz} = \Sigma \otimes I + \Psi_1 \otimes J.$$

Vektorizavus stebėjimo duomenų ir jų vidurkių matricas, pažymima: $z_{\text{vec}} = \text{vec}(z)$ ir $\mu_{\text{vec}} = \text{vec}(\underbrace{(\mu, \dots, \mu)}_T')$. Nesunku pastebėti, kad vektorius z_{vec} turi $(m + 1)T$ -matį Gauso skirstinį, todėl jo tikimybinis tankis yra:

$$p_{z_{\text{vec}}}(z_{\text{vec}}) = \frac{e^{-\frac{1}{2}d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}}}}{(2\pi)^{\frac{(m+1)T}{2}}|C_{zz}|^{1/2}}, \quad (2.5)$$

kur $d_{\text{vec}} = z_{\text{vec}} - \mu_{\text{vec}}$ (atkreiptinas dėmesys, kad čia nedaromas žymėjimo skirtumas tarp atsitiktinio vektoriaus ir tikimybinio tankio funkcijos argumento).

(2.5) lygtį logaritmuojant, gaunama logaritminė tikimybinio tankio funkcija:

$$\begin{aligned} \mathcal{L}(\theta) &= \mathcal{L}(\theta; z_{\text{vec}}) \\ &= -\frac{1}{2} (\ln(|C_{zz}|) + d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}} + (m + 1)T \ln(2\pi)), \end{aligned}$$

kur θ bendrai žymi modelio parametrus (visi jie priklauso μ_{vec} ir C_{zz}).

Kadangi galioja lygybė

$$\begin{aligned}\arg \max_{\theta} [\mathcal{L}(\theta)] &= \arg \max_{\theta} \left[\frac{\mathcal{L}(\theta)}{T} \right] \\ &= \arg \min_{\theta} \left[\frac{\ln(|C_{zz}|) + d'_{\text{vec}} C_{zz}^{-1} d_{\text{vec}}}{T} \right],\end{aligned}$$

toliau normalizavimo ir supaprastinimo tikslais yra nagrinėjamas tikėtinumo funkcijos $\mathcal{L}(\theta)$ minimizavimas, kur $\mathcal{L}(\theta)$ turi šią išraišką:

$$\mathcal{L}(\theta) = \frac{\ln(|C_{zz}|) + d'_{\text{vec}} C_{zz}^{-1} d_{\text{vec}}}{T}. \quad (2.6)$$

2.9 pastaba. Eiti tiesiogiai prie (2.6) optimizavimo (o tai yra įprasta elgsena [20]) yra neveiksminga skaičiavimo požiūriu dėl būtinybės apversti matricą C_{zz} , nes tai yra kubinio sudėtingumo operacija stebėjimų skaičiaus T atžvilgiu [85]. Tačiau galima bandyti išnaudoti specialią matricos C_{zz} struktūrą, siekiant galiausiai paspartinti skaičiavimus. Būtent tai sudaro pirmąjį operatyvinės PVT dalies etapą, pagal kurį matrica C_{zz} išskaidoma į blokus, susijusius: 1) tik su x , 2) ir su x , ir su Δy , ir 3) tik su Δy . Taip gaunama ekvivalenti, tam tikru būdu parametrizuota tikėtinumo funkcija, tinkama naudoti antrajame operatyvinės PVT dalies etape.

Dabar tikėtinumo funkcija $\mathcal{L}(\theta)$ išreiškiama taip, kad atsiskirtų skirtingi kovariacijų matricos C_{zz} blokai. Pažymima: $A = \Sigma \otimes I$, $B = \text{diag}(\lambda[J])$, $C = v[J]$ ir $D = \underbrace{(0_{T \times T}, \dots, 0_{T \times T}, C)'}_m$. Tada:

$$\begin{aligned}|C_{zz}| &= |A + D (\sigma_y^2 B) D'| \\ &= |A| |I + \sigma_y^2 B D' A^{-1} D| \\ &= |\Sigma|^T |C' C + \sigma_y^2 (C' J C) \Sigma_l I| \\ &= |\Sigma|^T |C' \sigma_y^2 \Sigma_l Q C| \\ &= |\Sigma|^T (\sigma_y^2 \Sigma_l)^T |C' C| |Q| \\ &= (|\Sigma| \Sigma_l)^T (\sigma_y^2)^T |I| |Q| \\ &= |\Sigma_{xx}|^T (\sigma_y^2)^T |Q| \\ &= (|\Sigma_{xx}| \sigma_y^2)^T |Q|\end{aligned} \quad (2.7)$$

ir

$$d'_{\text{vec}} C_{zz}^{-1} d_{\text{vec}} = d'_{\text{vec}} (A + D (\sigma_y^2 B) D')^{-1} d_{\text{vec}}$$

$$\begin{aligned}
&= d'_{\text{vec}} \left(A^{-1} - A^{-1} D \left((\sigma_y^2 B)^{-1} + D' A^{-1} D \right)^{-1} D' A^{-1} \right) d_{\text{vec}} \\
&= d'_{\text{vec}} \left(A^{-1} - A^{-1} D C \left((\sigma_y^2 J)^{-1} + \Sigma_l I \right)^{-1} C' D' A^{-1} \right) d_{\text{vec}} \\
&= d'_{\text{vec}} \left(A^{-1} - \frac{A^{-1} D C J Q^{-1} C' D' A^{-1}}{\Sigma_l} \right) d_{\text{vec}} \\
&= d'_{\text{vec}} \left(A^{-1} - \frac{A^{-1} D C \left(\frac{1}{\sigma_y^2 \Sigma_l} I + J - \frac{1}{\sigma_y^2 \Sigma_l} I \right) Q^{-1} C' D' A^{-1}}{\Sigma_l} \right) d_{\text{vec}} \\
&= d'_{\text{vec}} \left(A^{-1} - \frac{A^{-1} D C Q Q^{-1} C' D' A^{-1}}{\Sigma_l} \right. \\
&\quad \left. + \frac{A^{-1} D C \left(\frac{1}{\sigma_y^2 \Sigma_l} I \right) Q^{-1} C' D' A^{-1}}{\Sigma_l} \right) d_{\text{vec}} \\
&= d'_{\text{vec}} \left(A^{-1} - \frac{A^{-1} D C C' D' A^{-1}}{\Sigma_l} + \frac{A^{-1} D C Q^{-1} C' D' A^{-1}}{\sigma_y^2 \Sigma_l^2} \right) d_{\text{vec}} \\
&= \text{tr} \left(d \Sigma^{-1} d' \right) - \frac{(d \Sigma_r)' (d \Sigma_r)}{\Sigma_l} + \frac{(d \Sigma_r)' Q^{-1} (d \Sigma_r)}{\sigma_y^2 \Sigma_l^2} \\
&= \text{tr} \left(d_x \Sigma_{xx}^{-1} d'_x \right) + \frac{(d \Sigma_r)' Q^{-1} (d \Sigma_r)}{\sigma_y^2 \Sigma_l^2}, \tag{2.8}
\end{aligned}$$

čia $\Sigma_l = [\Sigma^{-1}]_{m+1, m+1}$, $\Sigma_r = [\Sigma^{-1}]_{1:(m+1), m+1}$, $\Sigma_{xx} = \Lambda_x \Lambda'_x + \text{diag}(\sigma_{x_1}^2, \dots, \sigma_{x_m}^2)$, $Q = \frac{1}{\sigma_y^2 \Sigma_l} I + J$, $d_x = z_x - \mathbf{1} \mu'_x$ ir $d = z - \mathbf{1} \mu'$, kur $z_x = (x_1, \dots, x_T)'$ ir $\mathbf{1} = 1_{T \times 1}$.

Vietoje originaliųjų modelio parametrų imami iš dalies transformuoti parametrai, nustatant $\theta = \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta \right)$, kur

$$s = e^{-\text{acosh}\left(\frac{1}{2} \frac{1}{\sigma_y^2 \Sigma_l}\right)},$$

$$\tilde{\Sigma}_{xy} = \Sigma_{xx}^{-1} \Sigma_{xy},$$

čia $\Sigma_{xy} = [\Sigma]_{1:m, m+1}$. Iš esmės $\tilde{\Sigma}_{xy}$ ir s yra surogatiniai parametrai, pakeičiantys tiesioginį Σ_{xy} ir $\Sigma_{yy} = [\Sigma]_{m+1, m+1}$ naudojimą (Σ_{xx} , Σ_{xy} ir Σ_{yy} sudaro matricą Σ , kurią reikia įvertinti). Pastebėtina, kad $s \in [0, 1]$ apibūdina santykinę išvesties matavimo triukšmo σ_y^2 svorį modelyje – didesnės reikšmės reiškia didesnę svorį.

Nors Σ_{xy} nepriklauso pasirinktam θ , jį galima nesunkiai apskaičiuoti iš θ sudedamųjų dalių:

$$\Sigma_{xy} = \Sigma_{xx} \tilde{\Sigma}_{xy}. \tag{2.9}$$

Atitinkamai Σ_{yy} turi šią išraišką:

$$\Sigma_{yy} = \frac{\sigma_y^2}{s} (s^2 + 1) + (\Sigma_{xy})' \tilde{\Sigma}_{xy} \quad (2.10)$$

(pastaba apie $\frac{\sigma_y^2}{s}$ neapibrėžtumą, kai $s = 0$, pateikiama tolesniame skirsnyje apie rekursinius skaičiavimus).

Turint taip apibrėžtą θ bei (2.7) ir (2.8) išraiškas, (2.6) tikėtinumo funkcija galiausiai pertvarkoma taip:

$$\begin{aligned} \mathcal{L} \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta \right) &= \frac{1}{T} \left(\ln \left((|\Sigma_{xx}| \sigma_y^2)^T |Q| \right) \right. \\ &\quad \left. + \text{tr} \left(d_x \Sigma_{xx}^{-1} d'_x \right) \right. \\ &\quad \left. + \frac{1}{\sigma_y^2} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right) \right) \right), \end{aligned} \quad (2.11)$$

čia $d_y = z_y - \mathbf{1}\mu_\eta$ ir $Q = \left(\frac{s^2+1}{s} \right) I + J$, kur $z_y = (\Delta y_1, \dots, \Delta y_T)'$.

Iš esmės (2.11) tikėtinumo funkcija skirta (2.3)–(2.4) lygtimis apibrėžto modelio teoriniams statistiniams matams įvertinti – μ ir nenuliniais C_{zz} elementams.

2.10 pastaba. (2.11) tikėtinumo funkcija laikytina pirmojo operatyvinės PVT dalies etapo įgyvendinimo rezultatu.

2.2.2. Dalinės išvestinės

Dabar reikia apskaičiuoti išvestines $\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}}, \frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}}, \frac{\partial \mathcal{L}}{\partial \sigma_y^2}, \frac{\partial \mathcal{L}}{\partial \mu_x}, \frac{\partial \mathcal{L}}{\partial \mu_\eta}$ ir išspręsti lygtis $\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}} = 0_{m \times m}, \frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}} = 0_{m \times 1}, \frac{\partial \mathcal{L}}{\partial \sigma_y^2} = 0, \frac{\partial \mathcal{L}}{\partial \mu_x} = 0_{m \times 1}, \frac{\partial \mathcal{L}}{\partial \mu_\eta} = 0$, atitinkamai parametrų $\Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta$, taip gaunant jų pirminius įvertinius: $\hat{\Sigma}_{xx}, \hat{\tilde{\Sigma}}_{xy}, \hat{\sigma}_y^2, \hat{\mu}_x, \hat{\mu}_\eta$.

Atlikus elementarius, tačiau gausius skaičiavimus, kurie dėl paprastumo yra praleidžiami, gaunamos šios išraiškos:

$$\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}} = \Sigma_{xx}^{-1} - \frac{\Sigma_{xx}^{-1} d'_x d_x \Sigma_{xx}^{-1}}{T} = 0_{m \times m} \implies \quad (2.12)$$

$$\hat{\Sigma}_{xx} = \frac{d'_x d_x}{T}, \quad (2.13)$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}} = \frac{2}{T\sigma_y^2} \left(d'_x Q^{-1} d_x \tilde{\Sigma}_{xy} - d'_x Q^{-1} d_y \right) = 0_{m \times 1} \implies \quad (2.14)$$

$$\hat{\tilde{\Sigma}}_{xy} = (d'_x Q^{-1} d_x)^{-1} d'_x Q^{-1} d_y, \quad (2.15)$$

$$\frac{\partial \mathcal{L}}{\partial \sigma_y^2} = \frac{1}{\sigma_y^2} - \frac{1}{T} \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{(\sigma_y^2)^2} = 0 \implies \quad (2.16)$$

$$\hat{\sigma}_y^2 = \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{T}, \quad (2.17)$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mu_x} = \frac{2}{T\sigma_y^2} & \left(\left(\mathbf{1}' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \tilde{\Sigma}'_{xy} + \sigma_y^2 T \Sigma_{xx}^{-1} \right) \mu_x \right. \\ & \left. + \left(d_y - z_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} - \sigma_y^2 \Sigma_{xx}^{-1} z'_x \mathbf{1} \right) = 0_{m \times 1} \implies \end{aligned} \quad (2.18)$$

$$\begin{aligned} \hat{\mu}_x = & \left(\mathbf{1}' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \tilde{\Sigma}'_{xy} + \sigma_y^2 T \Sigma_{xx}^{-1} \right)^{-1} \left(\sigma_y^2 \Sigma_{xx}^{-1} z'_x \mathbf{1} \right. \\ & \left. - \left(d_y - z_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \right), \end{aligned} \quad (2.19)$$

$$\frac{\partial \mathcal{L}}{\partial \mu_\eta} = \frac{2}{T\sigma_y^2} \left(d_x \tilde{\Sigma}_{xy} - d_y \right)' Q^{-1} \mathbf{1} = 0 \implies \quad (2.20)$$

$$\hat{\mu}_\eta = \frac{\left(z_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}. \quad (2.21)$$

Pažymėtina, kad gauti pirminiai įvertiniai yra stipriai priklausomi vienas nuo kito, kadangi $\hat{\Sigma}_{xx} = \hat{\Sigma}_{xx}(\mu_x)$, $\hat{\Sigma}_{xy} = \hat{\Sigma}_{xy}(s, \mu_x, \mu_\eta)$, $\hat{\sigma}_y^2 = \hat{\sigma}_y^2(s, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)$, $\hat{\mu}_x = \hat{\mu}_x(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_\eta)$, $\hat{\mu}_\eta = \hat{\mu}_\eta(s, \tilde{\Sigma}_{xy}, \mu_x)$.

Tačiau galutinis siekis (aptariamam tolesniame skirsnyje) yra gauti tokius $\hat{\Sigma}_{xx}$, $\hat{\Sigma}_{xy}$, $\hat{\sigma}_y^2$, $\hat{\mu}_x$, $\hat{\mu}_\eta$, kurie įgalintų išreikštinius (vienkartinius) skaičiavimus, priklausomai nuo parametro s , kurio optimali reikšmė (t. y. įvertis) savo ruožtu gaunama minimizuojant dabar jau vienmatę tikėtinumo funkciją $\tilde{\mathcal{L}}(s)$ ir taip sumažinant bendrąją funkcijos $\mathcal{L}$ minimavimą pagal visus jos kintamuosius problemą, duodančią optimalų θ (žymimą θ^*).

2.11 pastaba. Tikėtinumo funkcijos sumažinimas nuo $\mathcal{L}$ iki $\tilde{\mathcal{L}}$ (kartu su išreikštiniais įvertiniais redukuotiems parametrams) sudaro antrąjį operatyvinės PVT dalies etapą.

2.2.3. Didžiausio tikėtinumo įvertiniai

Manipuliuojant išvestinėmis ir iš jų gautais įvertiniais (2.12)–(2.21), galima sudaryti formules palaipsniniam θ^* komponentų radimui:

$$\mu_x^* = \hat{\mu}_x, \quad (2.22)$$

$$\Sigma_{xx}^* = \hat{\Sigma}_{xx}, \quad (2.23)$$

$$s^* = \arg \min_{s \in [0,1]} \tilde{\mathcal{L}}(s), \quad (2.24)$$

$$\tilde{\Sigma}_{xy}^* = \hat{\Sigma}_{xy}(s^*), \quad (2.25)$$

$$\mu_\eta^* = \hat{\mu}_\eta(s^*), \quad (2.26)$$

$$\sigma_y^{2*} = \hat{\sigma}_y^2(s^*), \quad (2.27)$$

čia

$$\hat{\mu}_x = \frac{z'_x \mathbf{1}}{T}, \quad (2.28)$$

$$\hat{\Sigma}_{xx} = \frac{\hat{d}'_x \hat{d}_x}{T}, \quad (2.29)$$

$$\hat{\Sigma}_{xy} = \left(I_{m \times m} - \frac{(\hat{d}'_x Q^{-1} \hat{d}_x)^{-1} \hat{d}'_x Q^{-1} \mathbf{1} (\hat{d}'_x Q^{-1} \mathbf{1})'}{\mathbf{1}' Q^{-1} \mathbf{1}} \right)^{-1} \quad (2.30)$$

$$(\hat{d}'_x Q^{-1} \hat{d}_x)^{-1} \hat{d}'_x Q^{-1} \left(z_y - \frac{z'_y Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}} \mathbf{1} \right),$$

$$\hat{\mu}_\eta = \frac{(z_y - \hat{d}_x \hat{\Sigma}_{xy})' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}, \quad (2.31)$$

$$\hat{\sigma}_y^2 = \frac{(\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy})' Q^{-1} (\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy})}{T}, \quad (2.32)$$

$$\tilde{\mathcal{L}} = \frac{\ln(|Q|)}{T} + \ln(\hat{\sigma}_y^2), \quad (2.33)$$

kur $\hat{d}_x = z_x - \mathbf{1} \hat{\mu}'_x$ ir $\hat{d}_y = z_y - \mathbf{1} \hat{\mu}_\eta$.

Čia įvertinys $\hat{\mu}_x$ gaunamas įterpus lygties $\frac{\partial \mathcal{L}}{\partial \mu_\eta} = 0$ sprendinį dėl d_y į $\frac{\partial \mathcal{L}}{\partial \mu_x}$ ir persprendus lygtį $\frac{\partial \mathcal{L}}{\partial \mu_x} = 0_{m \times 1}$ dėl μ_x .

Įvertinys $\hat{\Sigma}_{xx}$ yra kaip (2.13), su įterptu $\hat{\mu}_x$ pagal (2.28).

Įvertinys $\hat{\Sigma}_{xy}$ gaunamas įterpus $\hat{\mu}_x$ pagal (2.28) ir $\hat{\mu}_\eta(s, \hat{\Sigma}_{xy}, \hat{\mu}_x)$ pagal (2.21) į (2.15), ir iš gautos išraiškos išreiškus $\hat{\Sigma}_{xy}$.

Įvertiniai $\hat{\mu}_\eta$ ir $\hat{\sigma}_y^2$ yra atitinkamai kaip (2.21) ir (2.17), įterpus $\hat{\mu}_x$ pagal (2.28), $\hat{\Sigma}_{xy}$ pagal (2.30) ir $\hat{\mu}_\eta$ pagal (2.31).

Galiausiai, funkcija $\tilde{\mathcal{L}}$ gaunama įterpus $\hat{\mu}_x$, $\hat{\Sigma}_{xx}$, $\hat{\Sigma}_{xy}$, $\hat{\mu}_\eta$ ir $\hat{\sigma}_y^2$ atitinkamai pagal (2.28)–(2.32) į (2.11), ir iš gautos išraiškos pašalinus nuo s nepriklausančius narius.

2.12 pastaba. Pastebėtina, kad dabar $\hat{\mu}_x$ ir $\hat{\Sigma}_{xx}$ nebepriklauso nuo kitų parametrų, o visa kita priklauso tik nuo s , būtent: $\hat{\Sigma}_{xy} = \hat{\Sigma}_{xy}(s)$, $\hat{\mu}_\eta = \hat{\mu}_\eta(s)$, $\hat{\sigma}_y^2 = \hat{\sigma}_y^2(s)$, $\tilde{\mathcal{L}} = \tilde{\mathcal{L}}(s)$.

O s yra gaunamas sprendžiant vieno kintamojo funkcijos minimizavimo uždavinį, taip sumažinant reikalingo skaitinio optimizavimo naštą (ją galima dar labiau sumažinti, įverčius apskaičiuojant (2.22)–(2.27) lygtimis nurodyta tvarka, nes tai leidžia pakartotinai panaudoti jau atliktų skaičiavimų rezultatus).

Šis pasiekimas žymi antrąjį operatyvinės PVT dalies etapą, tai yra (2.28)–(2.33) įvertiniai (įskaitant $\tilde{\mathcal{L}}$) laikytini jo įgyvendinimo rezultatu.

2.13 pastaba. Antruoju operatyvinės PVT dalies etapu gauti įvertiniai vis dar apima nepageidaujamas (sudėtingas) išraiškas, būtent $|Q|$ ir Q^{-1} .

Trečiasis ir paskutinis operatyvinės PVT dalies etapas yra išnaudoti tridiagonalinę Q struktūrą, siekiant gauti veiksmingas rekursines formules, leidžiančias stebėjimų imties dydžio T atžvilgiu kubinio laiko algoritminio sudėtingumo operacijas pakeisti tiesinio laiko algoritminio sudėtingumo operacijomis.

Šiame skyriuje išimtinai susitelkiama į funkcijos $\tilde{\mathcal{L}}$ minimizavimą skaitiniais metodais, nereikalaujančiais informacijos apie išvestines. Tam naudojama auksinio pjūvio paieška, kuri yra vienas iš tokių metodų [90].

Norint pasiekti norimą skaičiavimo efektyvumą, reikia gebėti (2.28)–(2.33) įvertinių skaičiavimus atlikti per laiką, tiesiogiai proporcingą T , o tai yra kito skirsnio tema (jeigu būtų naudojami išvestinėmis paremti minimizavimo metodai, taip pat reiktų gebėti efektyviai apskaičiuoti ir $\tilde{\mathcal{L}}$ išvestines).

2.14 pastaba. Užuoat minimizavus funkciją $\tilde{\mathcal{L}}$, iš tikrųjų būtų galima apskaičiuoti jos išvestinę pagal s , kuri priklauso tik nuo tiesiogiai (2.32) ir (2.33) išraiškose pasirodančios matricos Q , ir tuomet tam tikra prasme tinkamai suderinti gautos lygties kairę ir dešinę puses.

Kartu su išvestais kitų parametrų (2.28)–(2.32) įvertiniais ir įgyvendintu trečiuoju operatyvinės PVT dalies etapu, aptartu 2.13 pastaboje, sprendžiamai parametrų vertinimo problemai būtų galima sudaryti atitinkamą realaus laiko algoritmą.

Tačiau šios pastabos įgyvendinimas traktuotinas kaip atskiras uždavinys, todėl atskirai nagrinėjamas kitame skyriuje.

2.2.4. Rekursiniai skaičiavimai

Pagrindinis trikdys efektyviai apskaičiuoti θ^* pagal (2.22)–(2.27) – būtinybė apversti $T \times T$ dydžio matricą Q , kuri pasirodo išraiškose $u'Q^{-1}v$. Čia u ir v gali būti $T \times 1$ vektoriai arba $T \times m$ matricos.

Tačiau dėl paprastumo šiame poskyryje visi skaičiavimai atliekami laikant, kad u ir v yra $T \times 1$ vektoriai. Šiuo atveju $u'Q^{-1}v$ yra skaliaras, o vektorinis bei matricinis $u'Q^{-1}v$ atvejai susideda iš tokių skaliarinių elementų.

Palyginus su tiesioginėmis matricų operacijomis, skaičiavimo efektyvumas gali būti nedelsiant pagerintas pasinaudojus galimybe analitiškai užrašyti Q^{-1} elementus [51].

Be to, tolesnius patobulinimus, lemiančius siekiamą tiesinį sudėtingumą T atžvilgiu, galima pasiekti $u'Q^{-1}v$ užrašant rekursiškai, kai kiekvienai iteracijai atlikti reikalingas pastovus elementariųjų operacijų kiekis (taigi pastovus laikas).

Be skaičiavimų sudėtingumo laiko atžvilgiu, taip pat svarbu, kad visos naudojamos išraiškos pasižymėtų apibrėžtumo ir skaitinio stabilumo savybėmis. Šie aspektai yra nagrinėjami šiame skirsnyje.

Esminę rolę čia atlieka dydis

$$c_T = \frac{1}{T} \frac{([u]_{1:T})' ([Q]_{1:T,1:T})^{-1} [v]_{1:T}}{q_T},$$

kur $q_T = \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}$ yra naudojamas dėl atvejo $s = 0$, o dalyba iš T naudojama dėl normalizavimo.

Taigi užuot (2.30)–(2.33) įvertiniuose skaičiavus formos $u'Q^{-1}v$ narius, bus skaičiuojami formos c_T nariai, nes $\frac{1}{T} \frac{1}{q_T}$ įvertiniuose $\hat{\Sigma}_{xy}$ ir $\hat{\mu}_\eta$ tiesiog išsiprastina, o c_T naudojimas įvertiniuose $\hat{\sigma}_y^2$ ir $\tilde{\mathcal{L}}$ teikia veiksmingas išraiškas, nes tokiu atveju jie apskaičiuojami taip:

$$\hat{\sigma}_y^2 = q_T g = s \frac{f_T}{f_{T+1}} g, \quad (2.34)$$

$$\tilde{\mathcal{L}} = h + \ln(g), \quad (2.35)$$

čia

$$h = \frac{\ln(|Q|)}{T} + \ln(q_T) = \ln\left(\frac{f_T}{(f_{T+1})^{1-\frac{1}{T}}}\right), \quad (2.36)$$

$$g = \frac{\hat{\sigma}_y^2}{q_T}, \quad (2.37)$$

kur g yra c_T su $u = v = \hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy}$, o $f_T = \sum_{i=0}^{T-1} s^{2i}$ yra skaičiuotinas rekursiškai:

$$f_T = f_{T-1} + s^{2(T-1)}. \quad (2.38)$$

Savo ruožtu dydis c_T turi šią rekursinę išraišką:

$$c_T = \left(1 - \frac{1}{T}\right) \frac{f_{T-1}f_{T+1}}{f_T^2} c_{T-1} + \frac{a_T b_T}{T}, \quad (2.39)$$

kur $a_T = \frac{([u]_{1:T})'[(Q)_{1:T,1:T}]^{-1}]_{1:T,T}}{q_T}$ ir $b_T = \frac{([v]_{1:T})'[(Q)_{1:T,1:T}]^{-1}]_{1:T,T}}{q_T}$ taip pat yra skaičiuotini rekursiškai:

$$a_T = s \frac{f_{T-1}}{f_T} a_{T-1} + [u]_T, \quad b_T = s \frac{f_{T-1}}{f_T} b_{T-1} + [v]_T. \quad (2.40)$$

Visų rekursijų pradinės reikšmės, būtent a_1, b_1, c_1 ir f_1 , yra:

$$a_1 = [u]_1, \quad b_1 = [v]_1, \quad c_1 = [u]_1[v]_1, \quad f_1 = 1. \quad (2.41)$$

(2.34)–(2.41) išraiškos leidžia apskaičiuoti θ^* per laiką, proporcingą stebėjimų imties dydžiui T .

Galiausiai, remiantis (2.34), s išsiprastina naryje $\frac{\hat{\sigma}_y^2}{s}$, kuris yra reikalingas apskaičiuoti Σ_{yy} įvertį (žr. (2.10)), gaunant:

$$\frac{\hat{\sigma}_y^2}{s} = \frac{f_T}{f_{T+1}} g. \quad (2.42)$$

2.15 pastaba. (2.34)–(2.42) formulės, kurių esmė yra tinkamai parinktų narių, kuriais išreiškiami parametrų įvertiniai, rekursinis apskaičiavimas, laikytinos trečiojo operatyvinės PVT dalies etapo įgyvendinimo rezultatu.

2.2.5. Vertinimo algoritmas

Gauti rezultatai apibendrinami sudaryto algoritmo pseudokodu, kuris ir yra šio poskyrio tikslas.

2.1 algoritmas: Efektyvus vienmatės išvesties grynųjų laiko eilučių duomenų KSLM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis didžiausio tikėtino parametru vertinimas esant fiksuotam duomenų rinkiniui.

Įvestis: Stebėjimų imtis $z_t, t = 1, \dots, T$

Išvestis: Įverčiai $\hat{\mu}, \hat{\Sigma}, \hat{\sigma}_y^2$

- 1: Apskaičiuoti $\hat{\mu}_x$ ir $\hat{\Sigma}_{xx}$ atitinkamai pagal (2.28) ir (2.29)
 - 2: Rasti $\hat{s}$ sprendžiant $\arg \min_{s \in [0,1]} \tilde{\mathcal{L}}(s)$ metodu be išvestinių, naudojant (2.35), su kiekvienu $u'Q^{-1}v$ formos nariu, pakeistu į c_T , o kiekvieną c_T apskaičiuojant vienalaikiu (t. y. viename `for` cikle) (2.38)–(2.41) rekursijų skaičiavimu
 - 3: Apskaičiuoti $\hat{\Sigma}_{xy}(\hat{s}), \hat{\mu}_y(\hat{s})$ ir $\frac{\hat{\sigma}_y^2(\hat{s})}{\hat{s}}$ atitinkamai pagal (2.30), (2.31) ir (2.42), su kiekvienu $u'Q^{-1}v$ formos nariu, pakeistu į c_T , o kiekvieną c_T apskaičiuojant vienalaikiu (2.38)–(2.41) rekursijų skaičiavimu
 - 4: Apskaičiuoti $\hat{\Sigma}_{xy}$ pagal (2.9) (naudojant $\hat{\Sigma}_{xx}$ ir $\hat{\Sigma}_{xy}$) ir $\hat{\Sigma}_{yy}$ pagal (2.10) (naudojant $\hat{s}, \hat{\Sigma}_{xy}, \hat{\Sigma}_{xy}$ ir $\frac{\hat{\sigma}_y^2}{\hat{s}}$)
 - 5: Sudaryti $\hat{\mu}$ iš $\hat{\mu}_x$ ir $\hat{\mu}_y$; sudaryti $\hat{\Sigma}$ iš $\hat{\Sigma}_{xx}, \hat{\Sigma}_{xy}$ ir $\hat{\Sigma}_{yy}$; $\hat{\sigma}_y^2$ priskirti išraiškos $\hat{s} \cdot \frac{\hat{\sigma}_y^2}{\hat{s}}$ rezultata
-

Kitame poskyryje atliekamas skaitinis 2.1 algoritmo tyrimas.

2.3. Empiriniai rezultatai

Šis poskyris skirtas empiriškai patikrinti 2.1 algoritmo veikimą. Siekiant šio tikslo, atlikta keletas eksperimentų, kuriais siekta:

- patikrinti algoritmu gaunamų parametru įverčių konvergavimą didėjant T ;
- palyginti algoritmu gaunamus parametrų įverčius su tradicinio DTM įverčiais (t. y. gaunamais tiesiogiai minimizuojant (2.6) tikėtino funkciją) ir empiriniais įverčiais (t. y. klasikiniu vidurkių ir kovariacijų vertinimo metodu);

- palyginti algoritmo vykdymo trukmę su šiuos metodus įgyvendinančių algoritmų vykdymo trukmėmis.

Eksperimentai atlikti naudojant R statistinį paketą [92], tiksliau – EMLI paketą (žr. skyrių „Priedas“ ir [29]), taip pat specifinį kodą, skirtą kreipiniams į EMLI funkcijas, konkuruojantiems algoritmams įvesti ir rezultatams pateikti.

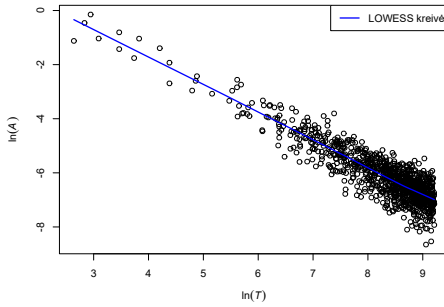
Konvergavimo eksperimentų atveju pirmiausia fiksuotos tam tikros m , k , L , σ ir μ reikšmės. Tuomet, naudojant EMLI paketo funkciją `generate_data`, sugeneruota 1000 duomenų imčių, kurių dydžiai žymimi T_i , $i = 1, \dots, 1000$, kur T_i priskiriamas operacijos `as.integer(runif(1, 10, 10000))` rezultatas.

Šioms duomenų imtims taikyta EMLI paketo funkcija `estimate_parameters`, skirta parametrų įverčiams gauti, su 10^{-5} konvergencijos tolerancijos (angl. *convergence tolerance*) slenksčiu. Gautos išvestys vėliau perduotos EMLI paketo funkcijai `evaluate_estimates`, skirtai įverčių tikslumui įvertinti. Šios funkcijos išvestis T_i dydžio imčiai žymima A_i , $i = 1, \dots, 1000$.

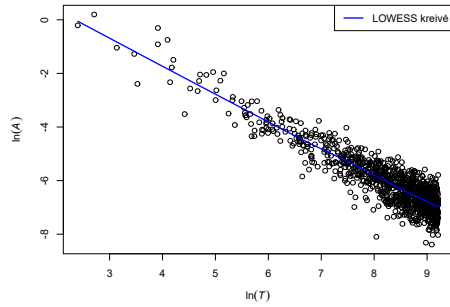
Kiekvienam fiksuotam parametrų rinkiniui m , k , L , σ ir μ sukurta sklaidos diagrama, kurioje kiekvienas taškas atitinka skaičių porą $(\ln(T_i), \ln(A_i))$. Visose sklaidos diagramose taip pat įbrėžta mėlyna LOWESS kreivė [21], kuri suglotnina individualių eksperimentų rezultatus.

Šie rezultatai papildomai pateikiami lentelės pavidalu, kurioje, su-skirsčius T reikšmes į keturis nuoseklius intervalus, pateikiamos į juos patenkančių taškų (T_i, A_i) kertinės statistinės charakteristikos.

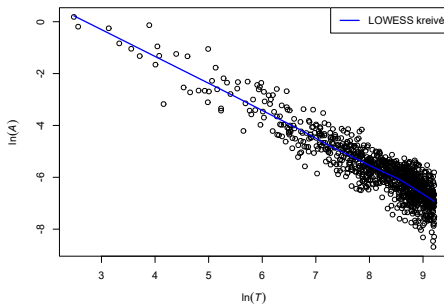
Iš viso atlikti eksperimentai su šešiais skirtingais parametrų rinkiniais m , k , L , σ ir μ . Rezultatai, kartu su pasirinktomis su modeliu susijusiomis reikšmėmis, pateikti toliau esančiuose 2.1 pav. ir 2.1 lentelėje bei juos lydinčioje informacijoje.



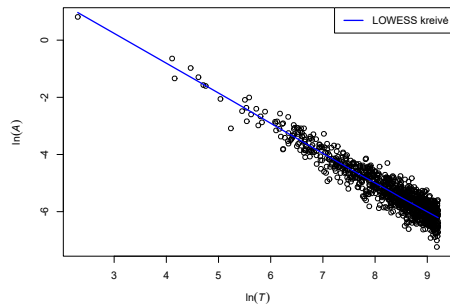
(a)



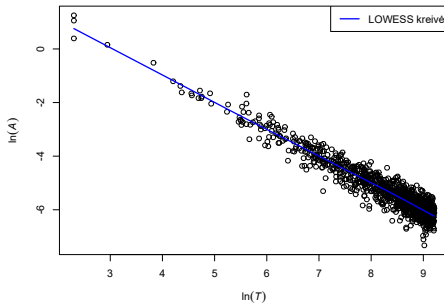
(b)



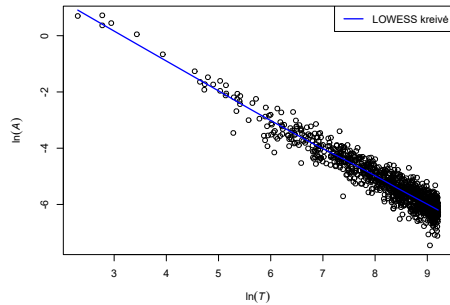
(c)



(d)



(e)



(f)

2.1 pav.: 2.1 algoritmo konvergavimo patikrinimas esant skirtingiems modelio parametrams: (a) parametų rinkinys Nr. 1; (b) parametų rinkinys Nr. 2; (c) parametų rinkinys Nr. 3; (d) parametų rinkinys Nr. 4; (e) parametų rinkinys Nr. 5; (f) parametų rinkinys Nr. 6.

2.1 lentelė: 2.1 algoritmo konvergavimo patikrinimas esant skirtingiems modelio parametrams.

T	[10, 2500]	(2500, 5000]	(5000, 7500]	(7500, 10000]
Parametrų rinkinys Nr. 1				
taškai	232	249	275	244
A min.	0,0011	0,0006	0,0003	0,0002
A vid.	0,0273	0,0026	0,0015	0,0012
A med.	0,0076	0,0023	0,0014	0,001
A maks.	0,8628	0,0078	0,0039	0,0042
A sn	0,0874	0,0013	0,0007	0,0006
Parametrų rinkinys Nr. 2				
taškai	242	236	247	275
A min.	0,0009	0,0003	0,0005	0,0002
A vid.	0,0343	0,0027	0,0016	0,0011
A med.	0,0085	0,0025	0,0015	0,001
A maks.	1,2182	0,008	0,0049	0,0038
A sn	0,116	0,0013	0,0008	0,0005
Parametrų rinkinys Nr. 3				
taškai	248	251	273	228
A min.	0,0015	0,0006	0,0004	0,0002
A vid.	0,0443	0,0036	0,002	0,0013
A med.	0,011	0,0033	0,0019	0,0012
A maks.	1,2031	0,0156	0,0052	0,004
A sn	0,1296	0,0016	0,001	0,0007
Parametrų rinkinys Nr. 4				
taškai	240	247	267	246
A min.	0,0039	0,0022	0,0013	0,0007
A vid.	0,0392	0,0059	0,0034	0,0024
A med.	0,0157	0,0055	0,0033	0,0023
A maks.	2,253	0,0159	0,008	0,0053
A sn	0,1527	0,0023	0,0011	0,0008
Parametrų rinkinys Nr. 5				
taškai	251	217	264	268
A min.	0,0044	0,0016	0,0012	0,0007
A vid.	0,071	0,0061	0,0034	0,0023
A med.	0,0174	0,0058	0,0032	0,0022

T	[10, 2500]	(2500, 5000]	(5000, 7500]	(7500, 10000]
A maks.	3,5036	0,0148	0,0075	0,0053
A sn	0,3094	0,0024	0,0011	0,0008
Parametrų rinkinys Nr. 6				
taškai	275	246	219	260
A min.	0,0033	0,0015	0,0014	0,0006
A vid.	0,0596	0,0058	0,0035	0,0024
A med.	0,0163	0,0055	0,0034	0,0023
A maks.	2,063	0,0131	0,0091	0,0055
A sn	0,2249	0,0022	0,0012	0,0008

Konvergavimo eksperimentuose naudoti parametrų rinkiniai pateikiami šiame sąrašė:

$$1. m = 2, k = 1, L = \begin{pmatrix} 5,784996 \\ 4,489983 \\ -9,822689 \end{pmatrix}, \sigma = \begin{pmatrix} 5,7348710 \\ 7,3034459 \\ 4,2458612 \\ 0,7684022 \end{pmatrix}, \mu =$$

$$\begin{pmatrix} 0,6637255 \\ -4,2957524 \\ 7,2816166 \end{pmatrix} \text{ (atitinkamas } s = 0,009926739);$$

$$2. m = 2, k = 1, L = \begin{pmatrix} 7,194264 \\ -8,883094 \\ -2,453192 \end{pmatrix}, \sigma = \begin{pmatrix} 0,07254785 \\ 4,00563476 \\ 5,11295875 \\ 7,20232401 \end{pmatrix}, \mu =$$

$$\begin{pmatrix} -0,08266602 \\ -1,70045560 \\ 1,44687459 \end{pmatrix} \text{ (atitinkamas } s = 0,4986795);$$

$$3. m = 2, k = 1, L = \begin{pmatrix} -8,359644 \\ 3,495408 \\ -8,493300 \end{pmatrix}, \sigma = \begin{pmatrix} 0,63484701 \\ 0,01266988 \\ 0,01098159 \\ 8,59915832 \end{pmatrix}, \mu =$$

$$\begin{pmatrix} -8,086132 \\ -8,308135 \\ 3,147155 \end{pmatrix} \text{ (atitinkamas } s = 0,99621);$$

$$4. m = 4, k = 1, L = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (atitinkamas)}$$

$$s = 0,3510004);$$

$$5. m = 4, k = 2, L = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (atitinkamas)}$$

$$s = 0,3479092);$$

$$6. m = 4, k = 4, L = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (atitinka-}$$

$$\text{mas } s = 0,3461196).$$

2.1 pav. ir 2.1 lentelėje pavaizduoti keli atlikti eksperimentai, apimantys įvairius modelių ir parametų pasirinkimus, dėl kurių skiriasi modelio identifikavimo ir parametro s scenarijai. Didėjant stebėjimų imties dydžiui T , visais atvejais stebima aiški parametų įverčių vidutinės paklaidos mažėjimo (artėjimo prie nulio) tendencija, o tai empiriškai patvirtina 2.1 algoritmo konvergavimą.

Toliau nagrinėjami santykinės kokybės eksperimentai. Šiems eksperimentams užfiksuota: $m = 2, k = 1, L = 1_{3 \times 1}, \sigma_{x_1} = \sigma_{x_2} = \sigma_{\eta} = 1$ ir $\mu = 0_{3 \times 1}$. Pasirinktiems T sugeneruota po 1000 s reikšmių, žymimų s_i , $i = 1, \dots, 1000$, kur kiekvienam s_i priskirtas operacijos `runif(1, 0, 1)` rezultatas.

Gautoms s reikšmėms apskaičiuoti atitinkami σ_y ir panaudota EMLI paketo funkcija `generate_data`. Paskui su sugeneruotais duomenimis kreiptasi tiek į EMLI paketo funkciją `estimate_parameters` (naudojant 10^{-5} konvergencijos tolerancijos slenkstį), tiek į nepaketinę funkciją,

vykdančią 2.2 algoritmą, kuris yra lengvai apskaičiuojama DTM alternatyva [57].

2.2 algoritmas: Vienmatės išvesties grynųjų laiko eilučių duomenų KS-LM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis empirinis parametru vertinimas esant fiksuotam duomenų rinkiniui.

Įvestis: Stebėjimų imtis $z_t, t = 1, \dots, T$

Išvestis: Įverčiai $\hat{\mu}, \hat{\Sigma}, \hat{\sigma}_y^2$

- 1: $\hat{\mu} \leftarrow \frac{1}{T} \sum_{t=1}^T z_t$
 - 2: $\hat{\Sigma} \leftarrow \frac{1}{T-1} \sum_{t=1}^T (z_t - \hat{\mu})(z_t - \hat{\mu})'$
 - 3: $\hat{\sigma}_y^2 \leftarrow - \left[\frac{1}{T-2} \sum_{t=1}^{T-1} \left(z_t - \left(\hat{\mu} - \frac{z_T}{T} \right) \frac{T}{T-1} \right) \left(z_{t+1} - \left(\hat{\mu} - \frac{z_1}{T} \right) \frac{T}{T-1} \right)' \right]_{(m+1), (m+1)}$
-

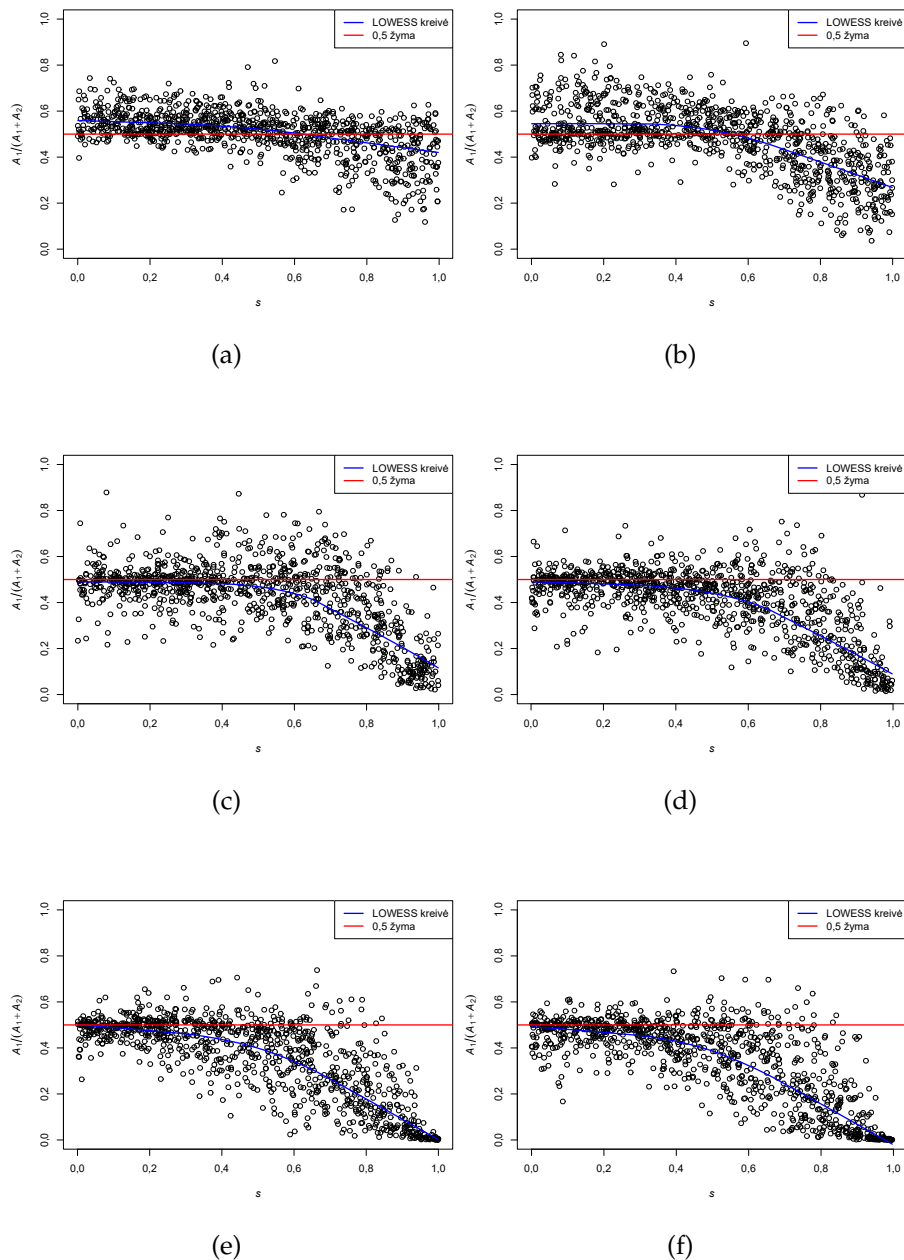
Galiausiai, tiek su 2.1 algoritmu, tiek su 2.2 algoritmu gautais įverčiais kreiptasi į EMLI paketo funkciją `evaluate_estimates`, kurios išvestis pažymėta atitinkamai $A_{1,i}$ ir $A_{2,i}, i = 1, \dots, 1000$.

Kiekvienai pasirinktai T reikšmei sudaryta po sklaidos diagramą, kurioje kiekvienas taškas atitinka skaičių porą $\left(s_i, \frac{A_{1,i}}{A_{1,i} + A_{2,i}} \right)$. Kiekvienoje sklaidos diagramoje taip pat įbrėžta mėlyna LOWESS kreivė, suglotninanti individualių eksperimentų rezultatus, bei raudona 0,5 lygio žyma, žyminti lygiavertį abiejų algoritmų parametru įverčių tikslumą. Vertės žemiau šios žymos rodo didesnę 2.1 algoritmo tikslumą, o vertės virš jos – didesnę 2.2 algoritmo tikslumą.

Pažymėtina, kad 2.2 algoritmas gali pateikti neprasmingus įverčius. Tai reiškia, kad atitinkamas $\hat{s}$, apskaičiuotas iš $\hat{\Sigma}$ ir $\hat{\sigma}_y^2$, gali būti kompleksinis (tai rodo, kad $\hat{C}_{zz}$ nėra kovariacijų matrica) arba neigiamas (tai rodo, kad $\hat{\sigma}_y^2$ yra neigiamas). Tokie atvejai siejami su modelio netinkamumu duomenims, kurių savo ruožtu gali lemti netinkamai apibrėžtas modelis arba imties atrankos sąlygoti svyravimai ir iš jų kylančios duomenų anomalijos [31].

Kuo vertinti naudojama mažesnė duomenų imtis, tuo tokie atvejai yra labiau tikėtini. Užuoat dirbtinai koregavus nelogiškus įverčius, eksperimentai atlikti tik su tomis imtimis, kurios nesukėlė jokių prieštaravimų. Tam tikra prasme 2.1 algoritmas buvo lyginamas su „geresniąja“ 2.2 algoritmo versija. O 2.1 algoritmas šios problemos neturi.

Iš viso atlikti eksperimentai su šešiomis skirtingomis T reikšmėmis. Rezultatai kartu su pasirinktomis T reikšmėmis pateikiami 2.2 pav.



2.2 pav.: 2.1 algoritmo lyginimas su 2.2 algoritmu esant skirtingiems stebėjimų imties dydžiams: (a) $T = 10$; (b) $T = 20$; (c) $T = 50$; (d) $T = 100$; (e) $T = 1000$; (f) $T = 10000$.

2.2 pav. rodo, kad didėjant parametro s reikšmei, 2.1 algoritmas tampa vidutiniškai tikslesnis už 2.2 algoritmą, o šis poveikis stiprėja didėjant stebėjimų imties dydžiui T . Esant pakankamai dideliui T (≥ 50), 2.1 algoritmas visais s atvejais tampa ne mažiau tikslus už 2.2 algoritmą (abu algoritmai užtikrina lygiavertį tikslumą, kai s yra arti 0). Tačiau esant mažoms T reikšmėms ir mažiems ar vidutiniams s , 2.1 algoritmas nusileidžia 2.2 algoritmui (ignoruoiant tai, kad esant nedidelėms duomenų imtims, 2.2 algoritmas yra linkęs pateikti neprasmingus įverčius).

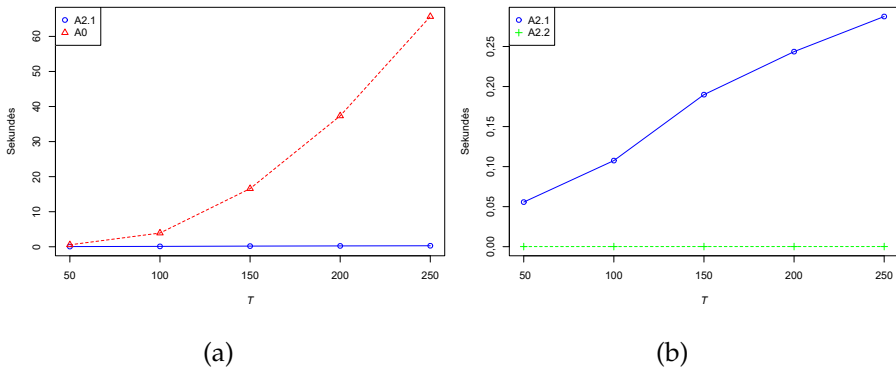
Atitinkamai 2.1 algoritmo įverčiai palyginti su minimizavus (2.6) tikėtinumo funkciją gautais įverčiais. Minimizuoti naudota R funkcija `optim` iš `stats` paketo, taikant optimizavimo metodą "BFGS". Atlikti eksperimentai parodė, kad suderinus abiejų algoritmų konvergencijos tolerancijos slenksčius ir `optim` funkcijoje nustačius konvergavimą užtikrinančią `maxit` reikšmę, abiejų algoritmų įverčiai yra ekvivalentiški.

2.16 pastaba. *Stebimi 2.1 algoritmo empiriniai rezultatai liudija apie sėkmingą DTM realizavimą. Vis dėlto tai tik užuomina, o ne įrodymas, kuris galiausiai turėtų būti atliktas analitiškai.*

Galiausiai nagrinėjami laiko eksperimentai. Šie eksperimentai atlikti naudojant R funkciją `microbenchmark` iš `microbenchmark` paketo, leidžiančią tiksliai užfiksuoti atliekamų operacijų trukmę. Kelioms pasirinktoms T reikšmėms sugeneruota po vieną duomenų imtį, su kuria, taikant visus aptartus algoritmus, po 1000 kartų apskaičiuoti parametru įverčiai, taip nustatant vidutinę kiekvieno algoritmo veikimo trukmę. Gauti rezultatai pateikiami 2.3 pav.

2.3 pav. rodo, kad sudarytas 2.1 algoritmas pasižymi tiesiniu sudėtingumu pagal T , taip reikšmingai pagreitindamas tradicinį DTM, kurio atitinkamas sudėtingumas yra kubinis. Kita vertus, 2.1 algoritmas savo greitaveika neprilygsta empiriniam vertinimui, kuris taip pat yra tiesinio sudėtingumo pagal T , tačiau turi reikšmingai mažesnę konstantą.

Šiame poskyryje pateikti skaitiniai eksperimentai patvirtina 2.1 algoritmo konvergavimą. Nors 2.1 algoritmas apskritai reikalauja daugiau skaičiavimų elementariųjų operacijų pavidalu už 2.2 algoritmą, pagal T jis vis tiek priklauso tai pačiai algoritminio sudėtingumo klasei (tiesinei), kaip ir 2.2 algoritmas. Tai leidžia jį taikyti praktiškai bet kokio dydžio



2.3 pav.: Algoritmų vidutinių vykdymo trukmių lyginimas esant skirtingiems stebėjimų imties dydžiams: (a) 2.1 algoritmas (A2.1) ir tiesioginis (2.6) tikėtinumo funkcijos minimizavimas (A0); (b) 2.1 algoritmas (A2.1) ir 2.2 algoritmas (A2.2).

sistemos stebėjimo duomenų imtims (kitaip nei įprastą DTM). Pridėjus reikšmingai geresnį nei 2.2 algoritmo parametrų vertinimo tikslumą, kuris atitinka įprasto DTM tikslumą, 2.1 algoritmas tampa konkurencinga alternatyva klasikiniam empiriniam vertinimui, ypač esant didesnėms s ir T reikšmėms.

2.17 pastaba. 2.1 algoritmas sėkmingai sujungia algoritmų, pasižyminčių žemu sudėtingumu (pvz., paremtų empiriniu vertinimu), ir algoritmų, pasižyminčių aukšta parametrų įverčių kokybe (pvz., paremtų DTM), gerąsias savybes.

2.4. Skyriaus išvados

- Vienmatės išvesties grynujų laiko eilučių duomenų KSLM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis yra tinkamas PVT principams atskleisti ir jų taikymui inicijuoti.
- PVT pagrindu sudarytas vienmatės išvesties grynujų laiko eilučių duomenų KSLM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis parametrų vertinimo algoritmas, taikomas esant fiksuotam duomenų rinkiniui, pasižymi tiesiniu sudėtingumu stebėjimų imties dydžio

atžvilgiu ir didžiausio tikėtinumo parametrų įverčiais. Tai leidžia jį apibūdinti kaip efektyvų algoritmą, sujungiantį gerąsias įprasto DTM (įverčių optimalumas) ir empirinio vertinimo (veikimo greitateika) savybes.

3. ALGORITMO ADAPTAVIMAS PARAMETRAMS VERTINTI REALIUOJU LAIKU

Šiame skyriuje, parengtame remiantis [28] straipsniu, pirmiau nagrinėta parametų vertinimo problema analizuojama realiuoju laiku. Siekiant platesnio supratimo, čia apžvelgiama (2.1)–(2.3) lygtimis apibrėžiamo kumuliacinio struktūrinių lygčių modelio (KSLM) daugiamatės išvesties versija:

$$\eta_t = \eta_{t-1} + \mu_\eta + \Gamma_0 \xi_t + \zeta_t, \quad \eta_0 = c, \quad (3.1)$$

$$y_t = \eta_t + \epsilon_t, \quad y_0 = \eta_0 + \epsilon_0, \quad (3.2)$$

$$x_t = \mu_x + \Lambda_x \xi_t + \delta_t, \quad (3.3)$$

čia $t = 1, \dots, T$ (T , kaip ir iki šiol, žymi stebėjimų imties dydį, tačiau realiuoju laiku pasižymi didėjančiu pobūdžiu); $\{\eta_0\} \cup \{\eta_t\}$ yra $n \times 1$ vektorių seka (kurios pradžia yra $n \times 1$ vektorius c), simbolizuojanti latentinę išvestį; μ_η – jos fiksuotų efektų $n \times 1$ vektorius; Γ_0 – latentinės įvesties į latentinį procesą $n \times k$ svorių matrica; $\{\xi_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių $k \times 1$ vektorių seka, simbolizuojanti latentinę įvestį (bendruosius faktorius), kur $\xi_t \sim \mathcal{N}(0_{k \times 1}, I_{k \times k})$; $\{\zeta_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių $n \times 1$ vektorių seka, simbolizuojanti latentinio proceso triukšmą, kur $\zeta_t \sim \mathcal{N}(0_{n \times 1}, \Sigma_\eta)$ (Σ_η yra $n \times n$ kovariacijų matrica); $\{y_0\} \cup \{y_t\}$ – $n \times 1$ vektorių seka, simbolizuojanti stebimą išvestį; $\{\epsilon_0\} \cup \{\epsilon_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių $n \times 1$ vektorių seka, simbolizuojanti išvesties stebėjimo triukšmą, kur ϵ_0 ir $\epsilon_t \sim \mathcal{N}(0_{n \times 1}, \Sigma_y)$ (Σ_y yra $n \times n$ kovariacijų matrica); $\{x_t\}$ – $m \times 1$ vektorių seka, ženklinanti stebimą įvestį; μ_x – jos fiksuotų efektų $m \times 1$ vektorius; Λ_x – bendrųjų faktorių koeficientų $m \times k$ matrica; $\{\delta_t\}$ – nepriklausomų ir vienodai pasiskirsčiusių $m \times 1$ vektorių seka, simbolizuojanti specifinius faktorius, kur $\delta_t \sim \mathcal{N}(0_{m \times 1}, \Sigma_x)$ (Σ_x yra įstrižaininė $m \times m$ kovariacijų matrica). Daroma prielaida, kad $\{\xi_t\}$, $\{\zeta_t\}$, $\{\delta_t\}$ ir $\{\epsilon_0\} \cup \{\epsilon_t\}$, $t = 1, \dots, T$, yra tarpusavyje nepriklausomos.

Kaip aptarta pirmajame skyriuje, efektyvių realaus laiko dinaminių struktūrinių lygčių modelių (DSLMM) vertinimo algoritmų sudarymas turi būti grindžiamas didžiausio tikėtinumo metodu (DTM) ir rekursija, kurie kartu sudaro palaipsninio vertinimo technikos (PVT) esmę. Nors antrajame skyriuje PVT pritaikyta sudarant algoritmą parametrus vertinti, kai duomenų rinkinys yra fiksuotas, gauti rezultatai gali būti

išplėsti iki atitinkamo realaus laiko algoritmo. Šis išplėtimas susijęs su tinkamu parametru, neturinčių išreikštinių įvertinių (t. y. parametro s), apdorojimu. Būtent šis apdorojimas sudaro šio skyriaus pagrindą.

Vadovaujantis analogiška motyvacija kaip ir antrajame skyriuje, šiame skyriuje toliau dirbama su stebimos išvesties y_t , $t = 0, \dots, T$, skirtumais:

$$\begin{aligned}\Delta y_t &= y_t - y_{t-1} \\ &= \mu_\eta + \Gamma_0 \xi_t + \zeta_t + (\epsilon_t - \epsilon_{t-1}),\end{aligned}\tag{3.4}$$

$t = 1, \dots, T$.

Prie μ_x pabaigos prijungus μ_η , o prie Λ_x pabaigos prijungus Γ_0 , gauti vektorius ir matrica atitinkamai žymimi μ ir L . Be to, prijungus Δy_t prie x_t , $t = 1, \dots, T$, gautas vektorius žymimas z_t , o visa seka – $z = (z_1, \dots, z_T)'$. Tuomet z vidurkis ir kovariacija atitinkamai yra:

$$Ez_t = \mu,$$

$t = 1, \dots, T$, ir

$$C_{zz}(\tau) = \begin{cases} \Sigma, & \tau = 0 \\ \Psi_1, & \tau = 1 \\ 0_{(m+n) \times (m+n)}, & \text{kitu atveju} \end{cases},$$

kur $\Sigma = LL' + \Psi_0$ (kuriame $\Psi_0 = \begin{bmatrix} \Sigma_x & 0_{m \times n} \\ 0_{n \times m} & \Sigma_\eta + 2\Sigma_y \end{bmatrix}$) ir $\Psi_1 = \begin{bmatrix} 0_{m \times m} & 0_{m \times n} \\ 0_{n \times m} & -\Sigma_y \end{bmatrix}$.

Toliau šiame skyriuje nagrinėjamas μ , Σ ir Σ_y , tai yra pagal (3.3)–(3.4) lygtis generuojamų duomenų vidurkių vektoriaus ir nenulinių kovariacijų matricos elementų vertinimas realiuoju laiku remiantis DTM (iš šių įverčių galima gauti (3.1)–(3.3) lygtimis apibrėžiamo pirminio modelio parametrus, jei jis identifikuojamas [59]). Šiuo atžvilgiu pirmajame poskyryje nagrinėjamos skirtingos (3.3)–(3.4) lygtimis apibrėžto modelio tikėtinumo funkcijos išraiškos. Antrajame poskyryje sudaromas šios užduoties algoritmas atvejui, kai $n = 1$. Trečiajame poskyryje sudarytas algoritmas tikrinamas atliekant skaitinius eksperimentus. Ketvirtajame poskyryje aptariamas pasiūlytas algoritmas ir jo išplėtimo galimybės atvejui $n \geq 1$, tai yra (3.3)–(3.4) lygtimis apibrėžtam modeliui. Galiausiai pateikiamos skyriaus išvados.

3.1. Bendrasis modelis

Šiame poskyryje nagrinėjamos (3.3)–(3.4) lygtimis apibrėžto modelio tikėtinumo funkcijos išraiškos.

3.1.1. Tiesioginė tikėtinumo funkcija

Tikėtinumo funkcija $\mathcal{L}$, kuri yra DTM optimizavimo objektas, (3.3)–(3.4) lygtimis apibrėžtam modeliui gali būti užrašyta taip:

$$\mathcal{L}(\theta) = \mathcal{L}(\theta; z) = \frac{\ln(|C_{zz}|) + d'_{\text{vec}} C_{zz}^{-1} d_{\text{vec}}}{T}, \quad (3.5)$$

čia $d_{\text{vec}} = \text{vec}(z) - \text{vec}(\underbrace{(\mu, \dots, \mu)'}_T)$ ir $C_{zz} = \Sigma \otimes I + \Psi_1 \otimes J$, kur $I = I_{T \times T}$

ir $J = J_{T \times T} = \begin{cases} 1, & |i - j| = 1 \\ 0, & \text{kitu atveju} \end{cases}, i = 1, \dots, T, j = 1, \dots, T$. Raidė θ

bendrai žymi modelio parametrus, kurie visi priklauso d_{vec} (kuriame taip pat yra ir duomenys z) ir C_{zz} .

Analogiškai antrajam skyriui, (3.5) tikėtinumo funkcija išplaukia iš vektoriaus $\text{vec}(z)$ $(m + n)T$ -mačio Gauso skirstinio, logaritvizavimo, dalybos iš T , nuo θ nepriklausančių narių pašalinimo bei maksimizavimo uždavinio pakeitimo minimizavimo uždaviniu. Ši išraiška gali būti laikoma pradine tikėtinumo funkcijos forma, kurios tiesioginis optimizavimas nėra efektyvus dėl būtinybės apversti matricą, kurios matmenys priklauso nuo T [124].

Vietoje to pasirenkama analogiška antrajam skyriui parametrų vektoriaus θ reprezentacija, su kuria toliau dirbama pagal PVT sampratą, siekiant izoliuoti parametrus nuo matricos, priklausančios nuo T . Čia tai atliekama bendruoju atveju, tai yra laikant, kad n yra bet koks natūralusis skaičius, o ne tik vienetas.

3.1.2. Pertvarkyta tikėtinumo funkcija

Konkrečiai pasirenkama $\theta = (s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \tilde{\Sigma}_y, \mu_x, \mu_\eta)$, kur

$$s = \textcircled{e}^{-\text{acosh}\left(\frac{\lambda|q|}{2}\right)},$$

$$\Sigma_{xx} = \Lambda_x \Lambda'_x + \Sigma_x,$$

$$\tilde{\Sigma}_{xy} = \Sigma_{xx}^{-1} \Sigma_{xy},$$

$$\tilde{\Sigma}_y = wv[q],$$

čia $q = w^{-1}\tilde{\Sigma}_{yy}^{-1}(w^{-1})'$, $w = v[\Sigma_y]' \text{diag}(\lambda[\Sigma_y])v[\Sigma_y]$ ir $\Sigma_{xy} = [\Sigma]_{1:m,(m+1):(m+n)}$, kur $\tilde{\Sigma}_{yy} = [\Sigma^{-1}]_{(m+1):(m+n),(m+1):(m+n)}$. Analogiškai antrajam skyriui, parametras s yra charakterizuojantis, nes apibūdina santykinę išvesties matavimo triukšmo svorį modelyje – didesnės jo reikšmės reiškia didesnę šio triukšmo svorį. Kiekvienas vektoriaus s elementas priklauso intervalui $[0, 1]$.

Naudojant tokį θ , sugrįžti prie originalių kovariacijų matricos elementų galima pagal šias formules:

$$\Sigma_{xy} = \Sigma_{xx}\tilde{\Sigma}_{xy}, \quad (3.6)$$

$$\Sigma_{yy} = \tilde{\Sigma}_y \text{diag}((s \odot s + 1_{n \times 1}) \oslash s) \tilde{\Sigma}_y' + \left(\Sigma_{xx}\tilde{\Sigma}_{xy}\right)' \tilde{\Sigma}_{xy}, \quad (3.7)$$

$$\Sigma_y = \tilde{\Sigma}_y \tilde{\Sigma}_y', \quad (3.8)$$

kur $\Sigma_{yy} = [\Sigma]_{(m+1):(m+n),(m+1):(m+n)}$.

Pritaikius antrojo skyriaus logiką daugiamatės išvesties modelio atvejui, tiesioginės (3.5) tikėtino funkcijos sudedamosios dalys gali būti išreiškiamos analogiškais formulėmis (išsamesnės informacijos apie aktualius matricinius skaičiavimus, tokius kaip Kronekerio sandauga, blokinės matricos apvertimas bei matricos skaidymas į tikrines reikšmes ir tikrinius vektorius, galima rasti, pvz., [1] arba [14]):

$$\begin{aligned} d_{\text{vec}}' C_{zz}^{-1} d_{\text{vec}} &= \text{tr} \left(d_x \Sigma_{xx}^{-1} d_x' \right) \\ &+ \left(\text{vec} \left(d_y - d_x \tilde{\Sigma}_{xy} \right) \right)' \left(\tilde{\Sigma}_{yy}^{-1} \otimes I - \Sigma_y \otimes J \right)^{-1} \text{vec} \left(d_y - d_x \tilde{\Sigma}_{xy} \right), \end{aligned} \quad (3.9)$$

$$|C_{zz}| = |\Sigma_{xx}|^T \left| \tilde{\Sigma}_{yy}^{-1} \otimes I - \Sigma_y \otimes J \right|, \quad (3.10)$$

čia $d_x = z_x - \mathbf{1}\mu_x'$ ir $d_y = z_y - \mathbf{1}\mu_y'$, kur $z_x = (x_1, \dots, x_T)'$, $z_y = (\Delta y_1, \dots, \Delta y_T)'$ ir $\mathbf{1} = 1_{T \times 1}$.

Tai atlikus, lieka matrica $\tilde{\Sigma}_{yy}^{-1} \otimes I - \Sigma_y \otimes J$, kurią reikia papildomai apdoroti, kad ji būtų tinkamai pritaikyta daugiamatės išvesties atvejui:

$$\begin{aligned} \tilde{\Sigma}_{yy}^{-1} \otimes I - \Sigma_y \otimes J &= \tilde{\Sigma}_{yy}^{-1} \otimes I - (ww') \otimes J \\ &= (w \otimes I) (q \otimes I - I_{n \times n} \otimes J) (w' \otimes I) \\ &= (w \otimes I) ((v[q] \text{diag}(\lambda[q]) v[q]') \otimes I - I_{n \times n} \otimes J) (w' \otimes I) \\ &= ((wv[q]) \otimes I) (\text{diag}(\lambda[q]) \otimes I - I_{n \times n} \otimes J) ((v[q]' w') \otimes I). \end{aligned} \quad (3.11)$$

Remiantis pasirinktu parametru vektoriumi θ ir (3.9)–(3.11) lygtimis, tiesioginė (3.5) tikėtinumo funkcija galiausiai pertvarkoma į ši ekvivalentų pavidalą:

$$\begin{aligned} \mathcal{L} \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \tilde{\Sigma}_y, \mu_x, \mu_\eta \right) = & \frac{1}{T} \left(\ln \left(\left(|\Sigma_{xx}| \left| \tilde{\Sigma}_y \tilde{\Sigma}_y' \right| \right)^T |Q| \right) \right. \\ & + \text{tr} \left(d_x \Sigma_{xx}^{-1} d_x' \right) \\ & + \left(\text{vec} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right) \left(\tilde{\Sigma}_y^{-1} \right)' \right) \right)' Q^{-1} \\ & \left. \text{vec} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right) \left(\tilde{\Sigma}_y^{-1} \right)' \right) \right), \end{aligned} \quad (3.12)$$

kur $Q = \text{diag}((s \odot s + 1_{n \times 1}) \otimes s) \otimes I - I_{n \times n} \otimes J$.

3.1 pastaba. Matrica Q pakeičia kovariacijų matricą C_{zz} (3.3)–(3.4) lygtimis apibrėžto modelio tikėtinumo funkcijoje. Ji priklauso tik nuo parametro s , o kiti su kovariacijų matrica susiję parametrai – būtent Σ_{xx} , $\tilde{\Sigma}_{xy}$ ir $\tilde{\Sigma}_y$ – yra už jos ribų, kartu su μ_x ir μ_η .

Toks parametru atskyrimas sudaro siūlomo realaus laiko DTM pagrindą, kuris iš esmės yra pakaitinis optimizavimo metodas, skirtas laipsniškai spręsti DTM uždavinį: optimizavimo problema suskaidoma į paprastesnes subproblemas, kuriose optimizuojama vieno kintamųjų rinkinio atžvilgiu, laikant kitą rinkinį fiksuotu, ir atvirkščiai. Šis procesas kartojamas iteratyviai, kiekvieną kartą pasirodžius naujiems duomenims.

3.1.3. Išskaidyta tikėtinumo funkcija

Verta atkreipti dėmesį, kad (3.12) lygtimi apibrėžta tikėtinumo funkcija parametro s atžvilgiu skyla į n atskirų narių:

$$\begin{aligned} \mathcal{L} \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \tilde{\Sigma}_y, \mu_x, \mu_\eta \right) = & \frac{1}{T} \left(\ln \left(\left(|\Sigma_{xx}| \left| \tilde{\Sigma}_y \tilde{\Sigma}_y' \right| \right)^T \right) + \text{tr} \left(d_x \Sigma_{xx}^{-1} d_x' \right) \right. \\ & + \sum_{i=1}^n \left(\ln \left(|\tilde{Q}_i| \right) + \left(\left[\left(d_y - d_x \tilde{\Sigma}_{xy} \right) \left(\tilde{\Sigma}_y^{-1} \right)' \right]_{,i} \right)' \right. \\ & \left. \left. \tilde{Q}_i^{-1} \left[\left(d_y - d_x \tilde{\Sigma}_{xy} \right) \left(\tilde{\Sigma}_y^{-1} \right)' \right]_{,i} \right) \right), \end{aligned} \quad (3.13)$$

kur $\tilde{Q}_i = \left(\frac{([s]_{i,})^2 + 1}{[s]_{i,}} \right) I - J$.

Tačiau modelyje stebima išvestis gali būti koreliuota (t. y. Σ_y , o kartu ir $\tilde{\Sigma}_y$, bendru atveju nėra įstrižainė), tad toks padalijimas nesuteikia tiesioginės galimybės išplėsti $n = 1$ atvejo rezultatų į $n > 1$.

Nepaisant to, toliau nagrinėjamas tikslinio parametrų vertinimo realiuoju laiku algoritmo sudarymas atveju, kai $n = 1$. Tai leidžia perspektyviai pažvelgti į tokio tipo algoritmų kūrimo aspektus, kuriuos vėliau galima pritaikyti, pasinaudojant galimybėmis, kurias suteikia (3.13) lygtis. Be to, kadangi, kai $n = 1$, Σ_y yra skaliaras, šis atvejis yra išskirtinai lankstus ir todėl įdomus pats savaime.

3.2. Vienmatės išvesties atvejis

Šiame poskyryje sudaromas algoritmas, skirtas (3.3)–(3.4) modelio atveju, kai $n = 1$, parametrams μ , Σ ir Σ_y vertinti realiuoju laiku, remiantis DTM. Kitaip tariant, čia pateikiama antrajame skyriuje sudaryto 2.1 algoritmo realaus laiko versija.

3.2.1. Tikėtinumo funkcija

Kai $n = 1$, (3.12) (arba ekvivalentiškai – (3.13)) lygtimi apibrėžta tikėtinumo funkcija įgyja šią formą:

$$\begin{aligned} \tilde{\mathcal{L}}(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \Sigma_y, \mu_x, \mu_\eta) = & \frac{1}{T} \left(\ln \left((|\Sigma_{xx}| \Sigma_y)^T |Q| \right) \right. \\ & + \text{tr} \left(d_x \Sigma_{xx}^{-1} d'_x \right) \\ & \left. + \frac{1}{\Sigma_y} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right) \right) \right), \end{aligned} \quad (3.14)$$

kur naudojami parametrai interpretuojami atsižvelgiant į prielaidą, kad $n = 1$. Pažymėtina, kad tokiu atveju Σ_y yra skaliaras (kaip ir s , todėl matrica $Q = \left(\frac{s^2 + 1}{s} \right) I - J$), tad kaip kintamasis yra laikomas jis pats, o ne jo faktORIZACIJA (t. y. šaknis).

Pasinaudojant antrojo skyriaus rezultatais, analitiniai parametrų μ_x , Σ_{xx} , $\tilde{\Sigma}_{xy}$, μ_η ir Σ_y įvertiniai, išreikšti per s , yra tokie (čia laikoma, kad stebėjimų imties dydis yra T):

$$\hat{\mu}_x = \frac{z'_x \mathbf{1}}{T}, \quad (3.15)$$

$$\hat{\Sigma}_{xx} = \frac{\hat{d}'_x \hat{d}_x}{T}, \quad (3.16)$$

$$\hat{\Sigma}_{xy} = \left(\hat{d}'_x Q^{-1} \hat{d}_x - \frac{\hat{d}'_x Q^{-1} \mathbf{1} \left(\hat{d}'_x Q^{-1} \mathbf{1} \right)'}{\mathbf{1}' Q^{-1} \mathbf{1}} \right)^{-1} \hat{d}'_x Q^{-1} \left(z_y - \frac{z'_y Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}} \mathbf{1} \right), \quad (3.17)$$

$$\hat{\mu}_\eta = \frac{\left(z_y - \hat{d}_x \hat{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}, \quad (3.18)$$

$$\hat{\Sigma}_y = \frac{\left(\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy} \right)' Q^{-1} \left(\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy} \right)}{T}, \quad (3.19)$$

kur $\hat{d}_x = z_x - \mathbf{1} \hat{\mu}'_x$ ir $\hat{d}_y = z_y - \mathbf{1} \hat{\mu}_\eta$.

Struktūriniu požiūriu (3.15)–(3.19) įvertiniai yra tinkami naudoti sudaromame algoritme, skirtame parametrąs vertinti realiuoju laiku. Tačiau praktiškai jų skaičiavimas turi būti atliekamas rekursiškai – kaip 2.1 algoritme, skirtame parametrąs vertinti esant fiksuotam duomenų rinkiniui.

Vis dėlto parametro s vertinimas realiuoju laiku reikalauja iš esmės kitokio sprendimo nei tas, kuris taikytas antrajame skyriuje.

3.2.2. Dalinė išvestinė

Konkrečiai šis sprendimas yra susijęs su (3.14) tikėtinumo funkcijos išvestine pagal kintamąjį s :

$$\begin{aligned} \frac{\partial \tilde{\mathcal{L}}}{\partial s} &= \frac{s^2 - 1}{s^2} \left(\text{tr} \left(\frac{Q^{-1}}{T} \right) - \frac{1}{\Sigma_y} \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-2} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{T} \right) \\ &= \frac{s^2 - 1}{s^2} \left(\frac{\frac{s^2+1}{2s} - \left(1 + \frac{1}{T}\right) [Q^{-1}]_{T,T}}{2 \left(\frac{s^2-1}{2s} \right)^2} \right. \\ &\quad \left. - \frac{[Q^{-1}]_{T,T}}{\Sigma_y} \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-2} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{T [Q^{-1}]_{T,T}} \right) \\ &= \left(1 + \frac{1}{T} \right) \left(1 - \frac{s^2 + 1}{s^2 - 1} \right) h, \end{aligned}$$

$$\text{kur } h = [Q^{-1}]_{T,T} - \frac{T}{T+1} \left(\frac{\frac{s^2+1}{2s} \frac{\Sigma_y}{[Q^{-1}]_{T,T}} - 2 \left(\frac{s^2-1}{2s} \right)^2 \frac{(d_y - d_x \tilde{\Sigma}_{xy})' Q^{-2} (d_y - d_x \tilde{\Sigma}_{xy})}{T [Q^{-1}]_{T,T}}}{\frac{\Sigma_y}{[Q^{-1}]_{T,T}}} \right).$$

3.2 pastaba. Pagal pakaitinio optimizavimo metodą, kaip ir (3.15)–(3.19) lygtimis apibrėžtų įvertinių atveju, s įvertinio, tinkamo realaus laiko DTM, konstravimas priklauso nuo subproblemos, gautos atitinkamą išvestinę prilyginus nuliui. Šis algoritmo plėtojimo metodas išnaudoja faktą, kad daugiamatžio Gauso skirstinio atveju tikėtinumo funkcija yra analitinė funkcija modelio parametrų atžvilgiu, todėl yra be galo diferencijuojama. Pagal klasikinę Ferma teoremą apie lokalius ekstremumus [136], jei optimalūs modelio parametrų įverčiai nėra apibrėžties intervalo krašte, yra stacionariame tikėtinumo funkcijos taške – tai yra taške, kuriame pirmos eilės išvestinė lygi nuliui. Nors šis rezultatas yra tik būtina tikėtinumo funkcijos optimizavimo sąlyga, jis atveria kelią realaus laiko DTM algoritmui sudaryti. Tačiau tokio algoritmo konvergavimas reiškia, kad tai taip pat turi būti ir pakankama sąlyga, kuri yra susijusi su tikėtinumo funkcijos iškilumu.

Čia aktuali tampa lygtis $h = 0$, kurios pagrindu siekiama gauti parametro s įvertinį, tinkamą skaičiuoti realiuoju laiku. Dydžio h sudėtyje esantį kintamąjį Σ_y pakeitus jo įvertiniu (žr. (3.19)), o kitus kintamuosius išlaikius nepakeistus, vietoje $h = h(s, \tilde{\Sigma}_{xy}, \Sigma_y, \mu_x, \mu_\eta)$ toliau nagrinėjama $\tilde{h} = \tilde{h}(s, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)$, nes šis žingsnis leidžia sujungti su Q^{-2} susijusį narį, esantį $\frac{\partial \tilde{\mathcal{L}}}{\partial s}$, su panašiu – su Q^{-1} susijusiu – nariu į vieną visumą, kuri siūlomame s įvertinyje sudaro veiksmingą santykį su nariu, susijusiu tik su Q^{-1} .

Pažymėję

$$a_T(\alpha, \beta) = \frac{([\alpha]_{1:T})' ([Q]_{1:T,1:T})^{-1} [\beta]_{1:T}}{T \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

$$b_T(\alpha, \beta) = \frac{([\alpha]_{1:T})' \left(\frac{s^2+1}{2s} ([Q]_{1:T,1:T})^{-1} - 2 \left(\frac{s^2-1}{2s} \right)^2 ([Q]_{1:T,1:T})^{-2} \right) [\beta]_{1:T}}{T \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

gauname tokią $\tilde{h}$ išraišką:

$$\tilde{h} = [Q^{-1}]_{T,T} - \frac{T}{T+1} \frac{b_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})}{a_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})}. \quad (3.20)$$

Siekiant sklandaus perėjimo tarp parametrų vertinimų, siūlomas algoritmas (3.20) lygties narius $a_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})$ ir $b_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})$ skaičiuoja per atominius narius, kurie priklauso tik nuo s :

$$a_{1,T} = a_T(\mathbf{1}, \mathbf{1}), \quad b_{1,T} = b_T(\mathbf{1}, \mathbf{1}), \quad (3.21)$$

$$a_{2,T} = a_T(z_x, \mathbf{1}), \quad b_{2,T} = b_T(z_x, \mathbf{1}), \quad (3.22)$$

$$a_{3,T} = a_T(z_x, z_x), \quad b_{3,T} = b_T(z_x, z_x), \quad (3.23)$$

$$a_{4,T} = a_T(z_y, \mathbf{1}), \quad b_{4,T} = b_T(z_y, \mathbf{1}), \quad (3.24)$$

$$a_{5,T} = a_T(z_y, z_y), \quad b_{5,T} = b_T(z_y, z_y), \quad (3.25)$$

$$a_{6,T} = a_T(z_x, z_y), \quad b_{6,T} = b_T(z_x, z_y). \quad (3.26)$$

Pažymėjus

$$\begin{aligned} f_T(\gamma, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta) &= \gamma_{1,T} \mu_\eta^2 - 2\gamma_{4,T} \mu_\eta + \gamma_{5,T} \\ &\quad + 2(\mu_\eta \gamma'_{2,T} - \gamma'_{6,T} + (\gamma_{4,T} - \mu_\eta \gamma_{1,T}) \mu'_x) \tilde{\Sigma}_{xy} \\ &\quad + \tilde{\Sigma}'_{xy} (\gamma_{3,T} + (\mu_x \gamma_{1,T} - 2\gamma_{2,T}) \mu'_x) \tilde{\Sigma}_{xy}, \end{aligned}$$

gaunama galutinė $\tilde{h}$ išraiška:

$$\tilde{h} = [Q^{-1}]_{T,T} - \frac{T}{T+1} \frac{f_T(b, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)}{f_T(a, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)}. \quad (3.27)$$

3.2.3. Rekursiniai skaičiavimai

Siūlomame algoritme (3.21)–(3.26) lygtimis apibrėžti nariai yra rekursinių skaičiavimų objektas, nes realiuoju laiku vertinami parametrai yra išreiškiami per juos. Pažymėjus

$$\begin{aligned} r_T^{(a)}(\gamma, \alpha) &= \left(1 - \frac{1}{T}\right) k_{3,T} \gamma + \frac{1}{T} \alpha, \\ r_T^{(b)}(\gamma, \alpha, \beta) &= \left(1 - \frac{1}{T}\right) k_{3,T} \gamma + \frac{1}{T} (k_{4,T} \alpha - k_{5,T} \beta), \end{aligned}$$

kur

$$k_{3,T} = \frac{\left[([Q]_{1:(T-1), 1:(T-1)})^{-1}\right]_{T-1, T-1}}{\left[([Q]_{1:T, 1:T})^{-1}\right]_{T, T}}, \quad (3.28)$$

$$k_{4,T} = \frac{1}{2} \left(\frac{s^2 + 1}{s} - \left(\frac{s^2 - 1}{s} \right)^2 \frac{[[Q]_{1:T,1:T})^{-2}]_{T,T}}{[[Q]_{1:T,1:T})^{-1}]_{T,T}} \right), \quad (3.29)$$

$$k_{5,T} = \left(\frac{s^2 - 1}{s} \right)^2 \left[([Q]_{1:(T-1),1:(T-1)})^{-2} \right]_{T-1,T-1}, \quad (3.30)$$

atitinkamos rekursijos yra tokios:

$$a_{1,T} = r_T^{(a)}(a_{1,T-1}, c_{1,T}^2), \quad b_{1,T} = r_T^{(b)}(b_{1,T-1}, c_{1,T}^2, d_{1,T-1}c_{1,T}), \quad (3.31)$$

$$a_{2,T} = r_T^{(a)}(a_{2,T-1}, c_{2,T}c_{1,T}), \quad b_{2,T} = r_T^{(b)}\left(b_{2,T-1}, c_{2,T}c_{1,T}, \frac{d_{1,T-1}c_{2,T} + d_{2,T-1}c_{1,T}}{2}\right), \quad (3.32)$$

$$a_{3,T} = r_T^{(a)}(a_{3,T-1}, c_{2,T}c'_{2,T}), \quad b_{3,T} = r_T^{(b)}\left(b_{3,T-1}, c_{2,T}c'_{2,T}, \frac{d_{2,T-1}c'_{2,T} + c_{2,T}d'_{2,T-1}}{2}\right), \quad (3.33)$$

$$a_{4,T} = r_T^{(a)}(a_{4,T-1}, c_{3,T}c_{1,T}), \quad b_{4,T} = r_T^{(b)}\left(b_{4,T-1}, c_{3,T}c_{1,T}, \frac{d_{3,T-1}c_{1,T} + d_{1,T-1}c_{3,T}}{2}\right), \quad (3.34)$$

$$a_{5,T} = r_T^{(a)}(a_{5,T-1}, c_{3,T}^2), \quad b_{5,T} = r_T^{(b)}(b_{5,T-1}, c_{3,T}^2, d_{3,T-1}c_{3,T}), \quad (3.35)$$

$$a_{6,T} = r_T^{(a)}(a_{6,T-1}, c_{2,T}c_{3,T}), \quad b_{6,T} = r_T^{(b)}\left(b_{6,T-1}, c_{2,T}c_{3,T}, \frac{d_{2,T-1}c_{3,T} + d_{3,T-1}c_{2,T}}{2}\right). \quad (3.36)$$

Pažymėjus

$$c_T(\alpha) = \frac{([\alpha]_{1:T})' \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}}{\left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

$$d_T(\alpha) = \frac{([\alpha]_{1:T})' \left[([Q]_{1:T,1:T})^{-2} \right]_{T,T}}{\left[([Q]_{1:T,1:T})^{-2} \right]_{T,T}},$$

čia c ir d nariai yra:

$$c_{1,T} = c_T(\mathbf{1}), \quad d_{1,T} = d_T(\mathbf{1}), \quad (3.37)$$

$$c_{2,T} = c_T(z_x), \quad d_{2,T} = d_T(z_x), \quad (3.38)$$

$$c_{3,T} = c_T(z_y), \quad d_{3,T} = d_T(z_y). \quad (3.39)$$

Pradinės reikšmės (t. y. kai $T = 1$) (3.31)–(3.36) lygtimis apibrėžtoms rekursijoms yra šios:

$$a_{1,1} = 1, \quad b_{1,1} = a_{1,1} \frac{2s}{s^2 + 1}, \quad (3.40)$$

$$a_{2,1} = [z'_x]_{,1}, \quad b_{2,1} = a_{2,1} \frac{2s}{s^2 + 1}, \quad (3.41)$$

$$a_{3,1} = [z'_x]_{,1} \left([z'_x]_{,1} \right)', \quad b_{3,1} = a_{3,1} \frac{2s}{s^2 + 1}, \quad (3.42)$$

$$a_{4,1} = [z_y]_{1,}, \quad b_{4,1} = a_{4,1} \frac{2s}{s^2 + 1}, \quad (3.43)$$

$$a_{5,1} = \left([z_y]_{1,} \right)^2, \quad b_{5,1} = a_{5,1} \frac{2s}{s^2 + 1}, \quad (3.44)$$

$$a_{6,1} = [z'_x]_{,1} [z_y]_{1,}, \quad b_{6,1} = a_{6,1} \frac{2s}{s^2 + 1}. \quad (3.45)$$

Antraeiliai nariai, apibrėžti (3.37)–(3.39) lygtimis, taip pat yra skaičiuotini rekursiškai. Pažymėjus

$$\begin{aligned} r_T^{(c)}(\gamma, \alpha) &= k_{1,T}\gamma + \alpha, \\ r_T^{(d)}(\gamma, \alpha) &= k_{2,T}\gamma + \alpha, \end{aligned}$$

kur

$$k_{1,T} = \left[([Q]_{1:(T-1),1:(T-1)})^{-1} \right]_{T-1,T-1}, \quad (3.46)$$

$$k_{2,T} = \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T} \frac{\left[([Q]_{1:(T-1),1:(T-1)})^{-2} \right]_{T-1,T-1}}{\left[([Q]_{1:T,1:T})^{-2} \right]_{T,T}}, \quad (3.47)$$

atitinkamos rekursijos yra tokios:

$$c_{1,T} = r_T^{(c)}(c_{1,T-1}, 1), \quad d_{1,T} = r_T^{(d)}(d_{1,T-1}, c_{1,T}), \quad (3.48)$$

$$c_{2,T} = r_T^{(c)}\left(c_{2,T-1}, [z'_x]_{,T}\right), \quad d_{2,T} = r_T^{(d)}(d_{2,T-1}, c_{2,T}), \quad (3.49)$$

$$c_{3,T} = r_T^{(c)}\left(c_{3,T-1}, [z_y]_{T,}\right), \quad d_{3,T} = r_T^{(d)}(d_{3,T-1}, c_{3,T}). \quad (3.50)$$

Pradinės reikšmės (t. y. kai $T = 1$) (3.48)–(3.50) lygtimis apibrėžtoms rekursijoms yra šios:

$$c_{1,1} = 1, \quad d_{1,1} = c_{1,1}, \quad (3.51)$$

$$c_{2,1} = [z'_x]_{,1}, \quad d_{2,1} = c_{2,1}, \quad (3.52)$$

$$c_{3,1} = [z_y]_{1,}, \quad d_{3,1} = c_{3,1}. \quad (3.53)$$

3.2.4. Asimptotinės aproksimacijos

Pateiktuose rekursiniuose skaičiavimuose (3.28)–(3.30) ir (3.46)–(3.47) dydžiai vis dar kelia skaičiavimo efektyvumo problemų, nes apima matricos Q apvertimą bei neapibrėžtumą ties $s = 0$. Tačiau galima pastebėti, kad jie turi tokias alternatyvias išraiškas:

$$k_{1,T} = \begin{cases} 0, & s = 0 \\ s \frac{1-s^{2(T-1)}}{1-s^{2T}}, & 0 < s < 1, \\ 1 - \frac{1}{T}, & s = 1 \end{cases} \quad (3.54)$$

$$k_{2,T} = \begin{cases} 0, & s = 0 \\ s \frac{1-s^{2(T+1)}}{1-s^{2T}} \frac{1-s^{2(2T-1)} - (1-s^2)s^{2(T-1)}(2T-1)}{1-s^{2(2T+1)} - (1-s^2)s^{2T}(2T+1)}, & 0 < s < 1, \\ \left(1 - \frac{1}{T}\right) \frac{2T-1}{2T+1}, & s = 1 \end{cases} \quad (3.55)$$

$$k_{3,T} = \begin{cases} 1, & s = 0 \\ \frac{(1-s^{2(T-1)})(1-s^{2(T+1)})}{(1-s^{2T})^2}, & 0 < s < 1, \\ 1 - \frac{1}{T^2}, & s = 1 \end{cases} \quad (3.56)$$

$$k_{4,T} = \begin{cases} 0, & s = 0 \\ s + \frac{1-s^2}{s} \left(T \frac{s^{2T}}{1-s^{2T}} - (T+1) \frac{s^{2(T+1)}}{1-s^{2(T+1)}} \right), & 0 < s < 1, \\ 1, & s = 1 \end{cases} \quad (3.57)$$

$$k_{5,T} = \begin{cases} 1, & s = 0 \\ (1-s^2) \frac{1-s^{2(2T-1)} - s^{2(T-1)}(1-s^2)(2T-1)}{(1-s^{2T})^2}, & 0 < s < 1, \\ 0, & s = 1 \end{cases} \quad (3.58)$$

kur kraštiniai atvejai (t. y. $s = 0$ ir $s = 1$) yra gaunami iš atitinkamo nekraštinio atvejo (t. y. $0 < s < 1$), atliekant tolydųjį priartėjimą.

Dydžiai k , išreikšti (3.54)–(3.58) lygtimis, nebereikalauja matricos apvertimo ir yra apibrėžti visoms parametro s reikšmėms. Tačiau kadangi DTM savybės pasireiškia tik didelėse duomenų imtyse ir

$$\lim_{T \rightarrow \infty} k_{1,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{2,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{3,T} = 1,$$

$$\lim_{T \rightarrow \infty} k_{4,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{5,T} = 1 - s^2$$

pakankamai sparčiai, siūlomam algoritmui nustatomos dydžių k ribos, tai yra vietoje dydžių k naudojami dydžiai $\tilde{k}$:

$$\tilde{k}_{1,T} = s \approx k_{1,T}, \quad (3.59)$$

$$\tilde{k}_{2,T} = s \approx k_{2,T}, \quad (3.60)$$

$$\tilde{k}_{3,T} = 1 \approx k_{3,T}, \quad (3.61)$$

$$\tilde{k}_{4,T} = s \approx k_{4,T}, \quad (3.62)$$

$$\tilde{k}_{5,T} = 1 - s^2 \approx k_{5,T}. \quad (3.63)$$

Tai leidžia gauti parametro s įvertinį, perkeliant s į kairiąją lygties $0 = \tilde{h}$ pusę, nes $[Q^{-1}]_{T,T}$ (žr. (3.27)) aproksimuoja s . Be to, tai taip pat sumažina elementariųjų operacijų skaičių rekursiniuose (3.21)–(3.26) dydžių skaičiavimuose ir pagerina jų stabilumą, visa tai pasiekiant už praktiškai nereikšmingą tikslumo praradimą.

3.2.5. Didžiausio tikėtino įvertiniai

Konkrečiai gaunamas toks parametro s įvertinys:

$$\hat{s} = \frac{T}{T+1} \frac{f_T(b, \hat{\Sigma}_{xy}, \hat{\mu}_x, \hat{\mu}_\eta)}{f_T(a, \hat{\Sigma}_{xy}, \hat{\mu}_x, \hat{\mu}_\eta)}.$$

Vertinant parametrus realiuoju laiku įverčių seka gaunama pagal:

$$\hat{s}_{T+1} = \frac{T}{T+1} \frac{f_T(b, \hat{\Sigma}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T})}{f_T(a, \hat{\Sigma}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T})}, \quad (3.64)$$

kur $\hat{\Sigma}_{xy,T}$, $\hat{\mu}_{x,T}$ ir $\hat{\mu}_{\eta,T}$ yra apibrėžti toliau kartu su $\hat{\Sigma}_{y,T}$ ir $\hat{\Sigma}_{xx,T}$.

Įvertiniai $\hat{\Sigma}_{xy}$, $\hat{\mu}_\eta$ ir $\hat{\Sigma}_y$, atitinkamai turintys (3.17)–(3.19) išraiškas, taip pat gali būti išreikšti per rekursiškai apskaičiuojamus (3.21)–(3.26)

dydžius:

$$\begin{aligned} \hat{\hat{\Sigma}}_{xy,T} = & \left(a_{3,T} - a_{2,T} \hat{\mu}'_{x,T} - \hat{\mu}_{x,T} a_{2,T} + \hat{\mu}_{x,T} a_{1,T} \hat{\mu}'_{x,T} \right. \\ & \left. - \frac{(a_{2,T} - \hat{\mu}_{x,T} a_{1,T})(a_{2,T} - \hat{\mu}_{x,T} a_{1,T})'}{a_{1,T}} \right)^{-1} \left(a_{6,T} - \hat{\mu}_{x,T} a_{4,T} \right. \\ & \left. - \frac{a_{4,T}}{a_{1,T}} (a_{2,T} - \hat{\mu}_{x,T} a_{1,T}) \right), \end{aligned} \quad (3.65)$$

$$\hat{\mu}_{\eta,T} = \frac{a_{4,T} - \hat{\hat{\Sigma}}'_{xy,T} (a_{2,T} - \hat{\mu}_{x,T} a_{1,T})}{a_{1,T}}, \quad (3.66)$$

$$\hat{\hat{\Sigma}}_{y,T} = s f_T \left(a, \hat{\hat{\Sigma}}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T} \right). \quad (3.67)$$

O įvertiniai $\hat{\mu}_x$ ir $\hat{\hat{\Sigma}}_{xx}$ (žr. (3.15)–(3.16)) nepriklauso nuo s ir yra nesunkiai apskaičiuojami rekursiškai:

$$\hat{\mu}_{x,T} = \left(1 - \frac{1}{T} \right) \hat{\mu}_{x,T-1} + \frac{[z'_x]_{,T}}{T}, \quad (3.68)$$

$$\hat{\hat{\Sigma}}_{xx,T} = u_T - \hat{\mu}_{x,T} \hat{\mu}'_{x,T}, \quad (3.69)$$

kur $u_T = \frac{[z'_x]_{,1:T} [z_x]_{1:T}}{T}$. Savo ruožtu u_T rekursija yra:

$$u_T = \left(1 - \frac{1}{T} \right) u_{T-1} + \frac{[z'_x]_{,T} [z_x]_{,T}}{T}. \quad (3.70)$$

Pradiniu laiko momentu (t. y. kai $T = 1$) (3.68) ir (3.70) rekursijos inicijuojamos taip:

$$\hat{\mu}_{x,1} = [z'_x]_{,1}, \quad (3.71)$$

$$u_1 = [z'_x]_{,1} [z_x]_{1,}. \quad (3.72)$$

3.2.6. Vertinimo algoritmas

Galiausiai šio poskyrio medžiaga apibendrinama pateikiant sudaryto algoritmo pseudokodą.

3.1 algoritmas: (3.3)–(3.4) lygtimis apibrėžto modelio $n = 1$ atvejo parametrų vertinimas realiuoju laiku pagal DTM.

Ivestis: Stebėjimų imtis $z_t, t = 1, \dots, T$, ir pradinė (spėjamoji) s reikšmė (t. y. $\hat{s}_1$)

Išvestis: Įverčiai $\hat{\mu}_t, \hat{\Sigma}_t, \hat{\Sigma}_{y,t}, t = 1, \dots, T$

- 1: Apskaičiuoti $c_{j,1}, d_{j,1}, j = 1, \dots, 3$, pagal (3.51)–(3.53) ir apskaičiuoti $a_{i,1}, b_{i,1}, i = 1, \dots, 6$, pagal (3.40)–(3.45) su $s = \hat{s}_1$
 - 2: Apskaičiuoti $\hat{\mu}_{x,1}$ ir $\hat{\Sigma}_{xx,1}$ pagal (3.71) bei (3.69) su (3.72) ir apskaičiuoti $\hat{\Sigma}_{xy,1}, \hat{\mu}_{\eta,1}$ ir $\hat{\Sigma}_{y,1}$ pagal (3.65)–(3.67) su $s = \hat{s}_1$
 - 3: Sudaryti $\hat{\mu}_1$ iš $\hat{\mu}_{x,1}$ bei $\hat{\mu}_{\eta,1}$ ir sudaryti $\hat{\Sigma}_1$ iš $\hat{\Sigma}_{xx,1}, \hat{\Sigma}_{xy,1}, \hat{s}_1$ ir $\frac{\hat{\Sigma}_{y,1}}{\hat{s}_1}$ pagal (3.6)–(3.8) su $n = 1$
 - 4: **for** $t \leftarrow 2$ to T **do**
 - 5: Apskaičiuoti $\hat{s}_t$ pagal (3.64)
 - 6: $\hat{s}_t \leftarrow \max(0, \min(\hat{s}_t, 1))$
 - 7: Apskaičiuoti $c_{j,t}, d_{j,t}, j = 1, \dots, 3$, pagal (3.48)–(3.50) ir apskaičiuoti $a_{i,t}, b_{i,t}, i = 1, \dots, 6$, pagal (3.31)–(3.36). Šiuose skaičiavimuose naudoti $\hat{k}_l, l = 1, \dots, 5$, apibrėžtus (3.59)–(3.63) lygtimis (su $s = \hat{s}_t$), vietoje $k_l, l = 1, \dots, 5$, apibrėžtų (3.54)–(3.58) lygtimis
 - 8: Apskaičiuoti $\hat{\mu}_{x,t}$ ir $\hat{\Sigma}_{xx,t}$ pagal (3.68) bei (3.69)–(3.70) ir apskaičiuoti $\hat{\Sigma}_{xy,t}, \hat{\mu}_{\eta,t}$ ir $\hat{\Sigma}_{y,t}$ pagal (3.65)–(3.67) su $s = \hat{s}_t$
 - 9: Sudaryti $\hat{\mu}_t$ iš $\hat{\mu}_{x,t}$ bei $\hat{\mu}_{\eta,t}$ ir sudaryti $\hat{\Sigma}_t$ iš $\hat{\Sigma}_{xx,t}, \hat{\Sigma}_{xy,t}, \hat{s}_t$ ir $\frac{\hat{\Sigma}_{y,t}}{\hat{s}_t}$ pagal (3.6)–(3.8) su $n = 1$
 - 10: **end for**
-

3.3 pastaba. 3.1 algoritmas neatsižvelgia į skaičiavimo problemas, kylančias dėl per mažo stebėjimų skaičiaus (pvz., $\hat{\Sigma}_{xy,1}$ apskaičiavimas apima matricos, sudarytos vien iš nulių, apvertimą). Vis dėlto į šiuos praktinius aspektus tinkamai atsižvelgta faktiškai vykdant 3.1 algoritmą tolesniame poskyryje, skirtame empiriniams rezultatams.

3.4 pastaba. 3.1 algoritmas pradeda parametrų vertinimą nuo vieno stebėjimo momento dydžio imties ir jį tęsia iki imties, susidedančios iš $T > 1$ stebėjimo momentų. Tai leidžia patikrinti algoritmo veikimą tiek nepalankiomis pradinėmis sąlygomis, tiek su iš anksto pateikta duomenų imtimi.

Tačiau praktikoje, prieš pradedant parametrus vertinti realiuoju laiku, įprasta turėti sukaupią netrivialaus dydžio duomenų aibę. Šiuos duomenis galima panaudoti parametrų vertinimo algoritme esant fiksuotam duomenų

rinkiniui, o jo išvestį paskui naudoti kaip realaus laiko algoritmo įvestį. Tokiu būdu išvengiama spėliojimo apie pradines reikšmes, tokias kaip $\hat{s}_1$ 3.1 algoritme.

Be to, praktikoje parametrų įverčiai turi būti atnaujinami kiekvieną kartą pasirodžius naujiems duomenims, nes duomenų generavimo procesas yra nuolatinis. Dėl to vieną kartą realiuoju laiku paleistas algoritmas turėtų atlikti tik vieną iteraciją (o ne T , kaip numatyta 3.1 algoritme).

3.3. Empiriniai rezultatai

Šiame poskyryje atliekamas 3.1 algoritmo empirinis patikrinimas. Šiuo tikslu pateikiami keli skaitiniai eksperimentai, kuriais siekiama patikrinti šiuo algoritmu gaunamų įverčių konvergavimą didėjant stebėjimo momentų skaičiui bei įvertinti jų kokybę, palyginus su empirinio vertinimo teikiamais įverčiais (žr. 3.2 algoritmą), kurie nesunkiai apskaičiuojami tiek fiksuoto duomenų rinkinio, tiek realaus laiko atveju [9].

Visi reikalingi skaičiavimai atlikti naudojant R statistinį paketą [92].

3.2 algoritmas: (3.3)–(3.4) lygtimis apibrėžto modelio $n = 1$ atvejo empirinis parametrų vertinimas realiuoju laiku.

Įvestis: Stebėjimų imtis $z_t, t = 1, \dots, T$

Išvestis: Įverčiai $\hat{\mu}_t, \hat{\Sigma}_t, \hat{\Sigma}_{y,t}, t = 3, \dots, T$

- 1: $\hat{\mu}_3 \leftarrow \frac{z_1 + z_2 + z_3}{3}$
 - 2: $u \leftarrow \frac{z_1 z'_1 + z_2 z'_2 + z_3 z'_3}{2}$
 - 3: $\hat{\Sigma}_3 \leftarrow u - \frac{3}{2} \hat{\mu}_3 \hat{\mu}'_3$
 - 4: $v \leftarrow z_1 z'_2 + z_2 z'_3$
 - 5: $\mu_a \leftarrow \left(\hat{\mu}_3 - \frac{z_1}{3} \right) \frac{3}{2}$
 - 6: $\mu_b \leftarrow \left(\hat{\mu}_3 - \frac{z_3}{3} \right) \frac{3}{2}$
 - 7: $\hat{\Sigma}_{y,3} \leftarrow -[(v - 2\mu_b \mu'_a)]_{m+1, m+1}$
 - 8: **for** $t \leftarrow 4$ **to** T **do**
 - 9: $\hat{\mu}_t \leftarrow \left(1 - \frac{1}{t}\right) \hat{\mu}_{t-1} + \frac{z_t}{t}$
 - 10: $u \leftarrow \frac{t-2}{t-1} u + \frac{z_t z'_t}{t-1}$
 - 11: $\hat{\Sigma}_t \leftarrow u - \frac{t}{t-1} \hat{\mu}_t \hat{\mu}'_t$
 - 12: $v \leftarrow \frac{t-3}{t-2} v + \frac{z_{t-1} z'_t}{t-2}$
 - 13: $\mu_a \leftarrow \left(\hat{\mu}_t - \frac{z_1}{t} \right) \frac{t}{t-1}$
 - 14: $\mu_b \leftarrow \left(\hat{\mu}_t - \frac{z_t}{t} \right) \frac{t}{t-1}$
 - 15: $\hat{\Sigma}_{y,t} \leftarrow -[(v - (t-1)(t-2)\mu_b \mu'_a)]_{m+1, m+1}$
 - 16: **end for**
-

3.5 pastaba. 3.2 algoritmas tiesiogiai seka iš 2.2 algoritmo, atlikus pastarojo skaičiavimus rekursiškai. Tačiau to paties negalima pasakyti apie 3.1 ir 2.1 algoritmų santykį, nes perėjimas nuo (net ir efektyvaus skaičiuojamuoju požiūriu) fiksuoto duomenų rinkinio DTM prie realaus laiko DTM nėra trivialus.

Remiantis Monte Karlo metodu [63], trims skirtingiems modelio parametrų rinkiniams, kurių kiekvienas ženkliną būdingą scenarijų, sugeneruota po 100 duomenų imčių, susidedančių iš $T = 100000$ stebėjimų. Nemažinant bendrumo, pasirinkta: $m = 2$, $k = 1$, $\mu = 0_{3 \times 1}$, $L = 0_{3 \times 1}$, $\text{diag}(\Sigma_x) = 1_{2 \times 1}$ ir $\Sigma_\eta = 1$.

Tada, keičiant Σ_y , gauti trys skirtingi modelio parametrų rinkiniai, atitinkantys skirtingus scenarijus:

1. Mažo išvesties stebėjimo triukšmo scenarijus („Mažas s “): $\Sigma_y = 0,01$, atitinkamas $s = 0,009805$.
2. Vidutinio išvesties stebėjimo triukšmo scenarijus („Vidutinis s “): $\Sigma_y = 1$, atitinkamas $s = 0,381966$.
3. Didelio išvesties stebėjimo triukšmo scenarijus („Didelis s “): $\Sigma_y = 100$, atitinkamas $s = 0,904875$.

3.1 ir 3.2 algoritmai paleisti su kiekviena iš sugeneruotų duomenų imčių (3.1 algoritmas inicijuotas atsitiktinai parinkta s reikšme). Gavus visus įverčius, jie palyginti su faktinėmis – tai yra duomenis generuojančiomis – parametrų reikšmėmis. Kiekvienam scenarijaus, algoritmo, iteracijos (stebėjimų imties dydžio) ir parametrų poaibio deriniui apskaičiuota vidutinė absoliutinė paklaida (VAP) [125].

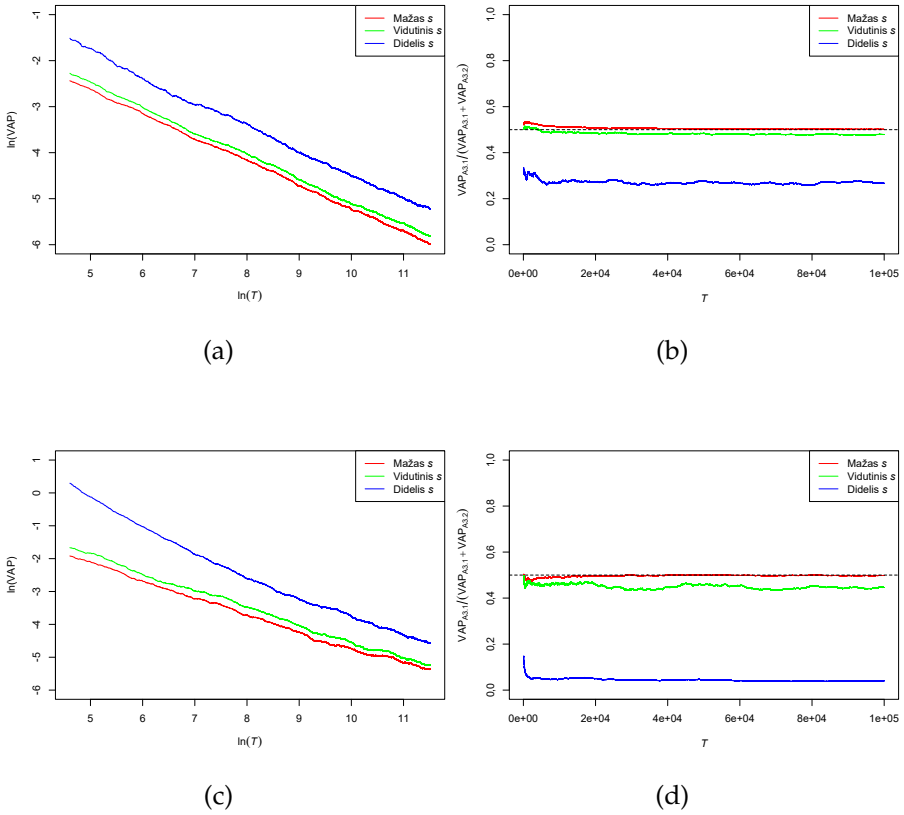
Parametrai suskirstyti į tris poaibius, neutralizuojančius mastelio poveikį:

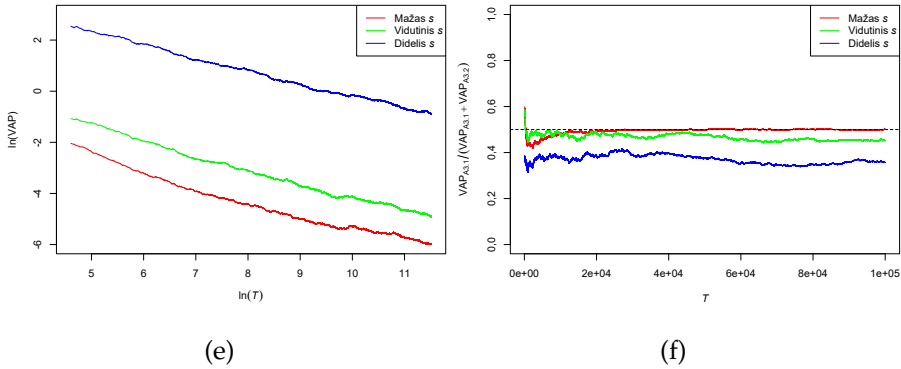
1. Nuliui lygūs elementai (į šį poaibį patenka μ ir neištrižaininiai Σ elementai).
2. Vienetui lygūs elementai (į šį poaibį patenka Σ , kurios elementui $[\Sigma]_{3,3}$ atimta $2\Sigma_y$, ištrižaininiai elementai).
3. Elementas, kurio reikšmė priklauso nuo scenarijaus (t. y. Σ_y).

Gauti rezultatai pavaizduoti 3.1 pav.

3.1a, 3.1c ir 3.1e pav. rodo, kad didėjant stebėjimų imties dydžiui T , VAP artėja prie nulio visų nagrinėtų scenarijų ir parametrų poaibių atveju. Tai empiriškai patvirtina 3.1 algoritmo konvergavimą.

O 3.1b, 3.1d ir 3.1f pav. rodo, kad didėjant išvesties stebėjimo triukšmo santykiniam svoriui modelyje, didėja 3.1 algoritmo pranašumas prieš 3.2 algoritmą įverčių kokybės atžvilgiu. Esant mažam išvesties stebėjimo triukšmui, abu algoritmai yra iš esmės lygiaverčiai; esant vidutiniam triukšmui, tarp jų pradeda ryškėti skirtumas; o esant dideliame triukšmui, šis skirtumas tampa kardinalus.



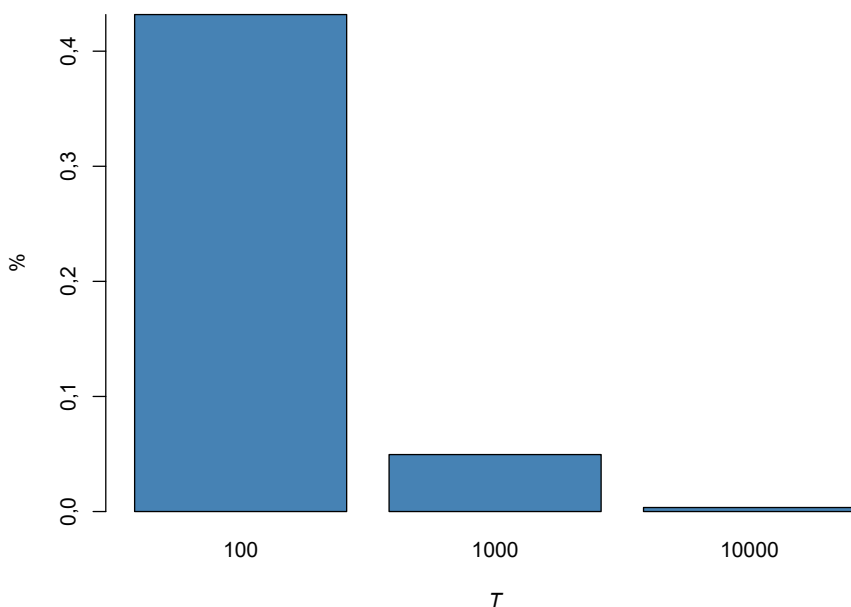


3.1 pav.: Empirinis 3.1 algoritmo (A3.1) patikrinimas (kreivės brėžiamos nuo $T = 100$): (a) konvergavimas parametų poaibio Nr. 1 atžvilgiu; (b) lyginimas su 3.2 algoritmu (A3.2) parametų poaibio Nr. 1 atžvilgiu; (c) konvergavimas parametų poaibio Nr. 2 atžvilgiu; (d) lyginimas su 3.2 algoritmu (A3.2) parametų poaibio Nr. 2 atžvilgiu; (e) konvergavimas parametų poaibio Nr. 3 atžvilgiu; (f) lyginimas su 3.2 algoritmu (A3.2) parametų poaibio Nr. 3 atžvilgiu.

Gautus empirinius rezultatus susiejus su faktu, kad vienai 3.1 algoritmo iteracijai reikalingas skaičiavimo laikas yra apribotas konstanta (t. y. nedidėja priklausomai nuo T), galima teigti, jog DTM pagrindu sukurtas 3.1 algoritmas pagrįstai konkuruoja su 3.2 algoritmu, kuris atlieka etaloninę funkciją įvairių euristinių metodų kontekste. Pastarieji dažnai yra lengvai apskaičiuojami, tačiau tai pasiekama aukojant įverčių kokybę.

Nors 3.1 algoritmas reikalauja daugiau elementariųjų operacijų vienai iteracijai už 3.2 algoritmą, jis ne tik pasižymi analogišku algoritminiu sudėtingumu, tinkamu parametrams vertinti realiuoju laiku, bet ir išsiskiria patikimumu triukšmingoje išvesties stebėjimo aplinkoje.

3.6 pastaba. Gautų empirinių rezultatų analogiškumas antrajame skyriuje pateiktiems rezultatams (konvergavimas didėjant stebėjimų imties dydžiui bei įverčių pranašumo prieš empirinį vertinimą didėjimas, didėjant parametro s reikšmei) ir 3.2 algoritmo teikiamų įverčių lygiavertiškumas 2.2 algoritmo teikiamiems įverčiams (žr. 3.5 pastabą) liudija apie sėkmingą 2.1 algoritmo perdarymą į realaus laiko DTM algoritmą (žr. 3.2 pav.).



3.2 pav.: Skirtumas tarp vidutinių (3.14) tikėtinumo funkcijos reikšmių, apskaičiuotų naudojant atitinkamai 3.1 ir 2.1 algoritmų įverčius (pagal vidutinio išvesties stebėjimo triukšmo scenarijų ir 100 duomenų imčių).

3.4. Diskusija

3.1 algoritmas pakaitomis vertina θ poaibį, kurį sudaro Σ_{xx} , $\tilde{\Sigma}_{xy}$, Σ_y , μ_x ir μ_η , bei θ poaibį, kurį sudaro tik s . Visa tai atliekama tarpininkaujant tinkamai parinktiems dydžiams, kuriems taikomi veiksmingi rekursiniai skaičiavimai, įgalinantys stabilų skaičiavimų krūvį kiekvieną kartą pasirodžius naujiems duomenims (t. y. su kiekviena iteracija).

Nors Σ_{xx} , $\tilde{\Sigma}_{xy}$, Σ_y , μ_x ir μ_η įvertiniai turi išreikštinius pavidalus (t. y. priklauso tik nuo duomenų ir s), s įvertinys yra funkcija nuo s . Tačiau tam tikra prasme 3.1 algoritmas jį taip pat traktuoja išreikštinais, nes vienoje algoritmo iteracijoje s įvertis nėra perskaičiuojamas cikliška iki sukonvergavimo. Tokiu būdu galima nėra maksimizuojamas s įverčio tikslumas, tačiau tai užtikrina, kad per vieną iteraciją atliekamų elementariųjų operacijų skaičius yra fiksuotas.

Iš esmės 3.1 algoritmas susideda iš išreikštinių įvertinių vienai θ daliai, tinkamų neišreikštinių įvertinių likusiai θ daliai ir veiksmingų rekursinių skaičiavimų. Pirmasis aspektas pasiektas dėl sėkmingo mani-

puliavimo atitinkamomis išvestinėmis (žr. antrąjį skyrių), antrasis – dėl tinkamo atitinkamos išvestinės ir iš jos kylančios lygties pertvarkymo, o trečiasis – dėl tridiagonalinių matricų savybių [51] ir tam tikrų specifinių korekcijų.

Papildant antrąjį aspektą, Σ_y pakeista jos įvertiniu tikėtinumo funkcijos išvestinėje pagal s . Tai leido sujungti rekursijas, kylančias iš Σ_y (t. y. tas, kurios apima Q^{-1}) ir išvestinės (t. y. tas, kurios apima Q^{-2}), į vieną visumą (žr. b (3.21)–(3.26) lygtyse), taip sukuriant gerai subalansuotą atsvarą tik su Q^{-1} susijusioms rekursijoms (žr. a (3.21)–(3.26) lygtyse ir f (3.27) lygtyje). Tokiu būdu pavyko sukonstruoti konverguojantį s įvertinį (pažymėtina, kad nors galima sukonstruoti daug skirtingų s įvertinių, konverguojančio įvertinio radimas nėra akivaizdus).

Gautas s įvertinys gali būti nagrinėjamas naudoto modelio daugiamatės išvesties versijoje (t. y. (3.3)–(3.4) lygtimis apibrėžtame modelyje), nes jo tikėtinumo funkcijos parametras s susideda iš n elementų (žr. (3.12)), kurių kiekvienas atitinka atskirą narį, analogišką $n = 1$ atvejui (žr. (3.13)). Tačiau pagrindinis šios tikėtinumo funkcijos savybės apribojimas pereinant nuo $n = 1$ prie $n > 1$ slypi likusiuose parametruose, nes nėra aišku, ar galima sudaryti išreikštinius Σ_{xx} , $\tilde{\Sigma}_{xy}$, $\tilde{\Sigma}_y$, μ_x ir μ_η įvertinius daugiamatės išvesties modelio atveju. Net jei atsakymas būtų teigiamas, gautoms sudėtingesnėms išraiškoms gali būti nebetyktytas analogiškas Σ_y pakeitimo jo įverčiu manevras, todėl gali tekti koreguoti s įvertinį (t. y. jį sudaryti naudojant tik tikėtinumo funkcijos išvestinės pagal s pertvarkymus). Jei atsakymas būtų neigiamas, šiems parametrams taip pat reiktų sudaryti atitinkamus iteracinius įvertinius (galbūt taip pat atliekant vieną iteraciją su kiekvienu naujų duomenų pasirodymu, kaip tai daroma vertinant parametą s vienmatės išvesties modelio atveju), kurie galėtų būti sudaromi remiantis įprastais metodais, tokiais kaip, pavyzdžiui, Niutono metodas [88]. Be to, remiantis dabartiniu problemos matymu, daugiamatės išvesties modelio atveju s apribojimams gali prireikti taikyti optimizavimo metodą su logaritminiu barjeru, kuris sėkmingai pritaikytas sprendžiant tiesinio optimizavimo problemas vidinio taško algoritmais [126].

3.1 spėjimas. *Egzistuoja efektyvus algoritmas, skirtas (3.3)–(3.4) lygtimis apibrėžto modelio parametrams vertinti realiuoju laiku. Kaip įvestį šis algoritmas priima stebėjimų duomenis z_t , $t = 1, 2, \dots$, (laikomus atsitiktinių dydžių Z_t , $t = 1, 2, \dots$, realizacijomis) ir, jei reikia, tam tikras pradines būtinų parametru*

reikšmes. Kaip išvesti šis algoritmas pateikia įverčius $\hat{\mu}_t = \hat{\mu}_t(z_1, \dots, z_t)$, $\hat{\Sigma}_t = \hat{\Sigma}_t(z_1, \dots, z_t)$ ir $\hat{\Sigma}_{y,t} = \hat{\Sigma}_{y,t}(z_1, \dots, z_t)$, kurie bendrai vektorizuojami ir žymimi $\hat{\theta}_t$, $t = 1, 2, \dots$ (atitinkamai tikrieji (t. y. duomenis generuojantys) parametrai μ , Σ ir Σ_y taip pat vektorizuojami ir žymimi θ). Šis algoritmas pasižymi šiomis savybėmis:

1. $\hat{\theta}_t(Z_1, \dots, Z_t) \xrightarrow{P} \theta$, kai $t \rightarrow \infty$.
2. $\sqrt{t} \left(\hat{\theta}_t(Z_1, \dots, Z_t) - \theta \right) \xrightarrow{d} \mathcal{N}(0, \mathcal{I}^{-1}(\theta))$, kai $t \rightarrow \infty$, kur $\mathcal{I}(\theta)$ yra Fišerio informacijos matrica [105].
3. Kiekvienoje iteracijoje (t. y. pereinant nuo t prie $t + 1$) atliekama $\mathcal{O}(1)$ elementariųjų operacijų.

Remiantis požiūriu „iš apačios į viršų“, algoritmo, tenkinančio 3.1 spėjimą, sudarymas gali būti laikomas natūraliu kitu žingsniu siekiant efektyvaus DSLM parametrų vertinimo realiuoju laiku. Vis dėlto svarbu pažymėti, kad bendrasis DSLM, kuris turėtų būti vienas iš galutinių tokių laipsniškų patobulinimų tikslų, yra gerokai sudėtingesnis už (3.1)–(3.3) lygtimis apibrėžtą modelį, nes apima autoregresinius narius (t. y. ilgėsnę atmintį) ir lanksčius regresinius koeficientus [20]. Net jei tokio bendro modelio atveju pavyktų realizuoti iniciatyvinę PVT dalį, kuria praretinama pasirinkto modelio kovariacijų matrica, kovariacijų matricos blokų ištirizainiškumas būtų platesnis nei šiame skyriuje nagrinėto modelio atveju (pastarajame yra diagonaliniai ir tridiagonaliniai blokai). Nors egzistuoja tam tikri uždaros formos rezultatai, skirti daugiadiagonalinėms matricoms apversti [60], šias matricas apskritai gerokai sudėtingiau apdoroti nei tridiagonalines, tad šio skyriaus parametrų vertinimo realiuoju laiku supratimo išplėtimas bendresniems DSLM neišvengiamai susijęs su iššūkiais.

3.5. Skyriaus išvados

- PVT realizacijos rezultatai gali būti panaudoti sudarant efektyvius realaus laiko parametrų vertinimo algoritmus, tačiau be jų taip pat reikalingi papildomi specifiniai veiksmai (efektyvių realaus laiko parametrų vertinimo algoritmų sudarymas kelia daugiau iššūkių nei atitinkamų algoritmų, skirtų parametrų vertinti esant fiksuotam duomenų rinkiniui).

- DTM ir rekursijos pagrindu sudarytas vienmatės išvesties grynujų laiko eilučių duomenų KSLM su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis parametrų vertinimo realiuoju laiku algoritmas pasižymi konkurencinga įverčių kokybe triukšmingų išvesties stebėjimų atvejais, o jo sudarymo principai gali būti pritaikyti kuriant atitinkamus algoritmus bendresniems modeliams.

4. PRAKTINIAI ASPEKTAI

Nors tokių modelio parametrų vertinimo algoritmu, kaip 2.1 ir 3.1, sudarymas reikalauja specifinių pastangų (kurių poreikis auga kartu su modelio bendrumu), jie yra nesunkiai įvykdomi programiškai (žr. skyrių „Priedas“), todėl jų panaudojimas yra techniškai nesudėtingas. Kita vertus, tokių algoritmų įvestis yra tam tikru modeliu aiškinami duomenys (duomenys sudaro įvestį į algoritmą, kuris yra skirtas tam tikro modelio parametrams įvertinti), tačiau šie komponentai nebūtinai yra akivaizdūs praktinių taikymų atveju. Kitaip sakant, modelio parametrų vertinimas, nors ir būtinas bei keliantis savų iššūkių, yra tik viso modeliavimo proceso dalis.

Šiame skyriuje, parengtame [95, 96] straipsnių pagrindu, praktiniams taikymams aktualūs aspektai nagrinėjami COVID-19 pandemijos bei socialinio kapitalo modeliavimo kontekstuose.

4.1. COVID-19 pandemijos modeliavimas

COVID-19 pandemija išlieka aktuali tyrimų sritis net ir jai nurimus [2, 40, 72, 94]. Oficialiai surinktų susirgimų, mirčių ir kitų susijusių statistinių duomenų analizė gali padėti gauti reikalingų įžvalgų, svarbių prognozuojant kitų pandemijų eigą ir kuriant strategijas joms suvaldyti [30, 86]. Tačiau egzistuoja tam tikras atotrūkis tarp oficialiosios COVID-19 pandemijos statistikos ir modelių, leidžiančių nuodugnai išanalizuoti jos suvaldymo priemonių, tokių kaip karantinas [7], medicininės kaukės [102], galimybių pasas [119] ar masinė vakcinacija [56], veiksmingumą [26]. Todėl kyla abejonių, ar turimi duomenys leidžia daryti aukštos kokybės išvadas [65]. Šis atotrūkis nagrinėjamas šiame poskyryje siekiant pademonstruoti galimą darbo su esamais duomenimis būdą, kuris bent iš dalies neutralizuoja jo ribojantį poveikį, ir atkreipti dėmesį į gilumines analizes leidžiantį duomenų rinkimą bei panaudojimą.

Motyvacija nagrinėti šią iš pažiūros neprobleminę temą kyla iš redakciniuose laiškuose kilusios diskusijos apie COVID-19 galimybių paso įtaką viruso plitimui Lietuvoje [55, 108, 109]. Diskusiją pradėjusiame laiške šis poveikis įvertintas remiantis aprašomosios statistikos principais ir metodais bei tam tikromis bendrosios nuovokos prielaidomis. Tačiau kita diskusijos šalis atkreipė dėmesį į tokio požiūrio trūkumus ir išreiškė nuomonę, kad gautos išvados gali būti klaidingos dėl problemų, susijusių tiek su metodu, tiek su duomenimis.

Iš tiesų tirdami su pandemijomis susijusių procesų ypatybes, tyrėjai dažnai naudoja aprašomąją statistiką, kuri leidžia apibendrinti arba apibūdinti duomenų rinkinio charakteristikas, apskaičiuojant centrinės tendencijos, kintamumo ir dažninio skirstinio rodiklius [33]. Tačiau tokiais metodais galima pasiekti tik tam tikrus iliustracinius tikslus, turinčius ribotas išvadų galimybes, o ne atlikti išsamią ir patikimą dominančių reiškinių analizę, nes bet koks duomenų agregavimas plačiąja prasme reiškia informacijos praradimą, kuris savo ruožtu mažina modeliavimo jautrumą dažnai svarbioms detalėms ir taip gali kenkti rezultatų kokybei [98].

Norint modeliuoti žemesniame lygmenyje (o tai galima padaryti tyrinėjamus objektus suskaidant į mažesnius conceptualius vienetus, kol išryškėja fundamentalūs veikimo mechanizmai [103]) epidemiologijoje dažnai naudojami segmentiniai modeliai (modelio aprūpinimas detaliais duomenimis sudaro dar vieną sąlygą modeliuoti žemame lygmenyje). Iš esmės segmentiniai modeliai yra labai bendra modeliavimo technika. Šioje modeliavimo sistemoje populiacija priskiriama segmentams, žymimiems tokiomis etiketėmis kaip S, I, R ir D (S yra angliško žodžio *Susceptible* trumpinys, žymintis galinčią užsikrėsti populiacijos dalį; I – *Infected*, žymi užsikrėtusiuosius; R – *Recovered*, žymi pasveikusiuosius; D – *Deceased*, žymi mirusiuosius) [32]. Populiaciją sudarantiems subjektams leidžiama judėti iš vieno segmento į kitą pagal tam tikras taisykles, kurias nustato modelio struktūra, kuri savo ruožtu yra modeliavimo prielaidų įgyvendinimas [84]. Segmentiniai modeliai dažnai aprašomi naudojant įprastas (deterministines) diferencialines lygtis, tačiau taip pat gali būti įgyvendinami stochastiniu (atsitiktiniu) pagrindu, kuris yra realesnis, bet sudėtingiau analizuojamas [17]. Šio tipo modeliai epidemiologijoje naudojami siekiant prognozuoti užkrato plitimą, užsikrėtusiųjų skaičių, pandemijos trukmę bei įvertinti įvairius epidemiologinius parametrus, tokius kaip reprodukcijos skaičius. Tokiais modeliais taip pat galima aiškinti, kaip įvairios valdžios institucijų nustatytos visuomenės sveikatos intervencijos gali paveikti pandemijos eigą [16, 133].

Iš esmės epidemiologiniai segmentiniai modeliai yra dinaminiai modeliai, dažniausiai išreiškiami bendrai netiesinių lygčių rinkiniu [115]. Norint tokius modelius pritaikyti praktinių procesų analizei, būtina, remiantis turimais faktiniais duomenimis, įvertinti nežinomus modelio

parametrus. Parametrų įverčių tikslumas priklauso nuo taikomo vertinimo metodo ir nuo prieinamų duomenų adekvatumo, kokybės bei gausos [22, 54].

Metodo atžvilgiu parametrų vertinimas idealiu atveju turėtų būti atliekamas taikant fundamentalius tradicinius metodus, tokius kaip didžiausio tikėtinumų metodas (DTM) [80], pasižyminčius (asimptotiškai) optimaliomis įvertinių savybėmis ir todėl leidžiančius daryti patikimiausias įmanomas išvadas. Kita vertus, DTM dažnai būna sunkiai įgyvendinamas (ypač dirbant su kompleksiniais modeliais ir dideliais duomenų kiekiais), todėl tyrėjai neretai pasitelkia įvairias nesunkiai apskaičiuojamas, tačiau įvertinių kokybės nemaksimizuojančias euristicines alternatyvas [50]. Taigi tenka pabrėžti, kad efektyvus parametrų vertinimas neišvengiamai reikalauja tam skirtų pastangų, aptartų ankstesniuose šio darbo skyriuose.

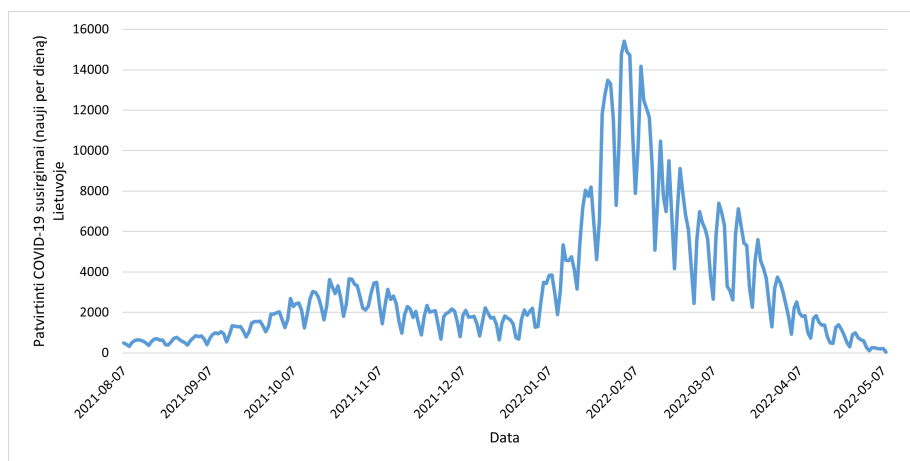
Duomenų atžvilgiu parametrų vertinimas idealiu atveju reikalauja duomenų, apibūdinančių visus segmentinį modelį sudarančius segmentus. Tačiau oficialiai surinkti ir pateikiami COVID-19 duomenys šiuo požiūriu yra sąlyginai riboti, o tai apsunkina adekvatų ir išsamų segmentinių modelių taikymą pandemijai modeliuoti. Taip yra todėl, kad arba paprasčiausiai nėra (nuo prielaidų nepriklausomų) duomenų apie specifinius modeliuotinus segmentus, arba iš pažiūros tinkami duomenys pasirodo turintys įvairių problemų, neigiamai veikiančių jų tikslumą modeliavimo tikslų atžvilgiu [54, 65, 76]. Tam tikrus duomenų apribojimus galima bandyti kompensuoti naudojant sudėtingesnius modelius – tai yra su daugiau lygčių ir parametrų, – kurie iš apribojimų atsižvelgia pagal savo dizainą. Vis dėlto didesnis modelio sudėtingumas gali pernelyg apsunkinti jo sprendžiamumą ir pritaikomumą, taip galiausiai sukeldamas panašias problemas kaip ir netinkami ar nepakankami duomenys.

4.1 pastaba. Šiame poskyryje oficialūs (prieinami) COVID-19 duomenys daugiausia suprantami kaip duomenys, pateikiami svetainės *OurWorldInData.org* COVID-19 teminiame puslapyje [74].

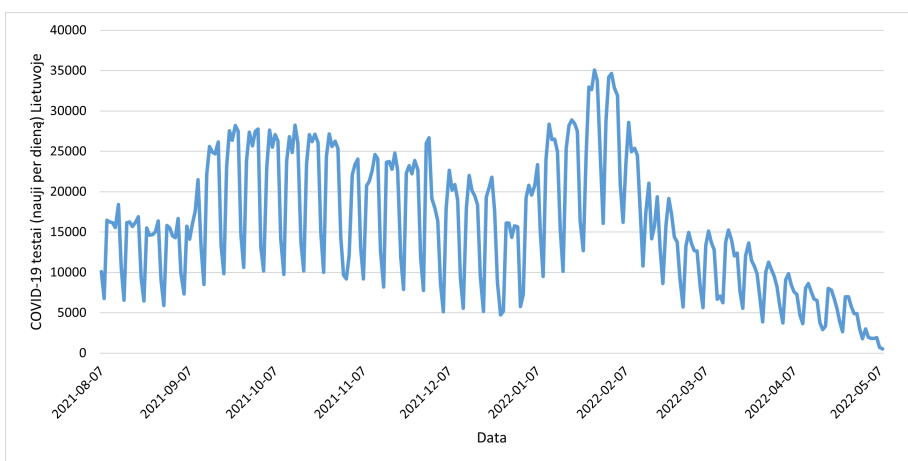
Prieinamų duomenų analizė, atliekama per pirmiau aprašyto (dieninio detalumo, kuris yra maksimalus prieinamų duomenų atveju) iš keturių segmentų sudaryto SIRD modelio prizmę, atskleidžia su šiais duomenimis susijusias problemas. Čia I segmentą labiausiai atitinka

patvirtinti COVID-19 susirgimai (nauji per dieną) [121] (žr. 4.1 pav.). Tačiau būtina pabrėžti, kad šiems duomenims būdingos kelios problemos:

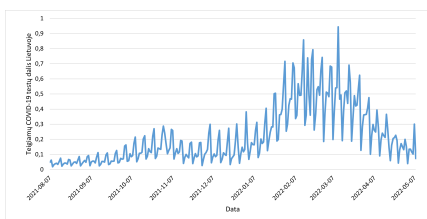
1. Jie yra neatsiejami nuo testavimo apimčių, kurias atspindi *COVID-19 testai (nauji per dieną)* [82], o šios reikšmingai varijuoja dienų lygmeniu (žr. 4.2 pav.).
2. Testuojama tik dalis populiacijos, todėl *patvirtinti COVID-19 susirgimai (nauji per dieną)* [121] neatspindi visos populiacijos.
3. Testavimas gali duoti klaidingų rezultatų – asmuo, kurio testas teigiamas, gali būti neužsikrėtęs, arba atvirkščiai.
4. Laiko tarpas nuo testų atlikimo iki jų rezultatų paskelbimo yra kintamas ir iš esmės nežinomas (4.3 pav. pavaizduota *teigiamų COVID-19 testų dalis* [83], apskaičiuota *patvirtintų COVID-19 susirgimų (naujų per dieną)* [121] duomenis paslinkus skirtingu dienų skaičiumi *COVID-19 testų (naujų per dieną)* [82] atžvilgiu – galima pastebėti, kad duomenys santykinai praranda periodiškumą, kai šis poslinkis yra viena diena, tačiau net ir tokiu atveju jis išlieka reikšmingas).



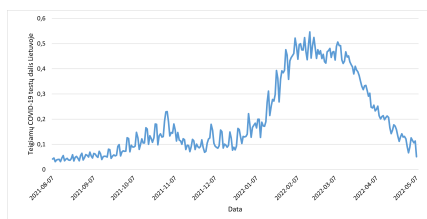
4.1 pav.: Patvirtinti COVID-19 susirgimai (nauji per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.



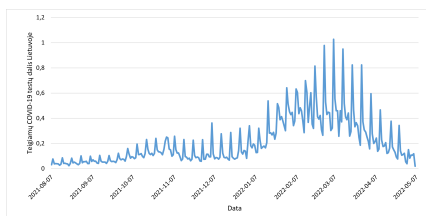
4.2 pav.: COVID-19 testai (nauji per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.



(a)



(b)

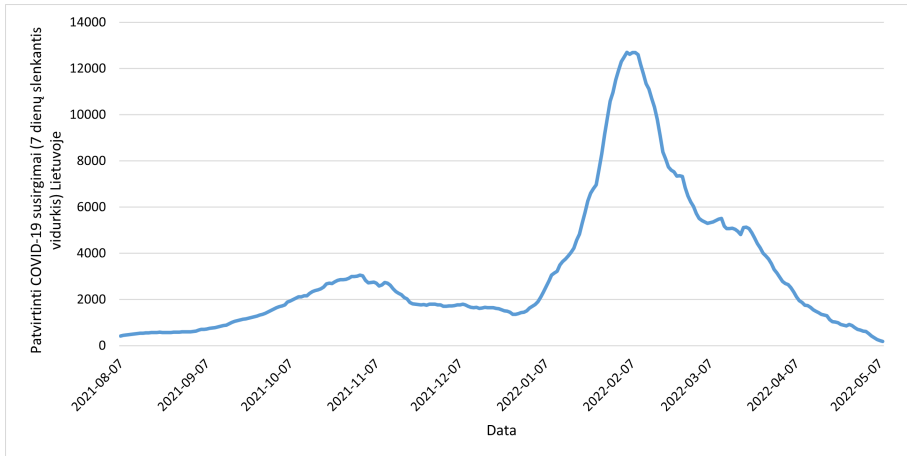


(c)

4.3 pav.: Teigiamų COVID-19 testų dalis Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.: (a) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 0 dienų į ateitį COVID-19 testų (naujų per dieną) atžvilgiu; (b) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 1 dieną į ateitį COVID-19 testų (naujų per dieną) atžvilgiu; (c) patvirtinti COVID-19 susirgimai (nauji per dieną), paslinkti 2 dienas į ateitį COVID-19 testų (naujų per dieną) atžvilgiu.

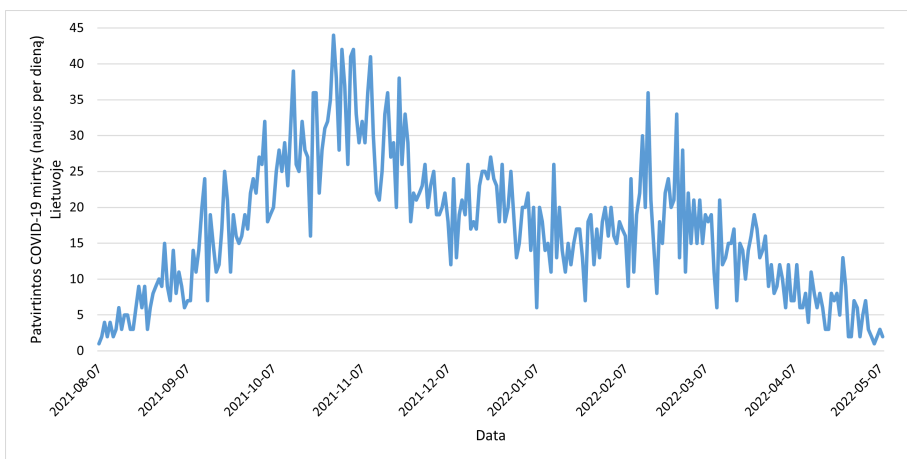
Šios problemos reikšmingai komplikuoja faktinio dieninio populiacijos srauto, patenkančio į I segmentą, nustatymą. Siekiant išvengti kai kurių problemų (pvz., 1), dažnai naudojami agreguoti duomenys (žr. 4.4 pav.). Tačiau, nepaisant anksčiau minėtų duomenų agregavimo trūkumų, toks sprendimas taip pat apsunkina segmentinių modelių taikymą ir techniniu požiūriu, nes taip įvedamos duomenų priklausomybės [39].

Negana to, patekimo į R segmentą identifikavimas kelia panašų iššūkį, nes nėra susiformavusios bendros praktikos testuoti dėl pasveikimo fakto patvirtinimo. Šiuo tikslu galima naudoti nebent tam tikrą įvertį, kylantį iš prielaidos, kad dabartinis užsikrėtusių asmenų skaičius atitinka bendrą per paskutines n dienas patvirtintų susirgimų skaičių (Valstybinė duomenų agentūra naudoja $n = 20$ [110]).



4.4 pav.: Patvirtinti COVID-19 susirgimai (7 dienų slenkantis vidurkis) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.

Tačiau situacija yra palankesnė nagrinėjant perėjimą į D segmentą, nes ji sąlyginai gerai atspindi *patvirtintos COVID-19 mirtys (naujos per dieną)* [122] (žr. 4.5 pav.). Taip yra todėl, kad šiuos duomenis nesunku užfiksuoti tikrąją mirties dieną pilna apimtimi. Pagrindinis jų trūkumas – potencialiai neteisingas mirties priežasties priskyrimas, analogiškas *patvirtintų COVID-19 susirgimų (naujų per dieną)* [121] trečiajam problemiam punktui.



4.5 pav.: Patvirtintos COVID-19 mirtys (naujos per dieną) Lietuvoje laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.

Jei trūksta geros kokybės duomenų apie kiekvieną iš modeliujamų segmentų, segmentinį modeliavimą gali būti prasminga atlikti modelį išreiškiant tik per tuos segmentus, kuriems būdingi patikimiausi duomenys. Šiame poskyryje teigiama, kad SIRD modelio atveju toks segmentas yra D, ir pasisakoma už DTM naudojimą SIRD modelio, pagrįsto tik COVID-19 mirtimis, į kurių trūkumą atsižvelgiama į modelį įtraukiant stebėjimo triukšmą, parametrus įvertinti. Paverstas konkrečiu modeliavimo sprendimu, šis požiūris taikomas faktiniams Lietuvos COVID-19 mirčių duomenims, analizuojant galimybių paso poveikį delta ir omikrono atmainų plitimo atvejais. Galiausiai aptariami galimi duomenų ir modelio patobulinimai.

4.2 pastaba. *Tenka aiškiai pripažinti COVID-19 mirčių duomenimis pagrįsto modeliavimo ribotumą, nes nėra standartinių ir patikimų kriterijų mirties priežasčiai nustatyti (skrodimai atlikti tik mažumai asmenų, kurie tariamai mirė nuo COVID-19). Nepaisant to, šie duomenys yra patikimesni nei COVID-19 susirgimų duomenys, plačiai naudojami kituose darbuose [43, 66, 108, 131], nes pastarieji priklauso nuo testavimo apimtys, strategijos ir testų kokybės.*

4.1.1. Duomenys ir modelis

Nagrinėjamas diskretaus laiko deterministinis SIRD modelis. Šiame modelyje judėjimas tarp segmentų apibrėžiamas šiomis lygtimis:

$$S_{t+1} = S_t - aI_tS_t, \quad (4.1)$$

$$I_{t+1} = I_t (1 + aS_t - b - c), \quad (4.2)$$

$$R_{t+1} = R_t + bI_t, \quad (4.3)$$

$$D_{t+1} = D_t + cI_t, \quad (4.4)$$

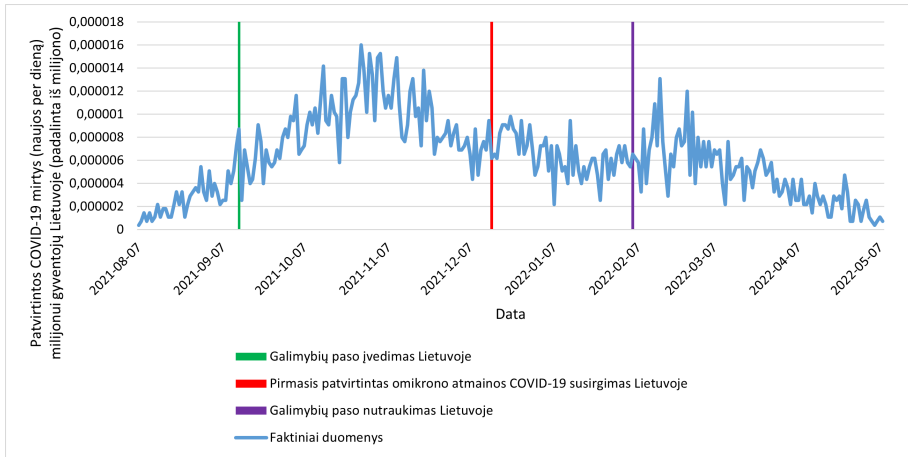
kur $t = 0, \dots, T - 1$, $S_0 = 1 - I_0$, $I_0 = \mu$, $R_0 = 0$, $D_0 = 0$. Koeficientai a , b ir c atitinkamai žymi perėjimo tikimybes iš S į I , iš I į R ir iš I į D , o μ – užkrėstos populiacijos dalį pradinio laiko momentu. Pastaroji reikšmė turi būti didesnė už nulį, kad modeliuojama infekcijos dinamika galėtų pasireikšti.

Norint panaudoti šį modelį, būtina įvertinti jo parametrus (t. y. a , b , c ir μ) pagal duomenis, apibūdinančius segmentus S , I , R ir D (idealiu atveju). Šiuo tikslu vienas iš tinkamiausių ir labiausiai patikimų metodų yra DTM, kuriam būdingos (asimptotiškai) optimalios įvertinių savybės. Svarbu pažymėti, kad pasirinktos modelio lygtys gali būti išreiškiamos per likusias, taip suteikiant modeliui tinkamą formą darbui tik su pasirinktais segmentais (duomenimis).

Čia naudojami (maksimalaus) dieninio detalumo COVID-19 mirčių duomenys (t. y. diskretaus laiko indeksas t atitinka dienas), siekiant įvertinti modelio parametrus taikant DTM. Su modeliu susijęs dydis, atitinkantis *patvirtintų COVID-19 mirčių (naujų per dieną)* [122] duomenis, apibrėžiamas taip (čia jis išreiškiamas vienetais milijonui gyventojų ir padalijamas iš milijono):

$$\Delta D_t = D_{t+1} - D_t. \quad (4.5)$$

Lietuvos ΔD laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d. yra pavaizduotas 4.6 pav.



4.6 pav.: Patvirtintos COVID-19 mirtys (naujos per dieną) milijonui gyventojų Lietuvoje (padalinta iš milijono) laikotarpiu nuo 2021 m. rugpjūčio 7 d. iki 2022 m. gegužės 8 d.

4.6 pav. pavaizduotas laikotarpis iš esmės apima delta ir omikrono atmainų plitimą Lietuvoje. Vidurinis vertikalus brūkšnis žymi datą, kai aptikti pirmieji omikrono atvejai – 2021 m. gruodžio 15 d. Kairysis brūkšnis žymi COVID-19 galimybių paso įvedimo datą (2021 m. rugsėjo 13 d.), o dešinysis – galiojimo pabaigą (2022 m. vasario 5 d.). Per visą nurodytą laikotarpį ΔD nėra lygus nuliui, o pirmosiomis į jį nepatenkančiomis dienomis abiejose kraštinėse (t. y. 2021 m. rugpjūčio 6 d. ir 2022 m. gegužės 9 d.) ΔD yra lygus nuliui. Šis laikotarpis nagrinėjamas 4.1.2 skirsnyje, kuriame analizuojamas COVID-19 galimybių paso poveikis pandemijos plitimui Lietuvoje.

Įvesto (4.5) dydžio atžvilgiu (4.1)–(4.4) lygtimis apibūdinamas modelis yra išreiškiamas taip:

$$\Delta D_0 = \mu c, \quad (4.6)$$

$$\Delta D_1 = \Delta D_0 \left(1 + a \left(1 - \frac{\Delta D_0}{c} \right) - b - c \right), \quad (4.7)$$

$$\Delta D_{t+1} = \Delta D_t \left(1 + a \left(1 - \frac{\Delta D_t}{c} - \left(\frac{b}{c} - 1 \right) \sum_{j=0}^{t-1} \Delta D_j \right) - b - c \right), \quad (4.8)$$

kur $t = 1, \dots, T - 2$.

Atsižvelgiant į ΔD trūkumą, aptartą šio poskyrio įžanginėje dalyje, šis dydis papildomas normaliuoju, multiplikatyvinio pobūdžio stebėjimo triukšmu (normalusis skirstinys yra plačiai taikomas paklaidoms modeliuoti [134, 135]):

$$Z_t = \Delta D_i (1 + \sigma \epsilon_t), \quad (4.9)$$

kur $t = 0, \dots, T-1$, ϵ_t yra stebėjimo neapibrėžtumai, modeliuojami kaip nepriklausomos standartinio normaliojo atsitiktinio dydžio (t. y. turinčio nulinį vidurkį ir vienetinę dispersiją) realizacijos, o σ yra atitinkamas standartinis nuokrypis (t. y. stebėjimo triukšmo stiprumas).

4.3 pastaba. (4.6)–(4.8) lygtimis apibrėžtas modelis nėra pajėgus paaiškinti tokių duomenų, kaip pavaizduotieji 4.6 pav., nes pagal jį gaunamos kreivės yra glotnios, o duomenys pasižymi dideliu kintamumu. Tačiau (4.9) lygtis būtent ir įveda šį kintamumą į modelį, taip suderindama jį su duomenų keliais reikalavimais. Reikia pažymėti, kad nors šis priedas čia vadinamas stebėjimo triukšmu (remiantis būsenų erdvės modeliavimo dizainu [5]), jis gali būti interpretuojamas ir plačiau – kaip bendresnis triukšmas, apimantis nežinomus bei nemodeliuojamus veiksnius.

(4.6)–(4.9) lygtimis apibrėžtas modelis sudaro siūlomo metodo matematinį pagrindą. Kitas (ir paskutinis) žingsnis – tai atitinkamų parametrų įvertinimas, o tai galima efektyviai atlikti remiantis DTM ir pasitelkiant specifinius palaipsninio vertinimo technikos (PVT) aspektus bei standartinę funkcijos optimizavimo algoritmą. Atsižvelgiant į tai, kad šių operacijų matematinės detalės nėra esminės šio poskyrio pagrindinei minčiai, toliau jos nėra detalizuojamos.

Siūlomas metodas leidžia taikyti SIRD modelį, suderinamą su faktine oficialių COVID-19 duomenų situacija. Tačiau, kaip ir patys duomenys, šis metodas taip pat nėra idealus, nes segmentinio modelio išreiškimas per tam tikrą segmentų poaibį sudaro prielaidas galimam parametrų peridentifikavimui. Pavyzdžiui, SIRD modelio atveju stebimas mirčių skaičius gali būti paaiškinamas mažesniu užsikrėtimo rodikliu a , bet didesniu mirtingumu c , arba atvirkščiai. Kita vertus, ar toks neapibrėžtumas turi įtakos konkrečiam (4.6)–(4.9) lygtimis apibrėžtam modeliui, išlieka tikėtinumo funkcijos analizės klausimas, kuris nėra šio poskyrio nagrinėjimo objektas.

Reikia pabrėžti, kad analitinė (4.1)–(4.4) lygtimis apibrėžto modelio analizė rodo, jog epidemijos pradžioje infekcijų skaičius didėja eksponentiškai. Iš pradžių užsikrėtusių asmenų skaičius gali sudaryti tik nedidelę gyventojų dalį. Tačiau modelio prielaida, kad infekcijų skaičius yra proporcingas nuolat mažėjančiam susirgti galinčių asmenų skaičiui, reiškia, jog infekuotųjų skaičius (taigi ir čia dominančių mirusiųjų skaičius) natūraliai pasiekia piką, po kurio seka nuosmukis.

Taigi norint bendrai (t. y. su viena tikėtinumo funkcija) nagrinėti laikotarpius, apimančius daugiau kaip vieną viruso bangą (t. y. su daugiau nei vienu lokaliu maksimumu), reikia naudoti laike kintančius modelio parametrus (tačiau jų įgyvendinimas nėra techniškai sudėtingas). Be to, kintantys parametrai yra reikalingi ir modeliuojant įvairius reiškinius (pvz., galimybių paso įvedimo poveikį) net vienos viruso bangos apimtyje.

4.1.2. Rezultatai

Analizuojami šie reiškiniai Lietuvos atveju:

1. COVID-19 galimybių paso įvedimo įtaka delta bangos plitimui.
2. COVID-19 galimybių paso nutraukimo įtaka omikrono bangos plitimui.

Abu reiškiniai modeliuojami nepriklausomai vienas nuo kito, naudojant atskirus duomenų rinkinius ir tikėtinumo funkcijas (alternatyviai galima naudoti bendrą duomenų rinkinį ir tikėtinumo funkciją, tačiau tai lemia techninio įgyvendinimo, o ne rezultatų skirtumus).

4.4 pastaba. *Pažymėtina, kad analizuojant tokius reiškinius, kaip išvardytieji pirmiau, nėra aktuali galima peridentifikavimo problema, aprašyta 4.1.1 skirsnyje. Taip yra todėl, kad tokiais atvejais aktualūs tik parametro a (t. y. užsikrėtimo intensyvumo) pokyčiai, o visi kiti parametrai išlieka pastovūs visame modeliavimo horizonte.*

Pirmojo reiškinio analizei pasirinktas laikotarpis nuo naudojamų duomenų pradžios (t. y. 2021 m. rugpjūčio 7 d.) iki 2021 m. gruodžio 27 d. Čia daroma prielaida, kad omikrono atmainos plitimas prasidėjo 2021 m. gruodžio 28 d. Nors pirmieji susirgimai oficialiai užfiksuoti 2021 m. gruodžio 15 d. (ir tikėtina, kad omikronas cirkuliavo dar anksčiau), ši atmaina pradėjo užgožti delta bangą tik vėliau. Atsižvelgiant į

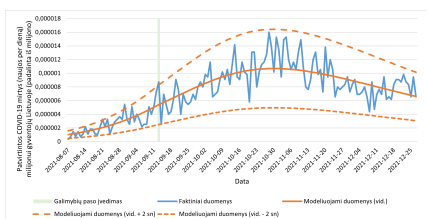
tai, kad tuo metu 7 dienų slenkantis patvirtintų COVID-19 susirgimų vidurkis buvo žemesnis nei 500 – tai rodė, kad delta banga traukiasi, o omikronas dar neįgavo reikšmingo masto, – laikoma, kad omikrono pradžia sutampa su atsinaujinusiu COVID-19 susirgimų augimu, kai šis vidurkis viršijo 500 ribą. Iš esmės omikrono pasireiškimo momentas gali būti modeliuojamas kaip parametras, kurį taip pat galima įvertinti pagal duomenis [10], tačiau dėl paprastumo šis momentas čia traktuojamas kaip iš anksto apibrėžtas.

Šis pasirinktas laikotarpis suskirstytas į dvi dalis: laikotarpį be COVID-19 galimybių paso (t. y. iki 2021 m. rugsėjo 12 d.) ir laikotarpį su COVID-19 galimybių pasu (t. y. nuo 2021 m. rugsėjo 13 d.). Kadangi COVID-19 galimybių pasas iš esmės gali turėti įtakos tik viruso plitimui (modeliuojamam parametru a), o ne pasveikimui ar mirtingumui (atitinkamai modeliuojamiems parametrais b ir c), šiuo atveju vienintelis kintantis modelio parametras buvo a . Jo kitimui pritaikyta stačiakampė tendencija, tai yra $a = a_1$ iki 2021 m. rugsėjo 12 d. ir $a = a_2$ nuo 2021 m. rugsėjo 13 d. (visi kiti modelio parametrai laikui bėgant buvo laikomi pastoviais), o tai atitinka prielaidą, kad COVID-19 galimybių paso poveikis buvo momentinis. Apskritai tokie modeliavimo pasirinkimai atitinka prielaidą, kad delta bangos charakteristikos bei bet kokie reikšmingi išoriniai veiksniai (išskyrus COVID-19 galimybių pasą) analizuojamu laikotarpiu išliko pastovūs.

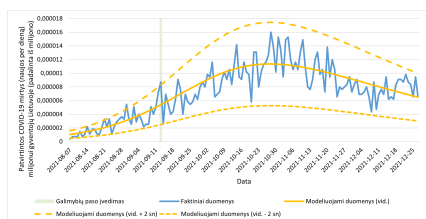
4.5 pastaba. *Kadangi analizuojant pandemijos kontrolės priemonę, susijusią su užsikrėtimų duomenimis, naudojami duomenys apie mirtis, atsiranda tam tikras apribojimas. Pavyzdžiui, gali būti, kad iškart po parametro a pasikeitimo stebimas didelis mirčių šuolis, kuris rodo, kad a_2 gali būti didesnis už a_1 . Tačiau tikėtina, kad šiuos padidėjusius mirčių skaičius atitinkančios infekcijos iš tikrųjų pasireiškė dar iki parametro a pasikeitimo. Todėl šioje situacijoje gali būti nepelnytai suponuojama, kad $a_2 > a_1$.*

Gauti rezultatai pateikiami 4.7 pav. 4.7a pav. pavaizduotas į duomenis įbrėžtas įvertintas modelis, sudarytas iš deterministinės dalies (t. y. vidurkio) ir dviejų standartinių nuokrypių apribojančio intervalo, į kurį patenka apie 95% modelio generuojamų duomenų. Šis modelis laikytinas gerai atitinkančiu faktinius duomenis, nes jo vidurkis eina per faktinių duomenų vidurį, o į jį apribojantį intervalą patenka beveik visi faktiniai duomenys.

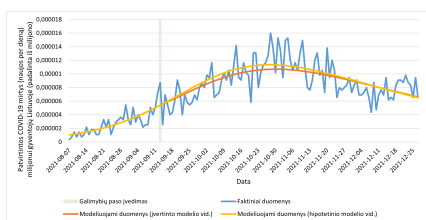
4.7b pav. vaizduoja tą patį, kaip ir 4.7a pav., tačiau taikant hipotetinį modelį, kuris visame modeliavimo horizonte naudoja įvertinto modelio parametą a_1 . Atsižvelgiant į tai, kad hipotetinis modelis taip pat gerai atitinka faktinius duomenis, darytina išvada, jog skirtumas tarp a_1 ir a_2 įverčių nėra reikšmingas. Tai reiškia, kad – išskyrus grynus atsitiktinumus – a_2 yra tapatus a_1 (žr. 4.7c pav., kur pavaizduoti abiejų modelių vidurkiai; matyti, kad a_2 įvertis yra mažesnis už a_1 įvertį).



(a)



(b)



(c)

4.7 pav.: COVID-19 galimybių pasą įvedimo įtakos delta bangai plisti Lietuvoje modeliavimas: (a) įvertintas modelis; (b) hipotetinis modelis; (c) įvertinto ir hipotetinio modelių lyginimas.

Taigi nėra pakankamai įrodymų daryti išvadą, kad COVID-19 galimybių pasas turėjo įtakos COVID-19 delta bangai plisti Lietuvoje.

4.6 pastaba. Šis rezultatas nesutampa su gautuoju darbe [108], kuriame autoriai daro išvadą, kad COVID-19 galimybių pasas turėjo stabdomąjį poveikį COVID-19 plisti Lietuvoje vyraujant delta bangai. Šis nesutapimas iš esmės gali būti siejamas su skirtingais natūralios pandemijos eigos aiškinimais taikomuose modeliuose.

[108] darbe daroma prielaida, kad tokie veiksniai kaip COVID-19 galimybių pasą naudojimas ir (arba) žmonių savimonė buvo būtini tam, kad infekcijų dinamika galiausiai pasiektų piką ir pereitų į atoslūgio fazę. Autoriai pateikia scenarijų, kuriame, nesant nė vieno iš šių veiksnių, užsikrėtimų

skaičius eksponentiškai auga iki begalybės. Tačiau toks aiškinimas iš esmės perversina situaciją, nes tokią dinamiką pirmiausia galima paaiškinti tuo, kad persirgus kritinei populiacijos masei, susirgimų skaičius neišvengiamai ima mažėti.

O šiame darbe pateiktas segmentinis modelis šį faktą įtraukia į savo struktūrą (žr. 4.7b pav., kur pavaizduotas hipotetinis modelis, atitinkantis analogišką eksponentiniam modeliui [108] darbe scenarijų, tačiau įgyvendinantis natūralią pandemijos dinamiką). Tokiu būdu sudaromos sąlygos adekvačiau palyginti skirtingus scenarijus (žr. 4.7c pav.).

Analizuojant antrąjį reiškinį, buvo elgiamasi analogiškai, tik naudojant omikrono bangą atitinkančią 2021 m. gruodžio 28 d.–2022 m. gegužės 8 d. duomenų dalį. Šiuo atveju parametro a pokyčio momentas laikomas 2022 m. vasario 5 d., kai COVID-19 galimybių pasas nustojo galioti.

Gauti rezultatai pateikti 4.8 pav. 4.8a pav. pavaizduotas įvertintas modelis, pritaikytas faktiniams duomenims ir juos atitinkantis.

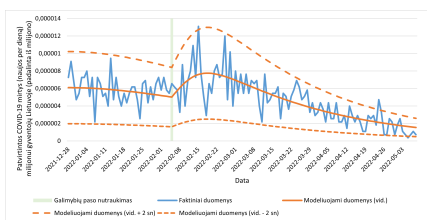
4.8b pav. pavaizduotas hipotetinis modelis, kuris neatitinka faktinių duomenų. Tai, kad hipotetinis modelis neatitinka faktinių duomenų, rodo, jog skirtumas tarp a_1 ir a_2 įverčių yra reikšmingas, o tai reiškia, kad a_2 skiriasi nuo a_1 net ir neatsižvelgiant į atsitiktinę galimybę (žr. 4.8c pav., kuriame vaizduojami abiejų modelių vidurkiai; matyti, kad a_2 įvertis yra didesnis už a_1 įvertį).

Vis dėlto šis rezultatas yra tam tikra apimtimi dviprasmiškas, nes detalesnė duomenų analizė (žr. 4.7 pastaba) rodo didelę tikimybę, kad šio reiškinio analizė kenčia nuo 4.5 pastaboje išsakyto apribojimo.

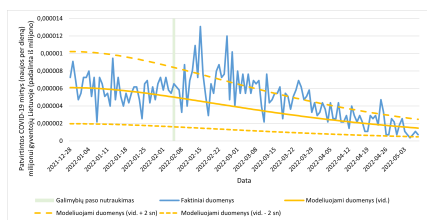
4.7 pastaba. Omikrono bangos metu užsikrėtimo atvejų skaičius Lietuvoje dramatiškai pradėjo mažėti iškart po COVID-19 galimybių paso panaikinimo. Tačiau COVID-19 mirčių skaičius dar kelias savaites toliau augo, kaip ir tikėtasi, atsižvelgiant į laiko tarpą tarp užsikrėtimo ir mirties.

Tyrėjai, kurie nepaiso natūralios pandemijos eigos ir analizuoja tik užsikrėtimo atvejus, gali daryti išvadą, kad COVID-19 galimybių paso panaikinimas sustabdė pandemiją. O tie, kurie nagrinėja tik COVID-19 mirčių dinamiką, gali daryti priešingą išvadą.

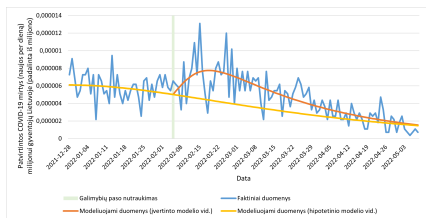
Vis dėlto tiek COVID-19 užsikrėtimo, tiek mirčių kreivės Lietuvoje buvo panašios į fiksuotas šalyse be COVID-19 galimybių pasų, pavyzdžiui, Norvegijoje, Švedijoje ir Lenkijoje.



(a)



(b)



(c)

4.8 pav.: COVID-19 galimybių paso nutraukimo įtakos omikrono bangai plisti Lietuvoje modeliavimas: (a) įvertintas modelis; (b) hipotetinis modelis; (c) įvertinto ir hipotetinio modelių lyginimas.

Taigi norint padaryti vienareikšmę išvadą, turėtų būti naudojami išsamesni duomenys.

4.1.3. Diskusija

Šiame poskyryje propaguojamas DTM taikymas vertinant dieninio lygmens diskretaus laiko deterministinio SIRD modelio parametrus, remiantis išimtinai COVID-19 mirčių duomenimis. Šie duomenys pasižymi santykinai nedaug trūkumų, į kuriuos galima atsižvelgti, į modelį įvedant jų stebėjimo triukšmą.

Siūlomas metodas leidžia įvertinti kertines pandemijos dinamines charakteristikas – užkrečiamumą, pasveikstamumą ir mirtingumą – taip, be kita ko, sudarant sąlygas įvairių pandemijos kontrolės priemonių veiksmingumo tyrimams.

Vis dėlto pažymėtina, kad nors DTM dėl savo (asimptotiškai) optimalių savybių vargiai galima pakeisti kaip parametrų vertinimo priemonę, to paties negalima pasakyti apie konkretų naudotą segmentinį modelį. Šis modelis tėra viena iš daugelio galimų abstrakcijų, iš kurių daugelis pasižymi didesniu detalumu. Tinkamiausio modelio

parinkimas konkrečiam tyrimo klausimui yra subtilus uždavinys, reikalaujantis gilių dalykinių įžvalgų, leidžiančių išskirti ir panaudoti aktualias detales.

Be to, siūlomas metodas yra susijęs su segmentinio modelio vertinimu, atsižvelgiant į vieno segmento perspektyvą. Tai veikia kaip išeitis, kai susiduriama su ribotą informaciją teikiančiais oficialiais COVID-19 duomenimis, tačiau tokia praktika savaime įveda tam tikrą apribojimą.

Norint pašalinti modeliavimo trukdžius, visų pirma būtina nukreipti pastangas į duomenų tobulinimą, nes joks metodas galiausiai negali visiškai pašalinti jų trūkumų. Šiuo atžvilgiu galima apsvarstyti galimybę susieti patvirtintų COVID-19 susirgimų duomenis su COVID-19 testų duomenimis, kurie šiuo metu yra nepagrįstai atsieti, nepaisant fundamentalaus jų tarpusavio ryšio.

Šį ryšį būtų galima atkurti fiksuojant patvirtinto susirgimo testo datą individualiu lygmeniu. Tokia procedūra iš esmės reikalauja tik kruopštesnio duomenų rinkimo ir (arba) pateikimo, o ne visiškai naujų procesų inicijavimo. Tai galėtų reikšmingai pagerinti COVID-19 modeliavimo galimybes, nes šis ryšys yra tiesiogiai susijęs su tikslesniu į I segmentą pereinančios populiacijos dalies nustatymu.

Negana to, analogiškai galima atskleisti ir ryšį tarp patvirtintų COVID-19 mirčių bei patvirtintų COVID-19 susirgimų, taip neutralizuojant 4.5 pastaboje minimą apribojimą.

Be prieinamų duomenų saugomos informacijos didinimo, prasminga nagrinėti ir lankstesnius segmentinius modelius – pavyzdžiui, modelius su stochastine latentine dalimi, kurie leistų atskirti skirtingos prigimties triukšmus, arba modelius su detalesniais segmentais ir ryšiais tarp jų. Tokiems plėtiniams būdingas sudėtingesnis matematinis aparatas, su kuriuo galima dirbti remiantis PVT principais.

Be to, vietoje modelių vertinimo remiantis COVID-19 mirčių duomenimis, galima naudoti bendrą mirtingumo perteklių, nes faktinės mirtys yra objektyvus kriterijus, o tikėtinos mirtys gali būti prognozuojamos [11].

4.2. Socialinio kapitalo modeliavimas

Socialinis kapitalas iš esmės reiškia veiksmingai veikiančių socialinių grupių ir bendruomenių elementus, tokius kaip tarpasmeniniai santy-

kiai, bendras tapatybės jausmas, bendras supratimas, bendros normos, bendros vertybės, pasitikėjimas, bendradarbiavimas ir abipusiškumas.

Ekonominio bendradarbiavimo ir plėtros organizacijos (EBPO) Statistikos direktoratas ėmėsi projekto, kurio tikslas – suprasti socialinio kapitalo sąvoką ir išaiškinti jo matavimo galimybes [100]. Galutinėje projekto ataskaitoje nurodomi keturi pagrindiniai būdai, kuriais galima plėtoti ir matuoti socialinį kapitalą:

1. Asmeniniai santykiai.
2. Socialinių tinklų palaikymas.
3. Pilietinis įsitraukimas.
4. Pasitikėjimo ir bendradarbiavimo normos.

Socialinio kapitalo formavimasis yra neatsiejamas nuo kultūrinių procesų. Tyrėjai yra nustatę tiesioginį ryšį tarp žmonių dalyvavimo kultūroje ir jų socialinio kapitalo pokyčių [4, 69], taip pat sudarę konceptualų modelį, atspindintį dalyvavimo kultūroje ir socialinio kapitalo sąveiką [67].

Tačiau [95] darbe apžvelgiama literatūra atskleidžia gana siaurą kultūrinio dalyvavimo ir socialinio kapitalo santykio tyrimų lauką. Dauguma mokslinių straipsnių yra susiję su specifinėmis kultūrinio dalyvavimo formomis ar sritimis bei tam tikrais socialinio kapitalo elementais (pvz., bibliotekų lankomumo ir savanorystės sąsajų matavimu).

Trūksta tyrimų, kurie analizuotų „daug su daug“ ryšius – tai yra ryšį tarp plataus kultūrinio dalyvavimo lauko ir bendresnio socialinio kapitalo lauko.

Tokie tyrimai atliktini išgryninant esmines procesą valdančių dėsnių prielaidas ir sudarant jas atitinkantį modelį, kuris leidžia darbą su jam tinkamais duomenimis. Alternatyviai galima iš karto ieškoti egzistuojančius duomenis atitinkančių modelių, tačiau toks požiūris gali riboti modeliavimo potencialą.

Kuriant kultūrinio dalyvavimo poveikio socialiniam kapitalui modelį pagal EBPO metodiką, [95] darbe atlikta literatūros analizė leidžia suformuluoti šias pagrindines prielaidas:

1. Egzistuoja kiekybinis ryšys tarp dalyvavimo kultūros procese ir socialinio kapitalo.

2. Kultūros procesų įtaka socialiniam kapitalui yra netiesinė ir priklauso nuo jau sukaupto socialinio kapitalo, kurio indeksus riboja nulis ir tam tikra maksimali vertė. Socialinio kapitalo kaupimo laikotarpiu ši įtaka yra reikšmingesnė, tačiau didėjant socialiniam kapitalui pasireiškia „prisotinimo“ arba „stabilizacijos“ efektas – tolesnis dalyvavimas kultūros procesuose nelemia reikšmingo socialinio kapitalo augimo.
3. Vieno kultūrinio įvykio įtaka yra nereikšminga, palyginti su bendru daugelio kultūrinių įvykių srauto poveikiu, kuris laikomas atskirų kultūrinių įvykių poveikių suma.
4. Socialinį kapitalą veikia ne tik kultūrinis dalyvavimas, bet ir kiti veiksniai – ekonominiai, psichologiniai bei politiniai.

Vadovaujantis EBPO socialinio kapitalo matavimo metodika, socialinis kapitalas suprantamas per keturis indeksus, kurių kiekvienas (vertinamas pagal klausimynus) įgyja reikšmę nuo 0 iki 60:

1. Asmeninių santykių indeksas C^{PR} , $0 \leq C^{PR} \leq 60$.
2. Socialinių tinklų palaikymo indeksas C^{SNS} , $0 \leq C^{SNS} \leq 60$.
3. Pilietinio įsitraukimo indeksas C^{CE} , $0 \leq C^{CE} \leq 60$.
4. Pasitikėjimo ir bendradarbiavimo normų indeksas C^{TCN} , $0 \leq C^{TCN} \leq 60$.

Socialinio kapitalo indeksų aibė žymima $\Theta = \{PR, SNS, CE, TCN\}$. Bendras individo (subjekto) socialinis kapitalas C gali būti išmatuotas kaip visų socialinio kapitalo indeksų suma: $C = \sum_{\theta \in \Theta} C^\theta$.

Kiekvienas individas tam tikru laiko momentu yra charakterizuojamas tam tikra socialinio kapitalo indekso reikšme prieš kultūrinį įvykį ir tam tikra reikšme po kultūrinio įvykio (prielaida Nr. 1). Nors viršutinė socialinio kapitalo indekso reikšmės riba pagal nagrinėjamą metodiką yra 60, atsižvelgiant į natūraliai tarp žmonių egzistuojančius skirtumus, galima manyti, kad kiekvienas individas pasižymi jam būdingu potencialiu socialiniu kapitalu (PSK), žymimu A^θ , $0 < A^\theta \leq 60$, $\theta \in \Theta$. Tada svarbią informaciją teikia socialinio kapitalo vystymosi santykis (SKVS), kuris apibūdina, kaip individas sugeba įsisavinti savo PSK:

$$D^\theta = \frac{C^\theta}{A^\theta - C^\theta}, \quad (4.10)$$

kur $0 < C^\theta < A^\theta$, $\theta \in \Theta$. Dėl socialinio kapitalo netiesiškumo ir aprėžtumo (prielaida Nr. 2), socialinio kapitalo indeksų pokyčiai adekvačiau aprašomi multiplikatyviai per SKVS, skirtingai nei ekonominis (finansinis) kapitalas, kurio pokyčiai suprantami adityviai.

Dalyvavimas kultūriniuose įvykiuose suprantamas kaip tikimybiniis atsakas į kultūrinių įvykių srautą laike. Šiuo tikslu įvedami parametrai $w_0^\theta \leq 0$ ir $b^\theta > 0$: w_0^θ apibūdina tikėtiną individo socialinio kapitalo pokytį jam nedalyvavus kultūriniame įvykyje, o b^θ – dalyvavus ($\theta \in \Theta$). Individo dalyvavimas kokiam nors kultūriniame įvykyje, vykstančiame per laiko vienetą Δt , žymimas indikatoriumi $\chi = 1$, o nedalyvavimas – $\chi = 0$. SKVS pokytis turėtų priklausyti nuo individo dalyvavimo kultūriniame įvykyje fakto. Taigi galima laikyti, kad šis santykis, turintis reikšmę D^θ prieš kultūrinį įvykį, po kultūrinio įvykio pasikeičia į $D_\Delta^\theta = D^\theta + \Delta D^\theta$ tokiu būdu:

$$D_\Delta^\theta = \begin{cases} D^\theta (1 + (w_0^\theta + \xi^\theta) \Delta t), & \chi = 0 \\ D^\theta (1 + (b^\theta + \xi^\theta) \Delta t), & \chi = 1 \end{cases}, \quad (4.11)$$

kur ξ^θ apima bendrą kitų veiksnių poveikį (prielaida Nr. 4), modeliuotiną pagal normalųjį dėsnį ($\theta \in \Theta$). Kadangi poveikis analizuojamas per pakankamai didelį kultūrinių įvykių skaičių, daroma prielaida (prielaida Nr. 3), kad vieno atskiros kultūrinio įvykio (o dėl stabilumo – atitinkamai ir kitų veiksnių) įtaka yra nedidelė, tai yra $|(w_0^\theta + \xi^\theta)\Delta t| \ll 1$ ir $|(b^\theta + \xi^\theta)\Delta t| \ll 1$, $\theta \in \Theta$.

Pažymėjus $\chi = (\chi_1, \dots, \chi_m)'$, kur χ_i yra dalyvavimo i tipo kultūriniame įvykyje indikatorius (laikant, kad yra m skirtingų kultūrinių įvykių tipų), $\chi_i \in \{0, 1\}$, $0 \leq \sum_{i=1}^m \chi_i \leq 1$, ir $w^\theta = (w_1^\theta, \dots, w_m^\theta)'$, kur $w_i^\theta = b_i^\theta - w_0^\theta$ (čia b_i^θ yra i tipo kultūrinio įvykio poveikis), $1 \leq i \leq m$, $\theta \in \Theta$, (4.11) lygtis gali būti perrašyta tokiu pavidalu:

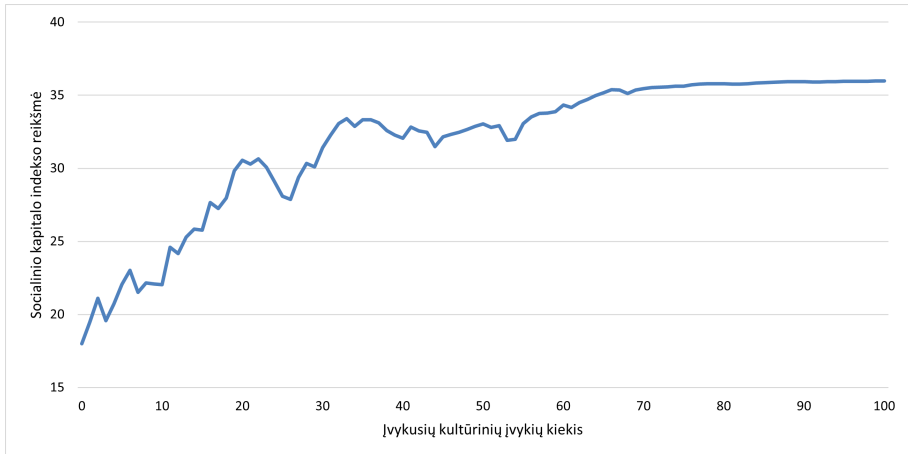
$$D_\Delta^\theta = D^\theta \left(1 + \left(w_0^\theta + \chi' w^\theta + \xi^\theta \right) \Delta t \right), \quad (4.12)$$

$\theta \in \Theta$. Pažymėtina, kad (4.12) lygtis atitinka logistinės regresijos modelį [89, 123].

(4.10) ryšys tarp SKVS, PSK ir socialinio kapitalo indekso gali būti perrašytas taip:

$$C^\theta = A^\theta \frac{D^\theta}{1 + D^\theta}, \quad (4.13)$$

$\theta \in \Theta$. Tada (4.12) ir (4.13) lygtys sudaro individo dalyvavimo kultūriniuose įvykiuose poveikio jo socialiniam kapitalui „daug su daug“ tipo modelį (pavyzdinis modelio įgyvendinimas pateikiamas 4.9 pav.).



4.9 pav.: Pavyzdinis (4.12) ir (4.13) lygtimis apibrėžiamo modelio įgyvendinimas suprantamas kaip individui būdinga kokio nors socialinio kapitalo indekso dinamika laike, kuris apskaitomas įvykusiais kultūriniais įvykiais.

4.3. Skyriaus išvados

- Pagrįstas išvadas generuojantis praktinių dinaminių sistemų modeliavimą yra neatsiejamas nuo gana lanksčių netiesinių modelių, aprašančių aktualius, fundamentaliai suvokiamus dėsnius, tačiau jų įgyvendinimas (parametrų įvertinimas) gali būti apribotas nepakankamai informacijos teikiančių prieinamų duomenų.

BENDROSIOS IŠVADOS

- Norint maksimaliai išnaudoti dinaminių sistemų modeliavimo galimybes didžiųjų duomenų eroje, būtini efektyvūs parametrų vertinimo algoritmai, kuriais būtų galima suderinti aukštą įverčių tikslumą su veikimo greitaveika. Vis dėlto esami sprendimai dažnai pasižymi tik viena iš šių savybių, ypač sudėtingesnių modelių atveju.
- Palaipsninio vertinimo technika suteikia metodologinį pagrindą efektyviems parametrų vertinimo algoritmams kurti stochastiniams dinaminiam modeliams, kai duomenų rinkinys yra fiksuotas. Remiantis didžiausio tikėtumo metodu ir rekursiniais skaičiavimais, ši technika leidžia suderinti įverčių optimalumą ir skaičiavimų efektyvumą, taip atliepiant šiuolaikinius duomenų analizės poreikius.
- Vienmatės išvesties grynąjį laiko eilučių duomenų kumuliacinis struktūrinių lygčių modelis su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis sudaro tinkamą pradinę terpę palaipsninio vertinimo technikos principams taikyti ir algoritmams kurti pagal metodologiją „iš apačios į viršų“. Taip sudarytas šio modelio parametrų vertinimo algoritmas, taikomas esant fiksuotam duomenų rinkiniui, pasižymi didžiausio tikėtumo parametrų įverčiais (kurių nauda ypač pasireiškia esant intensyviai išvesties stebėjimo triukšmui) bei stebėjimo momentų skaičiui proporcingu vykdymo laiku.
- Efektyvių algoritmų, skirtų parametrų vertinti realiuoju laiku, kūrimas kelia daugiau iššūkių už vertinimą naudojant fiksuotus duomenis. Vis dėlto jų sudarymas taip pat gali būti grindžiamas didžiausio tikėtumo metodu ir rekursija, todėl gali būti atliekamas remiantis palaipsninio vertinimo technikos rezultatais, juos atitinkamai praplečiant. Tokiu būdu sudarytas vienmatės išvesties grynąjį laiko eilučių duomenų kumuliacinio struktūrinių lygčių modelio su adityvinės paklaidos išvesties matavimo lygtimi bei normalumo ir nepriklausomumo prielaidomis parametrų vertinimo realiuoju laiku algoritmas pasižymi analogiškais įverčių savybėmis kaip ir atitinkamas algoritmas parametrų vertinti, kai duomenų rinkinys yra fiksuotas.

- Praktinis dinaminų sistemų modeliavimas reikalauja netiesinių modelių, atspindinčių fundamentalius dėsnius, ir išsamių duomenų, reikalingų jų parametrų tiksliai įvertinti. Kaip atskleidžia COVID-19 pandemijos dinamikos modeliavimo atvejis, empiriniai duomenų apribojimai gali reikšmingai sumažinti modeliavimo galimybes ir susilpninti išvadų patikimumą. Pavyzdžiui, viešai prieinamuose COVID-19 duomenyse trūksta informacijos, kuri leistų susieti testavimo, užsikrėtimo ir mirtingumo duomenis individo lygmeniu. Tokie duomenys yra būtini norint patikimai įvertinti tam tikrų intervencijų, pavyzdžiui, galimybių paso, poveikį pandemijos eigai. Todėl, taikant dinaminis modelius sprendimų priėmimo kontekste, ypač svarbu užtikrinti duomenų informacijos išsamumą ir kokybę.

LITERATŪROS SĄRAŠAS

- [1] K. M. Abadir and J. R. Magnus. *Matrix Algebra*. Cambridge University Press, 2012. doi: 10.1017/CBO9780511810800.
- [2] M. Aleem, M. Sufyan, I. Ameer, and M. Mustak. Remote Work and the COVID-19 Pandemic: An Artificial Intelligence-Based Topic Modeling and a Future Agenda. *Journal of Business Research*, 154:113303, 2023. doi: 10.1016/j.jbusres.2022.113303.
- [3] M. Allen. *The SAGE Encyclopedia of Communication Research Methods*. SAGE Publications, Inc, 2017. doi: 10.4135/9781483381411.
- [4] H. K. Anheier, R. A. List, O. Kononykhina, and J. L. Cohen. Cultural Participation and Inclusive Societies: A Thematic Report Based on the Indicator Framework on Culture and Democracy, 2017.
- [5] M. Aoki. *State Space Modeling of Time Series*. Springer Science & Business Media, 2013. doi: 10.1007/978-3-642-75883-6.
- [6] T. Asparouhov, E. L. Hamaker, and B. Muthén. Dynamic Structural Equation Models. *Structural Equation Modeling: A Multidisciplinary Journal*, 25(3):359–388, 2018. doi: 10.1080/10705511.2017.1406803.
- [7] K. Auranen, M. Shubin, E. Erra, S. Isosomppi, J. Kontto, T. Leino, and T. Lukkarinen. Efficacy and Effectiveness of Case Isolation and Quarantine During a Growing Phase of the COVID-19 Epidemic in Finland. *Scientific Reports*, 13(1):298, 2023. doi: 10.1038/s41598-022-27227-2.
- [8] A. Badshah, A. Daud, R. Alharbey, A. Banjar, A. Bukhari, and B. Alshemaimri. Big Data Applications: Overview, Challenges and Future. *Artificial Intelligence Review*, 57(11):290, 2024. doi: 10.1007/s10462-024-10938-5.
- [9] J. Bai and S. Shi. Estimating High Dimensional Covariance Matrices and Its Applications. *Annals of Economics and Finance*, 12(2): 199–215, 2011.
- [10] R. Bali Swain, X. Lin, and F. Y. Wallentin. COVID-19 Pandemic Waves: Identification and Interpretation of Global Data. *Heliyon*, 10(3):e25090, 2024. doi: 10.1016/j.heliyon.2024.e25090.
- [11] T. Beaney, J. M. Clarke, V. Jain, A. K. Golestaneh, G. Lyons, D. Salman, and A. Majeed. Excess Mortality: The Gold Standard in Measuring the Impact of COVID-19 Worldwide? *Journal of the Royal Society of Medicine*, 113(9):329–334, 2020. doi:

- 10.1177/0141076820956802.
- [12] I. Belovas and V. Starikovicius. Parallel Computing for Mixed-Stable Modelling of Large Data Sets. *Information Technology and Control*, 44(2):148–154, 2015. doi: 10.5755/j01.itc.44.2.6723.
 - [13] T. N. Beran and C. Violato. Structural Equation Modeling in Medical Research: A Primer. *BMC Research Notes*, 3(1):267, 2010. doi: 10.1186/1756-0500-3-267.
 - [14] D. S. Bernstein. *Matrix Mathematics: Theory, Facts, and Formulas*. Princeton University Press, 2009. doi: 10.1515/9781400833344.
 - [15] B. M. Byrne. *Structural Equation Modeling with Mplus: Basic Concepts, Applications, and Programming*. Routledge/Taylor & Francis Group, New York, NY, US, 2012. ISBN 978-1-84872-839-4 (Paperback); 978-0-8058-5986-7 (Hardcover).
 - [16] G. C. Calafiore, C. Novara, and C. Possieri. A Time-Varying SIRD Model for the COVID-19 Contagion in Italy. *Annual Reviews in Control*, 50:361–372, 2020. doi: 10.1016/j.arcontrol.2020.10.005.
 - [17] C. Champagne and B. Cazelles. Comparison of Stochastic and Deterministic Frameworks in Dengue Modelling. *Mathematical Biosciences*, 310:1–12, 2019. doi: 10.1016/j.mbs.2019.01.010.
 - [18] D. Child. *The Essentials of Factor Analysis*. Bloomsbury Academic, 2006. ISBN 9780826480002 (Paperback).
 - [19] S. Choo. *Aggregate Relationships Between Telecommunications and Travel: Structural Equation Modeling of Time Series Data*. PhD thesis, University of California, Davis, 2004.
 - [20] D. Ciraki. *Dynamic Structural Equation Models: Estimation and Interference*. PhD thesis, London School of Economics and Political Science, 2007.
 - [21] W. S. Cleveland. LOWESS: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression. *American Statistician*, 35(1):54, 1981. doi: 10.2307/2683591.
 - [22] P. Crepey, H. Noël, and S. Alizon. Challenges for Mathematical Epidemiological Modelling. *Anaesthesia, Critical Care & Pain Medicine*, 41(2):101053, 2022. doi: 10.1016/j.accpm.2022.101053.
 - [23] S. H. Dennis, K. Edler, and M. Cummings. Relations Between Yearly and Daily Patterns in Fathering and Mothering: An Application of Dynamic Structural Equation Modeling to the Family System. *Parenting*, 0(0):1–32, 2025. doi: 10.1080/15295192.2025.

2481437.

- [24] J. Ding, V. Tarokh, and Y. Yang. Model Selection Techniques: An Overview. *IEEE Signal Processing Magazine*, 35(6):16–34, 2018. doi: 10.1109/MSP.2018.2867638.
- [25] S. H. C. du Toit and M. W. Browne. Structural Equation Modeling of Multivariate Time Series. *Multivariate Behavioral Research*, 42(1): 67–101, 2007. doi: 10.1080/00273170701340953.
- [26] R. Dubakienė, V. Kučinskas, R. Leipus, K. Sasnauskas, S. Burokienė, O. Štikonienė, L. Jančorienė, T. Kačergius, V. Kasiulevičius, A. Račkauskas, V. Usonis, and A. Žvirblienė. Lietuvos mokslų akademijos COVID-19 komisija apie pandemiją ir jos valdymą, 2020. URL <https://www.lma.lt/uploads/Pranesimai%20naujienoms/straipsnis%20apie%20COVID-19.pdf>.
- [27] I. S. Duff, A. M. Erisman, and J. K. Reid. *Direct Methods for Sparse Matrices*. Oxford University Press, 2017. ISBN 9780198508380 (Hardback).
- [28] V. Dulskis and L. Sakalauskas. Toward Efficient Online Estimation of Dynamic Structural Equation Models: A Case Study. *Journal of Statistical Computation and Simulation*, 0(0):1–24, 2025. doi: 10.1080/00949655.2025.2515955.
- [29] V. Dulskis and L. Sakalauskas. *EMLI: Efficient Maximum Likelihood Identification*, 2025. URL <https://CRAN.R-project.org/package=EMLI>. R package version 0.3.0.
- [30] C. Fan, R. Lee, Y. Yang, and A. Mostafavi. Fine-Grained Data Reveal Segregated Mobility Networks and Opportunities for Local Containment of COVID-19. *Scientific Reports*, 11(1):16895, 2021. doi: 10.1038/s41598-021-95894-8.
- [31] R. Farooq. Heywood Cases: Possible Causes and Solutions. *International Journal of Data Analysis Techniques and Strategies*, 14(1): 79–88, 2022. doi: 10.1504/IJDATS.2022.121506.
- [32] J. Fernández-Villaverde and C. I. Jones. Estimating and Simulating a SIRD Model of COVID-19 for Many Countries, States, and Cities. *Journal of Economic Dynamics and Control*, 140:104318, 2022. doi: 10.1016/j.jedc.2022.104318.
- [33] M. J. Fisher and A. P. Marshall. Understanding Descriptive Statistics. *Australian Critical Care*, 22(2):93–97, 2009. doi: 10.1016/j.aucc.2008.11.003.

- [34] P. A. Fishwick. *Handbook of Dynamic System Modeling*. Chapman and Hall/CRC, 2007. doi: 10.1201/9781420010855.
- [35] D. J. Foster, Y. Han, J. Qian, and A. Rakhlin. Online Estimation via Offline Estimation: An Information-Theoretic Framework. *arXiv preprint arXiv:2404.10122*, 2024. doi: 10.48550/arXiv.2404.10122.
- [36] P. H. Garthwaite. Properties of Estimators. In M. Lovric, editor, *International Encyclopedia of Statistical Science*, pages 1132–1135. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-04898-2_462.
- [37] R. Glanville. Black Boxes. *Cybernetics & Human Knowing*, 16(1-2): 153–167, 2009.
- [38] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1996. ISBN 9780801854149 (Paperback).
- [39] C. Godske. On the Time Dependence of Smoothed Variables. *Tellus*, 18(4):714–721, 1966. doi: 10.3402/tellusa.v18i4.9691.
- [40] W. Gong, S. Parkkila, X. Wu, and A. Aspatwar. SARS-CoV-2 Variants and COVID-19 Vaccines: Current Challenges and Future Strategies. *International Reviews of Immunology*, 42(6):393–414, 2023. doi: 10.1080/08830185.2022.2079642.
- [41] A. K. Goodboy, M. R. Dillow, M. Shin, R. M. Chiasson, and M. J. Zyphur. Testing Relational Turbulence Theory in Daily Life Using Dynamic Structural Equation Modeling. *Journal of Communication*, 74(3):249–264, 2024. doi: 10.1093/joc/jqae010.
- [42] J. B. Grace. *Structural Equation Modeling and Natural Systems*. Cambridge University Press, 2006. doi: 10.1017/CBO9780511617799.
- [43] Q. Griette, J. Demongeot, and P. Magal. What Can We Learn From COVID-19 Data by Using Epidemic Models with Unidentified Infectious Cases? *Mathematical Biosciences and Engineering*, 19(1): 537–594, 2022. doi: 10.3934/mbe.2022025.
- [44] J. F. Hair, G. T. M. Hult, C. M. Ringle, M. Sarstedt, N. P. Danks, and S. Ray. An Introduction to Structural Equation Modeling. In *Partial Least Squares Structural Equation Modeling (PLS-SEM) Using R: A Workbook*, pages 1–29. Springer International Publishing, 2021. doi: 10.1007/978-3-030-80519-7_1.
- [45] E. Hamaker, T. Asparouhov, and B. Muthén. Dynamic Structural Equation Modeling as a Combination of Time Series Modeling, Multilevel Modeling, and Structural Equation Modeling. In R. H.

- Hoyle, editor, *Handbook of Structural Equation Modeling*, chapter 31. Guilford Press, 2023. ISBN 9781462544646 (Hardcover).
- [46] E. L. Hamaker, C. V. Dolan, and P. C. M. Molenaar. On the Nature of SEM Estimates of ARMA Parameters. *Structural Equation Modeling: A Multidisciplinary Journal*, 9(3):347–368, 2002. doi: 10.1207/S15328007SEM0903_3.
 - [47] E. L. Hamaker, C. V. Dolan, and P. C. M. Molenaar. ARMA-Based SEM When the Number of Time Points T Exceeds the Number of Cases N: Raw Data Maximum Likelihood. *Structural Equation Modeling: A Multidisciplinary Journal*, 10(3):352–379, 2003. doi: 10.1207/S15328007SEM1003_2.
 - [48] E. L. Hamaker, R. P. P. P. Grasman, and J. H. Kamphuis. Modeling BAS Dysregulation in Bipolar Disorder: Illustrating the Potential of Time Series Analysis. *Assessment*, 23(4):436–446, 2016. doi: 10.1177/1073191116632339.
 - [49] A. Hasl, M. Voelkle, J. Kretschmann, D. Richter, and M. Brunner. A Dynamic Structural Equation Approach to Modeling Wage Dynamics and Cumulative Advantage Across the Lifespan. *Multivariate Behavioral Research*, 58(3):504–525, 2023. doi: 10.1080/00273171.2022.2029339.
 - [50] L. S. T. Ho, F. W. Crawford, and M. A. Suchard. Direct Likelihood-Based Inference for Discretely Observed Stochastic Compartmental Models of Infectious Disease. *The Annals of Applied Statistics*, 12(3):1993–2021, 2018. doi: 10.1214/18-AOAS1141.
 - [51] G. Hu and R. O’Connell. Analytical Inversion of Symmetric Tridiagonal Matrices. *Journal of Physics A: Mathematical and General*, 29: 1511, 1999. doi: 10.1088/0305-4470/29/7/020.
 - [52] O. C. Ibe. *Elements of Random Walk and Diffusion Processes*. John Wiley & Sons, 2013. doi: 10.1002/9781118618059.
 - [53] R. Isermann and M. Münchhof. *Identification of Dynamic Systems: An Introduction with Applications*. Springer Berlin, Heidelberg, 2011. doi: 10.1007/978-3-540-78879-9.
 - [54] B. Jahn, S. Friedrich, J. Behnke, J. Engel, U. Garczarek, R. Münnich, M. Pauly, A. Wilhelm, O. Wolkenhauer, M. Zwick, et al. On the Role of Data, Statistics and Decisions in a Pandemic. *AStA Advances in Statistical Analysis*, 106(3):349–382, 2022. doi: 10.1007/s10182-022-00439-7.

- [55] R. Jankunas, L. Sakalauskas, K. Zamaryte-Sakaviciene, D. Stakisaitis, and M. Helmersen. Commentary on the Impact of the COVID-19 Passports in Lithuania. *Journal of Infection*, 86(3):e78–e79, 2022. doi: 10.1016/j.jinf.2022.12.013.
- [56] Z. L. Jiesisibieke, W.-Y. Liu, Y.-P. Yang, C.-W. Chien, and T.-H. Tung. Effectiveness and Safety of COVID-19 Vaccinations: An Umbrella Meta-Analysis. *International Journal of Public Health*, 68: 1605526, 2023. doi: 10.3389/ijph.2023.1605526.
- [57] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis*. Pearson India, 2015. ISBN 9789332549555 (Paperback).
- [58] K. G. Jöreskog and M. van Thillo. LISREL: A General Computer Program for Estimating a Linear Structural Equation System Involving Multiple Indicators of Unmeasured Variables. *ETS Research Bulletin Series*, 1972(2):i–71, 1972. doi: 10.1002/j.2333-8504.1972.tb00827.x.
- [59] S. Kassim, H. Hasan, A. Ismon, and F. Asri. Parameter Estimation in Factor Analysis: Maximum Likelihood Versus Principal Component. *AIP Conference Proceedings*, 1522:1293–1299, 2013. doi: 10.1063/1.4801279.
- [60] E. Kılıç and P. Stanica. The Inverse of Banded Matrices. *Journal of Computational and Applied Mathematics*, 237(1):126–135, 2013. doi: 10.1016/j.cam.2012.07.018.
- [61] Y. Kim and H. Bang. Introduction to Kalman Filter and Its Applications. In F. Govaers, editor, *Introduction and Implementations of the Kalman Filter*, chapter 2. IntechOpen, 2018. doi: 10.5772/intechopen.80600.
- [62] R. B. Kline. *Principles and Practice of Structural Equation Modeling*. The Guilford Press, New York, NY, US, 2016. ISBN 978-1-4625-2334-4 (Paperback); 978-1-4625-2335-1 (Hardcover).
- [63] D. P. Kroese, T. Brereton, T. Taimre, and Z. I. Botev. Why the Monte Carlo Method Is So Important Today. *Wiley Interdisciplinary Reviews: Computational Statistics*, 6(6):386–392, 2014. doi: 10.1002/wics.1314.
- [64] A. Kroll. Grey-Box Models: Concepts and Application. *New Frontiers in Computational Intelligence and Its Applications*, 57:42–51, 2000.
- [65] C. Kuhbandner, S. Homburg, H. Walach, and S. Hockertz. Was

- Germany's Lockdown in Spring 2020 Necessary? How Bad Data Quality Can Turn a Simulation Into a Delusion That Shapes the Future. *Futures*, 135:102879, 2022. doi: 10.1016/j.futures.2021.102879.
- [66] E. Kuhl. Data-Driven Modeling of COVID-19—Lessons Learned. *Extreme Mechanics Letters*, 40:100921, 2020. doi: 10.1016/j.eml.2020.100921.
- [67] R. Laužikas and D. Plikynas. Towards Conceptually Novel Oscillating Agent-Based Simulation of the Relationship Between Cultural Participation and Social Capital. In N. Agarwal, L. Sakalauskas, and G.-W. Weber, editors, *Modeling and Simulation of Social-Behavioral Phenomena in Creative Societies*, pages 126–144. Springer, Cham, 2019. doi: 10.1007/978-3-030-29862-3_10.
- [68] I. Lee. Big Data: Dimensions, Evolution, Impacts, and Challenges. *Business Horizons*, 60(3):293–303, 2017. doi: 10.1016/j.bushor.2017.01.004.
- [69] Y. Li, M. Savage, and A. Warde. Social Stratification, Social Capital and Cultural Practice in the UK. In Y. Li, editor, *Handbook of Research Methods and Applications in Social Capital*, pages 21–39. Edward Elgar Publishing, 2015. doi: 10.4337/9780857935854.00006.
- [70] G. Lillacci and M. Khammash. Parameter Estimation and Model Selection in Computational Biology. *PLOS Computational Biology*, 6(3):1–17, 2010. doi: 10.1371/journal.pcbi.1000696.
- [71] D. Luengo, L. Martino, M. Bugallo, V. Elvira, and S. Särkkä. A Survey of Monte Carlo Methods for Parameter Estimation. *EURASIP Journal on Advances in Signal Processing*, 2020(1):25, 2020. doi: 10.1186/s13634-020-00675-6.
- [72] S. Luthra, S. Agrawal, A. Kumar, M. Sharma, S. Joshi, and J. Kumar. Psychological Well-Being of Young Adults During COVID-19 Pandemic: Lesson Learned and Future Research Agenda. *Heliyon*, 9(5):e15841, 2023. doi: 10.1016/j.heliyon.2023.e15841.
- [73] F. J. Martínez-López, J. C. Gázquez-Abad, and C. M. Sousa. Structural Equation Modelling in Marketing and Business Research: Critical Issues and Practical Recommendations. *European Journal of Marketing*, 47(1/2):115–152, 2013. doi: 10.1108/03090561311285484.
- [74] E. Mathieu, H. Ritchie, L. Rodés-Guirao, C. Appel, C. Giat-

- tino, J. Hasell, B. Macdonald, S. Dattani, D. Beltekian, E. Ortiz-Ospina, and M. Roser. Coronavirus Pandemic (COVID-19). *Our World in Data*, 2020. URL <https://ourworldindata.org/coronavirus>.
- [75] D. McNeish and E. L. Hamaker. A Primer on Two-Level Dynamic Structural Equation Models for Intensive Longitudinal Data in Mplus. *Psychological Methods*, 25(5):610–635, 2019. doi: 10.1037/met0000250.
 - [76] J. Mendes and P. Coelho. Generalised SEIR Modelling of the COVID-19 Pandemic Course: Data Quality Issues and Structural Analysis. *Research Square (Preprint)*, 2021. doi: 10.21203/rs.3.rs-249261/v1.
 - [77] F. Morrison. *The Art of Modeling Dynamic Systems: Forecasting for Chaos, Randomness and Determinism*. Dover Publications, 2008. ISBN 9780486462950 (Paperback).
 - [78] A. Mustapha, L. Mohamed, and K. Ali. An Overview of Gradient Descent Algorithm Optimization in Machine Learning: Application in the Ophthalmology Field. In M. Hamlich, L. Bellatreche, A. Mondal, and C. Ordonez, editors, *Smart Applications and Data Analysis*, pages 349–359. Springer International Publishing, 2020. doi: 10.1007/978-3-030-45183-7.
 - [79] B. O. Muthén. Mplus Technical Appendices. *Los Angeles, CA: Muthén & Muthén*, 2004.
 - [80] I. J. Myung. Tutorial on Maximum Likelihood Estimation. *Journal of Mathematical Psychology*, 47(1):90–100, 2003. doi: 10.1016/S0022-2496(02)00028-7.
 - [81] B. Ninness and S. Henriksen. Bayesian System Identification via Markov Chain Monte Carlo Techniques. *Automatica*, 46(1):40–51, 2010. doi: 10.1016/j.automatica.2009.10.015.
 - [82] Official data collated by Our World in Data – processed by Our World in Data. “New tests” [dataset]. Official data collated by Our World in Data [original data].
 - [83] Official data collated by Our World in Data – processed by Our World in Data. “Positive test rate” [dataset]. Official data collated by Our World in Data [original data].
 - [84] Ö. Özmen, J. J. Nutaro, L. L. Pullum, and A. Ramanathan. Analyzing the Impact of Modeling Choices and Assumptions in Com-

- partmental Epidemiological Models. *Simulation*, 92(5):459–472, 2016. doi: 10.1177/0037549716640877.
- [85] V. Pan. Complexity of Computations with Matrices and Polynomials. *SIAM Review*, 34(2):225–262, 1992. doi: 10.1137/1034049.
- [86] D. Panarello and G. Tassinari. One Year of COVID-19 in Italy: Are Containment Policies Enough to Shape the Pandemic Pattern? *Socio-Economic Planning Sciences*, 79:101120, 2022. doi: 10.1016/j.seps.2021.101120.
- [87] P. M. Podsakoff, S. B. MacKenzie, and N. P. Podsakoff. Recommendations for Creating Better Concept Definitions in the Organizational, Behavioral, and Social Sciences. *Organizational Research Methods*, 19(2):159–203, 2016. doi: 10.1177/1094428115624965.
- [88] B. T. Polyak. Newton’s Method and Its Use in Optimization. *European Journal of Operational Research*, 181(3):1086–1096, 2007. doi: 10.1016/j.ejor.2005.06.076.
- [89] D. J. Polzella and J. S. Forbis. Relationships Between Traditional Music Audience Participation and Pro-Social Behaviors. *Empirical Studies of the Arts*, 32(1):109–120, 2014. doi: 10.2190/EM.32.1g.
- [90] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. Section 10.2.: Golden Section Search in One Dimension. In *Numerical Recipes: The Art of Scientific Computing*, pages 492–496. Cambridge University Press, 2007. ISBN 9780521880688 (Hardcover).
- [91] E. Pruyt, S. Cunningham, J. H. Kwakkel, and J. De Bruijn. From Data-Poor to Data-Rich: System Dynamics in the Era of Big Data. In P. Davidsen and E. A. J. A. Rouwette, editors, *Proceedings of the 32nd International Conference of the System Dynamics Society*, pages 2458–2469, 2014. ISBN 978-1-935056-13-3.
- [92] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022. URL <https://www.R-project.org/>.
- [93] E. E. Rigdon. Structural Equation Modeling. In G. A. Marcoulides, editor, *Modern Methods for Business Research*, pages 251–294. Psychology Press, 1998. doi: 10.4324/9781410604385.
- [94] L. Rinaldi. Accounting and the COVID-19 Pandemic Two Years On: Insights, Gaps, and an Agenda for Future Research. *Accounting Forum*, 47(3):333–364, 2023. doi: 10.1080/01559982.2022.

2045418.

- [95] L. Sakalauskas, V. Dulskis, R. Lauzikas, A. Miliauskas, and D. Plikynas. A Probabilistic Model of the Impact of Cultural Participation on Social Capital. *The Journal of Mathematical Sociology*, 45(2):65–78, 2021. doi: 10.1080/0022250X.2020.1725002.
- [96] L. Sakalauskas, V. Dulskis, and R. J. Jankunas. Compartmental Modeling for Pandemic Data Analysis: The Gap Between Statistics and Models. *Heliyon*, 10(11):e31410, 2024. doi: 10.1016/j.heliyon.2024.e31410.
- [97] L. Sakalauskas, V. Dulskis, and D. Plikynas. A Technique for Efficient Estimation of Dynamic Structural Equation Models: A Case Study. *Structural Equation Modeling: A Multidisciplinary Journal*, 31(04):635–650, 2024. doi: 10.1080/10705511.2023.2282378.
- [98] D. J. Salkeld and M. F. Antolin. Ecological Fallacy and Aggregated Data: A Case Study of Fried Chicken Restaurants, Obesity and Lyme Disease. *EcoHealth*, 17:4–12, 2020. doi: 10.1007/s10393-020-01472-1.
- [99] N. K. Schuurman, J. H. Houtveen, and E. L. Hamaker. Incorporating Measurement Error in n=1 Psychological Autoregressive Modeling. *Frontiers in Psychology*, 6:1038, 2015. doi: 10.3389/fpsyg.2015.01038.
- [100] K. Scrivens and C. Smith. Four Interpretations of Social Capital: An Agenda for Measurement, 2013. OECD Statistics Working Papers.
- [101] A. Sen and M. Srivastava. *Regression Analysis: Theory, Methods, and Applications*. Springer Science & Business Media, 2012. doi: 10.1007/978-1-4612-4470-7.
- [102] S. SeyedAlinaghi, A. Karimi, A. M. Afsahi, P. Mirzapour, S. Varshochi, H. Mojdeganlou, P. Mojdeganlou, A. Razi, S. Alilou, M. Dashti, et al. The Effectiveness of Face Masks in Preventing COVID-19 Transmission: A Systematic Review. *Infectious Disorders - Drug Targets*, 23(8):19–29, 2023. doi: 10.2174/1871526523666230601090905.
- [103] A. F. Siegenfeld, P. K. Kollepara, Y. Bar-Yam, et al. Modeling Complex Systems: A Case Study of Compartmental Models in Epidemiology. *Complexity*, 2022, 2022. doi: 10.1155/2022/3007864.
- [104] H. Singer. SEM Modeling with Singular Moment Matrices Part

- I: ML-Estimation of Time Series. *Journal of Mathematical Sociology*, 34:301–320, 2010. doi: 10.1080/0022250X.2010.509524.
- [105] J. C. Spall. On Monte Carlo Methods for Estimating the Fisher Information Matrix in Difficult Problems. In *2009 43rd Annual Conference on Information Sciences and Systems*, pages 741–746. IEEE, 2009. doi: 10.1109/CISS.2009.5054816.
- [106] J. S. Speagle. A Conceptual Introduction to Markov Chain Monte Carlo Methods. *arXiv preprint arXiv:1909.12313v2*, 2020. doi: 10.48550/arXiv.1909.12313.
- [107] R. K. Standish. Why Occam’s Razor. *Foundations of Physics Letters*, 17(3):255–266, 2004. doi: 10.1023/B:FOPL.0000032475.18334.0e.
- [108] M. Stankūnas, A. Džiugys, G. Skarbalius, E. Misiulis, and R. Navakas. Evaluating the Potential Impact of COVID-19 Passports in Lithuania. *Journal of Infection*, 85(3):334–363, 2022. doi: 10.1016/j.jinf.2022.05.039.
- [109] M. Stankūnas, A. Džiugys, G. Skarbalius, E. Misiulis, and R. Navakas. Authors’ Reply to a Commentary on the Potential Impact of COVID-19 Passports to Epidemiological Situation. *Journal of Infection*, 87(3):e51–e53, 2023. doi: 10.1016/j.jinf.2023.06.010.
- [110] Statistics Lithuania. COVID Dashboards. <https://osp.stat.gov.lt/covid-dashboards>. Accessed: 2023-12-29.
- [111] R. Sun. *Duality of the Mind: A Bottom-up Approach Toward Cognition*. Psychology Press, 2001. ISBN 9781410604378 (eBook).
- [112] P. Tarka. An Overview of Structural Equation Modeling: Its Beginnings, Historical Development, Usefulness and Controversies in the Social Sciences. *Quality & Quantity*, 52(1):313–354, 2018. doi: 10.1007/s11135-017-0469-8.
- [113] J. Terceiro Lomba. *Estimation of Dynamic Econometric Models with Errors in Variables*. Springer-Verlag, 1990. ISBN 9783540523581 (Paperback).
- [114] P. Toulis and E. M. Airolidi. Scalable Estimation Strategies Based on Stochastic Approximations: Classical Results and New Insights. *Statistics and Computing*, 25(4):781–795, 2015. doi: 10.1007/s11222-015-9560-y.
- [115] D. Y. Trejos, J. C. Valverde, and E. Venturino. Dynamics of Infectious Diseases: A Review of the Main Biological Aspects and Their Mathematical Translation. *Applied Mathematics and Nonlinear*

- Sciences*, 7(1):1–26, 2022. doi: 10.2478/amns.2021.1.00012.
- [116] J. Vaičiulytė and L. Sakalauskas. Recursive Estimation of Multivariate Hidden Markov Model Parameters. *Computational Statistics*, 34(3):1337–1353, 2019. doi: 10.1007/s00180-019-00877-z.
 - [117] J. Vaičiulytė and L. Sakalauskas. Recursive Parameter Estimation Algorithm of the Dirichlet Hidden Markov Model. *Journal of Statistical Computation and Simulation*, 90(2):306–323, 2020. doi: 10.1080/00949655.2019.1679144.
 - [118] M. C. Voelkle, J. H. L. Oud, T. von Oertzen, and U. Lindenberger. Maximum Likelihood Dynamic Factor Modeling for Arbitrary N and T Using SEM. *Structural Equation Modeling: A Multidisciplinary Journal*, 19(3):329–350, 2012. doi: 10.1080/10705511.2012.687656.
 - [119] M. P. Walkowiak, J. B. Walkowiak, and D. Walkowiak. COVID-19 Passport as a Factor Determining the Success of National Vaccination Campaigns: Does It Work? The Case of Lithuania vs. Poland. *Vaccines*, 9(12):1498, 2021. doi: 10.3390/vaccines9121498.
 - [120] Y. Wang, G. Gunter, M. Nice, M. L. Delle Monache, and D. B. Work. Online Parameter Estimation Methods for Adaptive Cruise Control Systems. *IEEE Transactions on Intelligent Vehicles*, 6(2): 288–298, 2021. doi: 10.1109/TIV.2020.3023674.
 - [121] WHO COVID-19 Dashboard – processed by Our World in Data. “New cases” [dataset]. WHO COVID-19 Dashboard [original data].
 - [122] WHO COVID-19 Dashboard – processed by Our World in Data. “New deaths” [dataset]. WHO COVID-19 Dashboard [original data].
 - [123] R. Williams. Understanding and Interpreting Generalized Ordered Logit Models. *The Journal of Mathematical Sociology*, 40(1): 7–20, 2016. doi: 10.1080/0022250X.2015.1112384.
 - [124] V. V. Williams, Y. Xu, Z. Xu, and R. Zhou. New Bounds for Matrix Multiplication: From Alpha to Omega. In D. P. Woodruff, editor, *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3792–3835. SIAM, 2024. doi: 10.1137/1.9781611977912.134.
 - [125] C. J. Willmott and K. Matsuura. Advantages of the Mean Absolute Error (MAE) Over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, 30(1):

- 79–82, 2005. doi: 10.3354/cr030079.
- [126] M. Wright. The Interior-Point Revolution in Optimization: History, Recent Developments, and Lasting Consequences. *Bulletin of the American Mathematical Society*, 42(1):39–56, 2005. doi: 10.1090/S0273-0979-04-01040-7.
 - [127] R. Wright. Statistical Structures Underlying Quantum Mechanics and Social Science. *International Journal of Theoretical Physics*, 46(8): 2026–2045, 2007. doi: 10.1007/s10773-006-9297-7.
 - [128] X. Yang. Introduction to Stochastic Calculus and Its Applications. Available at SSRN 3607647, 2020. doi: 10.2139/ssrn.3607647.
 - [129] A. G. Yong and S. Pearce. A Beginner’s Guide to Factor Analysis: Focusing on Exploratory Factor Analysis. *Tutorials in Quantitative Methods for Psychology*, 9(2):79–94, 2013. doi: 10.20982/tqmp.09.2.p079.
 - [130] Z. Zhang, E. L. Hamaker, and J. R. Nesselroade. Comparisons of Four Methods for Estimating a Dynamic Factor Model. *Structural Equation Modeling: A Multidisciplinary Journal*, 15(3):377–402, 2008. doi: 10.1080/10705510802154281.
 - [131] H. Zhao, N. N. Merchant, A. McNulty, T. A. Radcliff, M. J. Cote, R. S. Fischer, H. Sang, and M. G. Ory. COVID-19: Short Term Prediction Model Using Daily Incidence Data. *PLoS One*, 16(4): e0250110, 2021. doi: 10.1371/journal.pone.0250110.
 - [132] L. Zhou, M. Wang, and Z. Zhang. Intensive Longitudinal Data Analyses with Dynamic Structural Equation Modeling. *Organizational Research Methods*, 24(2):219–250, 2021. doi: 10.1177/1094428119833164.
 - [133] Y. Zhu, F. Liu, Y. Bai, Z. Zhao, C. Ma, A. Wu, L. Ning, and X. Nie. Effectiveness Analysis of Multiple Epidemic Prevention Measures in the Context of COVID-19 Using the SVIRD Model and Ensemble Kalman Filter. *Heliyon*, 9(3):e14231, 2023. doi: 10.1016/j.heliyon.2023.e14231.
 - [134] C. Zimmer, R. Yaesoubi, and T. Cohen. A Likelihood Approach for Real-Time Calibration of Stochastic Compartmental Epidemic Models. *PLoS Computational Biology*, 13(1):e1005257, 2017. doi: 10.1371/journal.pcbi.1005257.
 - [135] C. Zimmer, S. I. Leuba, T. Cohen, and R. Yaesoubi. Accurate Quantification of Uncertainty in Epidemic Parameter Estimates

- and Predictions Using Stochastic Compartmental Models. *Statistical Methods in Medical Research*, 28(12):3591–3608, 2019. doi: 10.1177/0962280218805780.
- [136] S. Zlobec. Characterizing Zero-Derivative Points. *Journal of Global Optimization*, 46:155–161, 2010. doi: 10.1007/s10898-009-9457-4.

PRIEDAS

R paketas EMLI

2.1 ir 3.1 algoritmai yra įdiegti R pakete EMLI, kuris pasiekiamas per CRAN [29, 92]. Šiame skyriuje siekiama atskleisti jo esmę ir veikimo principus (naudojami antrojo skyriaus žymėjimai). Prieš naudojantis paketu, rekomenduojama papildomai peržiūrėti jo dokumentaciją.

EMLI yra *computationally Efficient Maximum Likelihood Identification of linear dynamical systems* (liet. tiesinių dinaminių sistemų identifikavimas skaičiavimo požičiu efektyviu didžiausio tikėtumo metodu) trumpinys. Šis R paketas realizuoja skaičiavimo požičiu efektyvius didžiausio tikėtumo parametrų vertinimo algoritmus, skirtus modeliams, atspindintiems tiesines dinamines sistemas. Šio skyriaus rašymo metu įdiegti du tokie algoritmai (2.1 ir 3.1 algoritmai).

Be pagrindinės užduoties – parametrų vertinimo – EMLI paketas taip pat siūlo papildomų funkcijų, skirtų kitoms sistemų identifikavimo kontekste atsirandančioms užduotims, tokioms kaip duomenų simulavimas ir modelio tinkamumo vertinimas. Šio skyriaus rašymo metu pakete yra dešimt funkcijų, iš kurių penkios yra eksportuojamos (t. y. pasiekiamos naudotojui), o likusios penkios – neeksportuojamos ir naudojamos tik kodui organizuoti. Naudotojui prieinamos funkcijos yra šios:

1. `generate_data`: generuoja duomenis T laiko momentų (kadan-gi R statistiniame pakete simbolis T yra rezervuotas, vietoje jo naudojamas n , tačiau čia toliau naudojamas T) pagal (2.1)–(2.3) lygtimis aprašytą modelį su $c = 0$ ir pasirinktomis parametrų reikšmėmis L , σ ir μ . Ši funkcija naudinga eksperimentiniais tiks-lais. Naudotojas taip pat gali naudoti paketą su savo duomenimis, kurie tariamai gali būti paaiškinami vienmatės išvesties grynųjų laiko eilučių duomenų kumuliaciniu struktūrinių lygčių modeliu su adityvinės paklaidos išvesties matavimo lygtimi bei normalu-mo ir nepriklausomumo prielaidomis.

Prieš pereidama prie nurodyto tikslo, funkcija patikrina, ar įvestis yra teisinga tiek struktūros (atitinkamai: $(m + 1) \times k$ matrica dėl L , $(m + 2) \times 1$ matrica dėl σ , $(m + 1) \times 1$ matrica dėl μ ir vieno elemento vektorius dėl T), tiek reikšmių požičiu (visi argumentai

turi būti skaitiniai ir baigtiniai R prasme; T taip pat turi būti teigiamas sveikasis skaičius, o σ – neneigiamas). Jei tikrinimo metu aptinkama klaidų, pateikiamas įspėjimas ir gražinama NULL. Kitu atveju sugeneruoti duomenys gražinami kaip duomenų lentelė, kurioje pirmuosiuose stulpeliuose pateikiami z_x (pradedant nuo antrosios eilutės), o paskutiniame stulpelyje – $(y_0, y_1, \dots, y_T)'$.

2. `estimate_parameters`: pagal 2.1 algoritimą apskaičiuoja didžiausio tikėtinumo Σ , σ_y^2 ir μ įverčius iš duotų duomenų, kurių formatas atitinka `generate_data` funkcijos išvestį, bei nurodomos (2.24) lygtyje naudojamo aukšnio pjūvio paieškos algoritmo konvergencijos tolerancijos (angl. *convergence tolerance*) parametro reikšmės.

Prieš pereidama prie skaičiavimų, funkcija patikrina, ar įvestis yra teisinga tiek struktūros (duomenys turi būti pateikti bent iš dviejų eilučių ir dviejų stulpelių susidedančioje duomenų lentelėje, o konvergencijos tolerancijos parametras – vieno elemento vektorius), tiek reikšmių požiūriu (visi argumentai turi būti skaitiniai ir baigtiniai R prasme (ši taisyklė netaikoma z_x atitinkančiai pirmajai eilutei), o konvergencijos tolerancijos parametras taip pat turi būti teigiamas). Jei tikrinimo metu aptinkama klaidų, pateikiamas įspėjimas ir gražinama NULL. Kitu atveju įverčiai gražinami kaip sąrašas, kurio pirmasis elementas – Σ įvertis kaip matrica, antrasis elementas – σ_y^2 įvertis kaip vektorius, trečiasis elementas – μ įvertis kaip matrica.

3. `evaluate_estimates`: naudojant tikruosius (duomenis generuojančius) Σ , σ_y^2 , μ ir jų įverčius $\hat{\Sigma}$, $\hat{\sigma}_y^2$, $\hat{\mu}$ kartu su jiems gauti naudotu stebėjimų imties dydžiu T , apskaičiuojama ši įverčių tikslumo metrika:

$$\begin{aligned} & \frac{1}{T} \left(\ln(|C_{zz}|) - \ln \left(|\hat{C}_{zz}| \right) + \text{tr} \left(\hat{C}_{zz} C_{zz}^{-1} \right) \right. \\ & \left. + (\hat{\mu}_{\text{vec}} - \mu_{\text{vec}})' C_{zz}^{-1} (\hat{\mu}_{\text{vec}} - \mu_{\text{vec}}) - (m+1)T \right). \end{aligned} \quad (14)$$

(14) metrika iš esmės yra faktorinėje analizėje naudojama didžiausių tikėtinumų skirtumo funkcija [79] su įvestais (2.3)–(2.4) lygtimis apibrėžiamo modelio statistiniais matais. Kadangi

matricos C_{zz} matmenys tiesiškai priklauso nuo T , formulėje įtraukta dalyba iš T .

Jei $\hat{C}_{zz}$ iš tiesų yra kovariacijų matrica (jei ne – įvertis neturi prasmės), (14) metrika yra neneigiama ir lygi nuliui tada, kai $\hat{C}_{zz} = C_{zz}$ ir $\hat{\mu}_{vec} = \mu_{vec}$. Todėl geresnis iš dviejų konkuruojančių įverčių yra tas, kuriam metrikos reikšmė yra mažesnė.

Atsižvelgiant į (2.3)–(2.4) modelio kontekstą, funkcija `evaluate_estimates` vertina tik tokias kovariacijų matricas $\hat{C}_{zz}$, kurioms atitinkamas $\hat{s} \in [0, 1]$. Ši funkcija gali būti naudojama eksperimentiniais tikslais, pavyzdžiui, siekiant nustatyti, kuris iš konkuruojančių parametrų vertinimo algoritmų yra tikslesnis, arba testuojant tą patį algoritmą su skirtingomis duomenų imtimis.

(14) metrikos apskaičiavimas įgyvendinamas panašiai kaip ir 2.1 algoritmo, todėl jos skaičiavimo sudėtingumas yra tiesinis T atžvilgiu.

Prieš pereidama prie skaičiavimų, funkcija patikrina, ar įvestis yra teisinga tiek struktūros (3 elementų ilgio sąrašas, kurio elementai atitinkamai yra: $\Sigma - (m+1) \times (m+1)$ matrica, σ_y^2 – vieno elemento vektorius, $\mu - (m+1) \times 1$ matrica; analogiškas sąrašas dėl $\hat{\Sigma}$, $\hat{\sigma}_y^2$ ir $\hat{\mu}$; bei vieno elemento vektorius T), tiek reikšmių požiūriu (visi argumentai turi būti skaitiniai ir baigtiniai R prasme; T turi būti teigiamas sveikasis skaičius; Σ ir $\hat{\Sigma}$ turi būti simetriškos ir teigiamai apibrėžtos; atitinkami s ir $\hat{s}$ turi priklausyti intervalui $[0, 1]$). Jei patikrinimo metu aptinkama klaidų, pateikiamas išpėjimas ir gražinama `NULL`. Kitu atveju apskaičiuota metrikos reikšmė gražinama kaip vektorius.

4. `calculate_likelihood`: apskaičiuoja (2.11) tikėtinumo funkciją duotiems duomenims (kaip `generate_data` išvestyje) ir parametrams (kaip `estimate_parameters` išvestyje). Ši funkcija yra naudinga tais atvejais, kai reikalinga originalios tikėtinumo funkcijos reikšmė. Skaičiavimai atliekami panašiai kaip 2.1 algoritme, tad šios funkcijos skaičiavimo sudėtingumas yra tiesinis T atžvilgiu.

Prieš pereidama prie skaičiavimų, funkcija atlieka analogiškus

patikrinimus kaip ir funkcijos `estimate_parameters` (duomenims) bei `evaluate_estimates` (parametrams). Jei tikrinimo metu aptinkama klaidų, pateikiamas išpėjimas ir grąžinama `NULL`. Kitu atveju apskaičiuota tikėtinumo funkcijos reikšmė grąžinama kaip vektorius.

5. `estimate_parameters_on`: pagal 3.1 algoritmą apskaičiuoja didžiausio tikėtinumo Σ , σ_y^2 ir μ įverčių seką iš duotų duomenų, kurių formatas atitinka `generate_data` funkcijos išvestį, bei nurodomos pradinės parametro s reikšmės.

Prieš pereidama prie skaičiavimų, funkcija patikrina, ar įvestis yra teisinga tiek struktūros (duomenys turi būti pateikti bent iš dviejų eilučių ir dviejų stulpelių susidedančioje duomenų lentelėje, o parametras s – vieno elemento vektorius), tiek reikšmių požįūriu (visi argumentai turi būti skaitiniai ir baigtiniai R prasme (ši taisyklė netaikoma z_x atitinkančiai pirmajai eilutei), o s parametras taip pat turi priklausyti intervalui $[0, 1]$). Jei tikrinimo metu aptinkama klaidų, pateikiamas išpėjimas ir grąžinama `NULL`. Kitu atveju įverčių seka grąžinama kaip T ilgio sąrašas, kurio kiekvienas elementas yra `estimate_parameters` funkcijos išvesties formato sąrašas (elemento eilės numeris atitinka įvesties duomenų poaibio, sudaryto iš nuoseklių stebėjimų, dydį).

Paketo naudojimas iliustruojamas pavyzdžiu. Kaip ir kiekvieno ne bazinio R paketo atveju, pirmiausia reikia jį įdiegti ir įkelti:

```
> install.packages("EMLI")
> library(EMLI)
```

Nors pavyzdys yra hipotetinis, jame naudojami realūs duomenys. Keletui valiutų porų imamos nuosekliai einančių dienų uždarymo kainos, būtent EURUSD (p^{eurusd}), GBPUSD (p^{gbpusd}) ir USDJPY (p^{usdjpy}). Daroma prielaida, kad (2.3)–(2.4) modelis yra taikomas su $\Delta y_t = \ln p_t^{\text{eurusd}} - \ln p_{t-1}^{\text{eurusd}}$ ir $x_t = \begin{pmatrix} \ln p_t^{\text{gbpusd}} - \ln p_{t-1}^{\text{gbpusd}} \\ \ln p_t^{\text{usdjpy}} - \ln p_{t-1}^{\text{usdjpy}} \end{pmatrix}$, $t = 1, \dots, 503$. Tuomet kreipiamasi į funkciją `estimate_parameters`, siekiant gauti parametrų įverčius pagal 2.1 algoritmą (čia praleidžiami pradiniai duomenų apdorojimo į `rdat` veiksmai):

```
> mle <- estimate_parameters(rdat, 0.00001)
```

Įvertinus šiuos parametrus, iš esmės turimas įvertintas modelis (atsižvelgiant į tai, kad jis yra identifikuojamas). Tačiau išlieka klausimas, ar apskaičiuotas modelis atitinka duomenis. Prieinamos `EMLI` funkcijos leidžia įgyvendinti šiam tikslui skirtą empirinę procedūrą, kuri remiasi duomenų simuliacija ir šiems duomenims apskaičiuotomis tikėtinumo funkcijos reikšmėmis, tokiu būdu įvertinant realių duomenų ir įvertintu modeliu paaiškinamų duomenų panašumą.

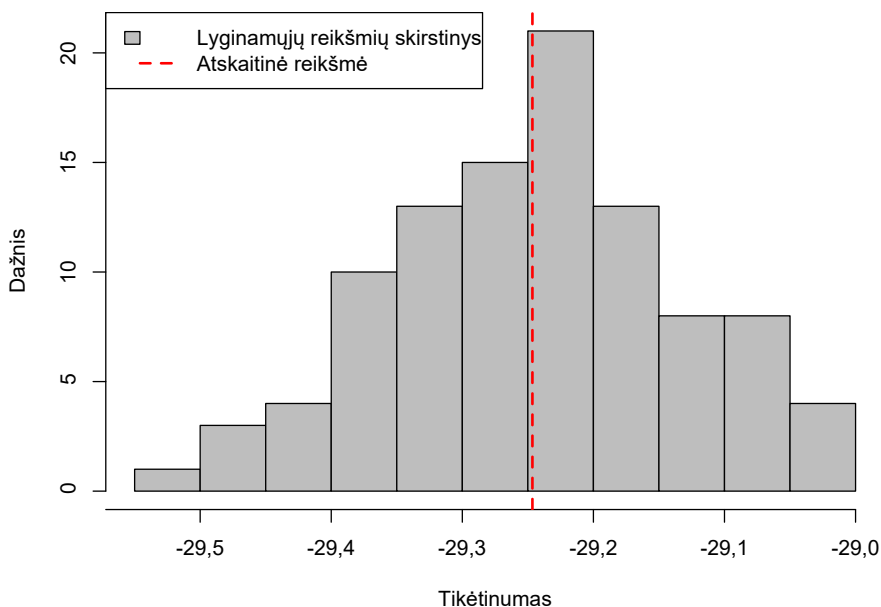
Siekiant atlikti šią procedūrą, pirmiausia apskaičiuojama minimizuotos tikėtinumo funkcijos reikšmė su realiais duomenimis, kuri bus laikoma *atskaitine reikšme*:

```
> ref <- calculate_likelihood(rdat, mle)
```

Tada, pagal įvertintą modelį, sugeneruojama 100 duomenų rinkinių (darant prielaidą, kad yra vienas bendras faktorius, t. y. $k = 1$; čia praleidžiamas pradinių modelio parametrų L , σ ir μ atstatymas) ir apskaičiuojamos atitinkamos tikėtinumo funkcijos reikšmės, kurios bus laikomos *lyginamosiomis reikšmėmis*:

```
> cmps <- numeric()
> T <- nrow(rdat)
> set.seed(999)
> for (i in 1:100) {
>   sdat <- generate_data(T, L, sigma, mu)
>   cmps[i] <- calculate_likelihood(sdat, mle)
> }
```

Gauti rezultatai pavaizduoti A.1 pav. (čia praleistas jį sukuriantis kodas), kuris rodo, kad įvertintas modelis tinkamai apibūdina realius nagrinėtus duomenis, nes atskaitinė reikšmė užtikrintai patenka į lyginamųjų reikšmių skirstinį.



A.1 pav.: Realių duomenų ir pagal įvertintą modelį sugeneruotų duomenų panašumas (matuojamas remiantis tikėtinumo funkcija).

Pateiktas pavyzdys iliustruoja modelio parametrų įvertinimo ir jo tikslumo patikrinimo veiksmus, o tai atskleidžia EMLI paketo daugiafunkčiškumą.

Toliau trumpai apžvelgiamos likusios (naudotojui neprieinamos) EMLI paketo funkcijos:

1. `calculate_recursive`: atsižvelgiant į u , v ir s , funkcija rekursiškai apskaičiuoja c_T , taip pat gražindama f_T ir f_{T+1} kaip šalutinį produktą, siekiant išvengti perskaičiavimų kitose funkcijose. Funkcija naudoja matricinę algebrą ir leidžia u bei v turėti kelis stulpelius, taip leisdama visų rūšių c_T (skaliaro, vektoriaus ir matricos) apskaičiavimą vienu metu.
2. `calculate_selected`: naudodama funkciją `calculate_recursive`, ši funkcija apskaičiuoja $\hat{\Sigma}_{xy}$, $\hat{\mu}_\eta$ ir g pateiktiems duomenims ir s , taip pat gražindama f_T ir f_{T+1} . Skaičiavimai atliekami nuosekliai, jau gautus rezultatus panaudojant vėlesniuose žingsniuose. Kiekvienai unikaliai `calculate_recursive` išvesčiai funkcija iškviečiama tik vieną kartą, o gauti rezultatai, kiek įmanoma, išsaugomi tolesniam naudojimui.

3. `calculate_1D_likelihood`: funkcija apskaičiuoja $\tilde{\mathcal{L}}$ (kaip apibrėžta (2.35)) duotiems duomenims ir s , kreipdamasi į funkciją `calculate_selected`.
4. `minimize_1D_likelihood`: duotiems duomenims ir konvergencijos tolerancijai funkcija vykdo auksinio pjūvio paiešką funkcijai `calculate_1D_likelihood` intervale $[0, 1]$, ieškodama jos minimumo. Funkcija pakartotinai naudoja jau apskaičiuotas funkcijos reikšmes, taip sutaupydama pusę skaičiavimų kiekvienos iteracijos metu, ir konstruoja apribojantį intervalą. Konvergencijos tolerancija naudojama iš anksto nustatyti reikiamą iteracijų skaičių. Gražinama s reikšmė (s^*) yra 0, 1 arba apribojančio intervalo vidurio taškas – pasirenkama ta reikšmė, kuriai funkcijos `calculate_1D_likelihood` reikšmė yra mažiausia.
5. `minimize_likelihood`: duotiems duomenims apskaičiuoja (2.22)–(2.27) nurodyta tvarka, kreipdamasi į funkciją `minimize_1D_likelihood`, o tuomet – į funkciją `calculate_selected` su rastu s^* . Pažymėtina, kad šią funkciją tiesiogiai naudoja funkcija `estimate_parameters`, todėl vietoje σ_y^{2*} ji gražina apskaičiuotą $\frac{\sigma_y^{2*}}{s^*}$ reikšmę, kuri užtikrina veiksmingą Σ_{yy} įvertimo atstatymą.

Apskritai EMLI dizainas pasižymi racionalių kodo organizavimu, kuris pasireiškia specifinėms užduotims skirtomis funkcijomis ir pakartotiniu skaičiavimo rezultatų panaudojimu.

SUMMARY IN ENGLISH

This chapter presents a shortened version of the dissertation in English.

Research Area

The era of big data enables new possibilities for modeling the dynamics of complex natural, engineering, and social systems [91]. However, fully exploiting this potential is inseparable from the challenges related to the ability to efficiently process the growing volume of information [8].

Formally, a dynamic system can be described by a set of rules that define how the system's state changes over time. These rules can be expressed as difference equations for discrete systems or differential equations for continuous systems. The state of a dynamic system at any given time is a point in the space of possible states, and its evolution is a trajectory through this space [34].

A model used to represent a dynamic system defines the possible trajectories, and the values of its parameters specify them in detail (a deterministic model with fixed parameter values describes a specific trajectory, while a stochastic model defines a probabilistic distribution of trajectories [77]). Thus, modeling the evolution of a studied phenomenon is inseparable from the appropriate selection of a model and the accurate estimation of its parameters. This, in turn, requires both extensive system observation data and efficient algorithms that can maximally exploit the information stored in the accumulated data within a practically acceptable time frame. The development of such algorithms is meaningfully addressed by exploring the theoretical formulations of dynamic system models, while model selection and data-related issues are best revealed in the context of specific practical applications.

Research Aim and Objectives

The aim of this work is to define and demonstrate efficient parameter estimation for stochastic dynamic system models and to highlight aspects of their practical application.

To achieve this aim, the following objectives are addressed sequentially:

1. Identification of the conditions and steps for constructing efficient algorithms for parameter estimation in stochastic dynamic system models.
2. Development of an efficient parameter estimation algorithm for a dynamic structural equation model when the dataset is fixed.
3. Transformation of the developed algorithm for parameter estimation with a fixed dataset into a real-time parameter estimation algorithm.
4. Application of stochastic dynamic system models to the modeling of real-world process dynamics.

Research Methods

The objectives of the dissertation are addressed using methods from multivariate statistics, linear algebra, and computational modeling. Efficient parameter estimation algorithms for stochastic dynamic system models are developed using the maximum likelihood method and recursion through data transformation, matrix decomposition, and function restructuring. The developed algorithms are evaluated experimentally using the Monte Carlo method.

Scientific Novelty

The dissertation presents the following results:

1. A stepwise estimation technique is proposed, outlining a methodology for constructing efficient parameter estimation algorithms for stochastic dynamic system models.
2. A cumulative structural equation model is defined, and a maximum likelihood parameter estimation algorithm is developed for the case of a fixed dataset, with computational complexity linear in the number of observation time points.
3. A real-time maximum likelihood parameter estimation algorithm is developed for the cumulative structural equation model.

4. Based on examples of modeling the dynamics of the COVID-19 pandemic and social capital, the selection of stochastic dynamic system models compatible with the studied phenomenon and available data is demonstrated.

Defended Statements

- The stepwise estimation technique outlines the guidelines for constructing efficient algorithms for parameter estimation in stochastic dynamic system models when the dataset is fixed.
- The algorithms developed in this work for parameter estimation in the cumulative structural equation model—both for fixed datasets and in real time—combine the advantages of simpler but less accurate methods with those of more accurate but computationally intensive classical estimation techniques.
- Officially available COVID-19 data lack information linking testing, infection, and mortality at the individual level, making it difficult to unambiguously assess the impact of the COVID-19 passport on the dynamics of the pandemic in Lithuania.

Research Dissemination and Publications

Presentations at international conferences:

1. Efficient Online Estimation of Dynamic Structural Equation Models: Conjecturing on Feasibility. Presented at the 3rd International Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 18–20 September 2024, Almaty (Kazakhstan).
2. On the Gap Between Data and Models in COVID-19 Analysis. Presented at the 33rd European Conference on Operational Research, 30 June – 3 July 2024, Copenhagen (Denmark).
3. Efficient Maximum Likelihood Batch Estimation With Pure Time Series Data of a One-Dimensional Cumulative Structural Equation Model. Presented at the 2nd International Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 21–23 September 2022, Vilnius (Lithuania).

4. Incremental Maximum Likelihood Estimation of Noisy Gaussian Random Walk. Presented at the 31st European Conference on Operational Research, 11–14 July 2021, Athens (Greece).
5. Probabilistic Model of Cultural Participation Impact on Social Capital. Presented at the 1st International & EURO Mini Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies, 18–20 September 2019, Vilnius (Lithuania).

Publications in journals indexed in the *Web of Science (Clarivate Analytics)* database:

1. Vytautas Dulskis & Leonidas Sakalauskas (2025). Toward Efficient Online Estimation of Dynamic Structural Equation Models: A Case Study. *Journal of Statistical Computation and Simulation*, 1–24. DOI: 10.1080/00949655.2025.2515955.
2. Leonidas Sakalauskas, Vytautas Dulskis, & Rimas Jonas Jankūnas (2024). Compartmental Modeling for Pandemic Data Analysis: The Gap Between Statistics and Models. *Heliyon*, 10(11):e31410. DOI: 10.1016/j.heliyon.2024.e31410.
3. Leonidas Sakalauskas, Vytautas Dulskis, & Darius Plikynas (2024). A Technique for Efficient Estimation of Dynamic Structural Equation Models: A Case Study. *Structural Equation Modeling: A Multidisciplinary Journal*, 31(4), 635–650. DOI: 10.1080/10705511.2023.2282378.
4. Leonidas Sakalauskas, Vytautas Dulskis, Rimvydas Laužikas, Arūnas Miliauskas, & Darius Plikynas (2021). A Probabilistic Model of the Impact of Cultural Participation on Social Capital. *The Journal of Mathematical Sociology*, 45(2), 65–78. DOI: 10.1080/0022250X.2020.1725002.

The author’s contribution to each of the listed publications includes the development of core results (models, algorithms), performing computational experiments, and preparing the manuscript.

Structure of the Dissertation

The dissertation consists of an introduction, four main chapters, general conclusions, a list of references (not included in this summary; however,

the original list is cited), and an appendix (also not included in this summary).

The main part of the dissertation, comprising four chapters, presents the research results. Chapter 1 reviews the relevant literature through the lens of the dynamic structural equation model, refines the concept of efficient parameter estimation algorithms, and introduces the technique for their development. Chapter 2 defines the cumulative structural equation model and applies the technique introduced in the first chapter. Chapter 3 describes the steps by which the algorithm for parameter estimation with a fixed dataset, developed in the second chapter, is transformed into a real-time parameter estimation algorithm. Chapter 4 emphasizes the practical aspects of applying dynamic systems models by analyzing the gap between models and data in the context of modeling the spread of the COVID-19 pandemic, as well as by constructing a model of social capital dynamics (the latter is not included in this summary).

S.1. Technique for Constructing Efficient Parameter Estimation Algorithms for Stochastic Dynamic System Models

In this chapter, based on the articles [28, 97], stochastic dynamic system models are represented through a dynamic structural equation modeling framework, which forms the foundation for the subsequent investigation into efficient parameter estimation.

S.1.1. Dynamic Structural Equation Model

The general dynamic (panel) structural equation model (DSEM), encompassing most traditional linear dynamic (and static) models, for subject $i = 1, \dots, N$ at time $t = 1, \dots, T$, is formulated as follows [20]:

$$\eta_{i,t} = \mu_{\eta,i} + \sum_{j=0}^p B_j \eta_{i,t-j} + \sum_{j=0}^q \Gamma_j \xi_{i,t-j} + \zeta_{i,t}, \quad (\text{S.1})$$

$$y_{i,t} = \mu_{y,i} + \Lambda_y \eta_{i,t} + \epsilon_{i,t}, \quad (\text{S.2})$$

$$x_{i,t} = \mu_{x,i} + \Lambda_x \xi_{i,t} + \delta_{i,t}, \quad (\text{S.3})$$

where $\eta_{i,t} = \left(\eta_{i,t}^{(1)}, \dots, \eta_{i,t}^{(l)} \right)'$ and $\xi_{i,t} = \left(\xi_{i,t}^{(1)}, \dots, \xi_{i,t}^{(k)} \right)'$ are vectors of unobserved (latent) variables; $y_{i,t} = \left(y_{i,t}^{(1)}, \dots, y_{i,t}^{(n)} \right)'$ and $x_{i,t} =$

$(x_{i,t}^{(1)}, \dots, x_{i,t}^{(m)})'$ are vectors of observed variables; $\mu_{\eta,i}$ ($l \times 1$), $\mu_{y,i}$ ($n \times 1$), and $\mu_{x,i}$ ($m \times 1$) are vectors of fixed effects; B_j ($l \times l$), Γ_j ($l \times k$), Λ_y ($n \times l$), and Λ_x ($m \times k$) are matrices of regression coefficients; $\zeta_{i,t}$ ($l \times 1$), $\epsilon_{i,t}$ ($n \times 1$), and $\delta_{i,t}$ ($m \times 1$) are vectors of measurement errors (noise).

Parameter estimation for the DSEM in the case of $N = 1$ (known as pure time series data) can be performed within the conventional covariance structure analysis framework by constructing an artificial covariance matrix, where time-lagged values of time series vectors are treated as additional variables (the so-called Toeplitz method) [46, 130]. Although this method is not complex, it violates the assumption of independent cases (rows), and therefore yields moment-based estimates rather than maximum likelihood estimates.

As an alternative, one may consider reformulating the model in state-space form and using the Kalman filter for recursive evaluation of the model's likelihood function, where new parameter estimates are obtained at each step until convergence [104, 113, 130]. While Kalman filter-based estimation yields maximum likelihood estimates, it requires analytical derivatives and good initial values. Moreover, the recursive nature of this estimation contributes to slow convergence, especially in large models, since each estimation step involves Kalman filter recursions.

The classical method for obtaining maximum likelihood estimates is direct maximum likelihood estimation, where the likelihood function is written for the entire data sample [20, 25, 47, 118]. However, since computing the likelihood function requires inverting a matrix of at least size $T \times T$, which is computationally intensive (with algorithmic complexity cubic in T) [85], traditional procedures for likelihood optimization are practical only for small datasets. Another potential issue is the high dimensionality of the model, $n + m$, but this work focuses exclusively on challenges posed by data volume.

When facing computational difficulties, especially with more complex models, another frequently used method is Bayesian estimation with Gibbs (or other) sampling [75, 130]. Although Bayesian estimation is relatively easy to implement for complex distributions in high-dimensional spaces using Markov Chain Monte Carlo methods, which allow working with intricate modeling scenarios, this method also has drawbacks—among them, difficulties in assessing accuracy and conver-

gence (even empirically) [106].

S.1.2. Efficient Parameter Estimation Algorithms

Given the existing possibilities for parameter estimation in DSEM using pure time series data, it can be summarized that current methods either produce low-quality estimates or are computationally inefficient. Some methods rely on various simplifications, making them easy to compute but suboptimal—in the sense that other estimators may be more concentrated around the true parameter value. Other methods are based on a solid theoretical foundation, such as maximum likelihood estimation [80], and can be at least asymptotically optimal, but they require substantial computational resources.

Although developing an efficient algorithm for a given model is often not a trivial task, it is possible to formulate general principles for such a process. For simplicity, we will refer to these principles as the stepwise estimation technique (SET).

SET is related to the computationally efficient application of the maximum likelihood method (MLM)—as opposed to transforming a computationally efficient but statistically suboptimal method into an optimal one, or developing a fundamentally new method.

Since MLM is used for parameter estimation, the object of SET is the likelihood function, formulated for possibly transformed data (according to the initial part of SET). Its processing is then carried out in three stages (according to the operational part of SET), the successful implementation of which enables the construction of computationally efficient maximum likelihood parameter estimation algorithms.

Initial part of SET. The initial DSEM likelihood function is a function of model parameters associated with the mean vector and the covariance matrix, which has a block structure [20]. The goal of the initial part of SET is to transform the model data in such a way that the blocks of the covariance matrix are sparsified, converting them into multidagonal matrices. This is pursued because sparse matrices are easier to process than dense ones, and multidagonal (or, more broadly, banded) matrices form an important and relatively well-studied subset of sparse matrices [27].

Operational part of SET. The operational part of SET consists of three sequential steps that define the necessary actions:

1. Block decomposition of the covariance matrix obtained in the initial part of SET. This step replaces the original likelihood function, which depends on the model parameters, with an equivalent reparameterized likelihood function. This new function depends on the unique elements of the mean vector and the covariance matrix (possibly transformed to facilitate further processing), hereafter referred to simply as parameters. The resulting likelihood function involves a covariance matrix substitute—at least a $T \times T$ matrix that must be inverted and whose determinant must be computed—that depends on a minimal number of parameters.
2. Differentiation of the equivalent likelihood function concerning the parameters that are not part of the covariance matrix substitute, followed by manipulation of the resulting preliminary estimators to obtain explicit estimators that depend only on the parameters forming the covariance matrix substitute. Substituting these explicit estimators into the equivalent likelihood function yields a function that depends solely on the parameters of the covariance matrix substitute, thereby reducing the dimensionality of the optimization problem.
3. Derivation of efficient recursive formulas, each step of which requires a constant number of elementary operations. These formulas are necessary for computing both the complex terms related to the covariance matrix substitute and the obtained estimators in time proportional to the sample size T .

Due to its nature, SET is best developed and applied using a bottom-up approach—that is, starting from specificity and moving toward generality [111]. This is essentially because the more complex the model, the more complex its covariance structure, making the SET stages more demanding in terms of the required mathematical apparatus and depth of insight.

Nevertheless, the successful application of SET to a specific model allows the acquired experience to be extrapolated in a selective small

step toward the final general model (e.g., the one described by Equations (S.1)–(S.3)). This process can be repeated, thereby realizing the bottom-up approach.

S.1.3. Real-Time Parameter Estimation

Traditionally, DSEM research has focused on parameter estimation using a fixed dataset [20, 44]. This means that all relevant data are collected first, and then the parameters are estimated—typically concerning the entire dataset at once. By definition, this approach separates data collection and parameter estimation into two distinct research stages. Less obviously, it may also imply that the relationships between variables are stable over time, which is often a reasonable assumption.

However, it may also be the case that relationships between variables change over time, or that the data stream is continuous (possibly without a defined endpoint), and decisions influenced by model parameter estimates must be made at different time points based on the data collected up to those points [120]. In such cases, real-time parameter estimation is required, involving the updating of parameter estimates as new data become available [35]. Real-time estimation allows for the progressive refinement of parameter estimates as the data volume increases and enables adaptation to changes in system dynamics—particularly relevant in non-stationary environments where the underlying data-generating process may evolve.

Although the concept of real-time parameter estimation has existed for decades, with numerous algorithms proposed across various fields, progress in DSEM in this regard remains relatively limited. This may partly be due to the development challenges involved in designing real-time estimation algorithms that do not sacrifice estimation quality for computational speed. On the other hand, there may still be a lack of broader recognition of the potential and importance of real-time estimation in the era of big data, which has limited the effort devoted to advancing this area. Nevertheless, for DSEM to remain relevant in the modern context, efficient real-time parameter estimation algorithms are essential, and their development, according to SET, should be based on MLM and recursion.

S.1.4. Section Conclusions

- DSEM is an important linear stochastic dynamic system model that provides a contextual framework for exploring the topic of efficient parameter estimation.
- Existing DSEM parameter estimation algorithms typically offer either high estimation quality or high computational speed, but fully leveraging modern conditions requires algorithms that possess both properties.
- SET defines a sequence of steps whose successful execution enables the development of efficient algorithms for parameter estimation in DSEM and more general stochastic dynamic system models when the dataset is fixed.
- Real-time parameter estimation for DSEM has not received sufficient attention in the era of big data, but the development of efficient algorithms for this purpose can be pursued as an extension of SET.

S.2. Application of the Stepwise Estimation Technique to the Cumulative Structural Equation Model

This section, based on the article [97], demonstrates—or, considering its bottom-up application, initiates—SET using a relatively simple yet fundamental case of the model defined by Equations (S.1)–(S.3), where $B_j = 0_{l \times l}$ ($j \neq 1$), $B_1 = I_{l \times l}$, and $\Gamma_j = 0_{l \times k}$ ($j > 0$). The resulting model is referred to as the cumulative structural equation model (CSEM). CSEM is a special case of DSEM, obtained by constraining only the structural equation, although other special cases of CSEM (and thus DSEM) can be derived by constraining the measurement equations. From a practical application perspective, CSEM adheres to the principle of Occam’s razor [107], balancing parsimony, accuracy, and adequacy. This model includes latent variables measured by a factor output model, with their increments regressed on other modeled latent variables measured by a factor input model [18]. Given that $B_1 = I_{l \times l}$, the process modeled by CSEM resembles a multivariate, noise-observed Wiener

process, which is known to be non-stationary and fundamental in both theoretical and applied mathematics [52, 128].

S.2.1. Model

The model under consideration is defined by three related equations.

The first equation is the transition (structural) equation:

$$\eta_t = \eta_{t-1} + \mu_\eta + \Gamma_0 \xi_t + \zeta_t, \quad \eta_0 = c, \quad (\text{S.4})$$

where $\{\eta_0\} \cup \{\eta_t\}$ is a sequence of scalar latent states (c is the scalar initial value), μ_η is a scalar fixed effect, Γ_0 is a $1 \times k$ vector of weights from latent input to the latent process, $\{\xi_t\}$ is a sequence of independent and identically distributed $k \times 1$ latent input vectors (common factors), with $\xi_t \sim \mathcal{N}(0_{k \times 1}, I_{k \times k})$, and $\{\zeta_t\}$ is a sequence of independent and identically distributed scalar latent process noise terms $\mathcal{N}(0, \sigma_\eta^2)$ ($t = 1, \dots, T$).

The second equation is the output measurement equation:

$$y_t = \eta_t + \epsilon_t, \quad (\text{S.5})$$

where $\{y_t\}$ is a sequence of observed scalar outputs, and $\{\epsilon_t\}$ is a sequence of independent and identically distributed scalar output noise terms $\mathcal{N}(0, \sigma_y^2)$ ($t = 0, \dots, T$).

The third equation is the input measurement equation:

$$x_t = \mu_x + \Lambda_x \xi_t + \delta_t, \quad (\text{S.6})$$

where $\{x_t\}$ is a sequence of observed $m \times 1$ input vectors, μ_x is an $m \times 1$ vector of fixed effects, Λ_x is an $m \times k$ matrix of common factor loadings, and $\{\delta_t\}$ is a sequence of independent and identically distributed $m \times 1$ specific factor vectors, with $\delta_t \sim \mathcal{N}(0_{m \times 1}, \text{diag}(\sigma_{x_1}^2, \dots, \sigma_{x_m}^2))$ ($t = 1, \dots, T$).

It is assumed that $\{\xi_t\}$, $\{\zeta_t\}$, $\{\delta_t\}$, and $\{\epsilon_0\} \cup \{\epsilon_t\}$, $t = 1, \dots, T$, are mutually independent.

Instead of using the directly observed output y_t , $t = 0, \dots, T$, we use its differences between adjacent time points, which introduces sparsity into the covariance matrix, giving it a tridiagonal structure:

$$\begin{aligned} \Delta y_t &= y_t - y_{t-1} \\ &= \mu_\eta + \Gamma_0 \xi_t + \zeta_t + (\epsilon_t - \epsilon_{t-1}), \end{aligned} \quad (\text{S.7})$$

$t = 1, \dots, T$.

By appending μ_η to the end of μ_x and Γ_0 to the end of Λ_x , the resulting vector and matrix are denoted as μ and L , respectively (also, for later use, we define $\sigma = (\sigma_{x_1}, \dots, \sigma_{x_m}, \sigma_\eta, \sigma_y)'$).

Similarly, by appending Δy_t to the end of x_t , the resulting vector is denoted as z_t , and $z = (z_1, \dots, z_T)'$. Then

$$Ez_t = \mu,$$

$t = 1, \dots, T$, and

$$C_{zz}(\tau) = \begin{cases} \Sigma, & \tau = 0 \\ -\Psi_1, & \tau = 1 \\ 0_{(m+1) \times (m+1)}, & \text{otherwise} \end{cases},$$

where $\Sigma = LL' + \Psi_0$, and

$$\Psi_0 = \begin{bmatrix} \sigma_{x_1}^2 & 0 & \dots & 0 & 0 \\ 0 & \sigma_{x_2}^2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & \sigma_{x_m}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\eta^2 + 2\sigma_y^2 \end{bmatrix}, \quad \Psi_1 = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_y^2 \end{bmatrix}.$$

We consider the computationally efficient maximum likelihood estimation of μ , Σ , and σ_y^2 (i.e., the theoretical mean vector and non-zero elements of the covariance matrix of the data modeled by Equations (S.6)–(S.7)) given a fixed dataset. The goal is to develop an algorithm based on MLM that provides maximum likelihood estimates of these quantities and is linear in the sample size T .

S.2.2. Likelihood Function

We define:

$$I = I_{T \times T}, \quad J = J_{T \times T} = \begin{cases} -1, & |i - j| = 1 \\ 0, & \text{otherwise} \end{cases},$$

where $i = 1, \dots, T, j = 1, \dots, T$.

The full covariance matrix is composed of $(m + 1) \times (m + 1)$ blocks, each containing $T \times T$ elements:

$$C_{zz} = \Sigma \otimes I + \Psi_1 \otimes J.$$

After vectorizing the observation data and their mean matrices, we denote $z_{\text{vec}} = \text{vec}(z)$ and $\mu_{\text{vec}} = \text{vec}(\underbrace{(\mu, \dots, \mu)'}_T)$. It is easy to see that z_{vec} follows an $(m+1)T$ -dimensional Gaussian distribution, and its probability density function is:

$$p_{z_{\text{vec}}}(z_{\text{vec}}) = \frac{e^{-\frac{1}{2}d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}}}}{(2\pi)^{\frac{(m+1)T}{2}}|C_{zz}|^{1/2}}, \quad (\text{S.8})$$

where $d_{\text{vec}} = z_{\text{vec}} - \mu_{\text{vec}}$ (note that no distinction is made between the random vector and the argument of the density function).

Taking the logarithm of Equation (S.8), we obtain the log-likelihood function:

$$\begin{aligned} \mathcal{L}(\theta) &= \mathcal{L}(\theta; z_{\text{vec}}) \\ &= -\frac{1}{2} \left(\ln(|C_{zz}|) + d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}} + (m+1)T \ln(2\pi) \right), \end{aligned}$$

where θ collectively denotes the model parameters (all of which are contained in μ_{vec} and C_{zz}).

Since the following equality holds:

$$\begin{aligned} \arg \max_{\theta} [\mathcal{L}(\theta)] &= \arg \max_{\theta} \left[\frac{\mathcal{L}(\theta)}{T} \right] \\ &= \arg \min_{\theta} \left[\frac{\ln(|C_{zz}|) + d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}}}{T} \right], \end{aligned}$$

we proceed by minimizing the normalized and simplified function $\mathcal{L}(\theta)$, which has the following form:

$$\mathcal{L}(\theta) = \frac{\ln(|C_{zz}|) + d'_{\text{vec}}C_{zz}^{-1}d_{\text{vec}}}{T}. \quad (\text{S.9})$$

Instead of the original model parameters, we use partially transformed parameters, i.e., we define $\theta = (s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta)$, where

$$\begin{aligned} s &= e^{-\text{acosh}\left(\frac{1}{2} \frac{1}{\sigma_y^2 \Sigma_l}\right)}, \\ \Sigma_{xx} &= \Lambda_x \Lambda'_x + \text{diag}(\sigma_{x_1}^2, \dots, \sigma_{x_m}^2), \\ \tilde{\Sigma}_{xy} &= \Sigma_{xx}^{-1} \Sigma_{xy}, \end{aligned}$$

here $\Sigma_l = [\Sigma^{-1}]_{m+1, m+1}$ and $\Sigma_{xy} = [\Sigma]_{1:m, m+1}$. Essentially, $\tilde{\Sigma}_{xy}$ and s are surrogate parameters replacing the direct use of Σ_{xy} and $\Sigma_{yy} =$

$[\Sigma]_{m+1,m+1}$ (Σ_{xx} , Σ_{xy} , and Σ_{yy} form the matrix Σ to be estimated). Note that $s \in [0, 1]$ and describes the relative weight of the output measurement noise σ_y^2 in the model—larger values indicate greater weight.

Although Σ_{xy} does not belong to the chosen θ , its retrieval is straightforward:

$$\Sigma_{xy} = \Sigma_{xx} \tilde{\Sigma}_{xy}. \quad (\text{S.10})$$

Accordingly, Σ_{yy} has the following expression:

$$\Sigma_{yy} = \frac{\sigma_y^2}{s} (s^2 + 1) + (\Sigma_{xy})' \tilde{\Sigma}_{xy}. \quad (\text{S.11})$$

With θ defined as above, the likelihood function defined by Equation (S.9) can be reformulated as:

$$\begin{aligned} \mathcal{L} \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta \right) = & \frac{1}{T} \left(\ln \left((|\Sigma_{xx}| \sigma_y^2)^T |Q| \right) \right. \\ & + \text{tr} \left(d_x \Sigma_{xx}^{-1} d_x' \right) \\ & \left. + \frac{1}{\sigma_y^2} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right) \right) \right), \end{aligned} \quad (\text{S.12})$$

where $d_x = z_x - \mathbf{1}\mu_x'$, $d_y = z_y - \mathbf{1}\mu_\eta'$, and $Q = \left(\frac{s^2+1}{s} \right) I + J$, with $z_x = (x_1, \dots, x_T)'$, $z_y = (\Delta y_1, \dots, \Delta y_T)'$, and $\mathbf{1} = \mathbf{1}_{T \times 1}$.

Essentially, the likelihood function defined by Equation (S.12) is used to estimate the theoretical statistical measures of the model defined by Equations (S.6)–(S.7)—specifically, μ and the non-zero elements of C_{zz} .

S.2.3. Partial Derivatives

We now need to compute the derivatives $\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}}$, $\frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}}$, $\frac{\partial \mathcal{L}}{\partial \sigma_y^2}$, $\frac{\partial \mathcal{L}}{\partial \mu_x}$, $\frac{\partial \mathcal{L}}{\partial \mu_\eta}$, and solve the equations $\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}} = 0_{m \times m}$, $\frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}} = 0_{m \times 1}$, $\frac{\partial \mathcal{L}}{\partial \sigma_y^2} = 0$, $\frac{\partial \mathcal{L}}{\partial \mu_x} = 0_{m \times 1}$, $\frac{\partial \mathcal{L}}{\partial \mu_\eta} = 0$, to obtain the initial estimators: $\hat{\Sigma}_{xx}$, $\hat{\tilde{\Sigma}}_{xy}$, $\hat{\sigma}_y^2$, $\hat{\mu}_x$, $\hat{\mu}_\eta$.

After performing elementary but extensive calculations (omitted here for simplicity), we obtain the following expressions:

$$\frac{\partial \mathcal{L}}{\partial \Sigma_{xx}} = \Sigma_{xx}^{-1} - \frac{\Sigma_{xx}^{-1} d_x d_x' \Sigma_{xx}^{-1}}{T} = 0_{m \times m} \implies \quad (\text{S.13})$$

$$\hat{\Sigma}_{xx} = \frac{d'_x d_x}{T}, \quad (\text{S.14})$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{\Sigma}_{xy}} = \frac{2}{T\sigma_y^2} \left(d'_x Q^{-1} d_x \tilde{\Sigma}_{xy} - d'_x Q^{-1} d_y \right) = 0_{m \times 1} \implies \quad (\text{S.15})$$

$$\hat{\Sigma}_{xy} = (d'_x Q^{-1} d_x)^{-1} d'_x Q^{-1} d_y, \quad (\text{S.16})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma_y^2} = \frac{1}{\sigma_y^2} - \frac{1}{T} \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{(\sigma_y^2)^2} = 0 \implies \quad (\text{S.17})$$

$$\hat{\sigma}_y^2 = \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{T}, \quad (\text{S.18})$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mu_x} = \frac{2}{T\sigma_y^2} & \left(\left(\mathbf{1}' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \tilde{\Sigma}'_{xy} + \sigma_y^2 T \Sigma_{xx}^{-1} \right) \mu_x \right. \\ & \left. + \left(d_y - z_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} - \sigma_y^2 \Sigma_{xx}^{-1} z'_x \mathbf{1} \right) = 0_{m \times 1} \implies \end{aligned} \quad (\text{S.19})$$

$$\begin{aligned} \hat{\mu}_x = & \left(\mathbf{1}' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \tilde{\Sigma}'_{xy} + \sigma_y^2 T \Sigma_{xx}^{-1} \right)^{-1} \left(\sigma_y^2 \Sigma_{xx}^{-1} z'_x \mathbf{1} \right. \\ & \left. - \left(d_y - z_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1} \tilde{\Sigma}_{xy} \right), \end{aligned} \quad (\text{S.20})$$

$$\frac{\partial \mathcal{L}}{\partial \mu_\eta} = \frac{2}{T\sigma_y^2} \left(d_x \tilde{\Sigma}_{xy} - d_y \right)' Q^{-1} \mathbf{1} = 0 \implies \quad (\text{S.21})$$

$$\hat{\mu}_\eta = \frac{\left(z_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}. \quad (\text{S.22})$$

It is important to note that the obtained initial estimators are highly interdependent, i.e., $\hat{\Sigma}_{xx} = \hat{\Sigma}_{xx}(\mu_x)$, $\hat{\Sigma}_{xy} = \hat{\Sigma}_{xy}(s, \mu_x, \mu_\eta)$, $\hat{\sigma}_y^2 = \hat{\sigma}_y^2(s, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)$, $\hat{\mu}_x = \hat{\mu}_x(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_\eta)$, $\hat{\mu}_\eta = \hat{\mu}_\eta(s, \tilde{\Sigma}_{xy}, \mu_x)$.

However, the ultimate goal is to obtain such $\hat{\Sigma}_{xx}$, $\hat{\Sigma}_{xy}$, $\hat{\sigma}_y^2$, $\hat{\mu}_x$, $\hat{\mu}_\eta$ that enable one-off computations depending on the parameter s , whose optimal value (i.e., estimate) is obtained by minimizing the now univariate likelihood function $\tilde{\mathcal{L}}(s)$, thereby reducing the overall problem of minimizing the full function $\mathcal{L}$ concerning all its variables, yielding the optimal θ (denoted θ^*).

S.2.4. Maximum Likelihood Estimators

By manipulating the derivatives and the resulting estimators defined by Equations (S.13)–(S.22), we can construct formulas for sequentially

obtaining the components of θ^* :

$$\mu_x^* = \hat{\mu}_x, \quad (\text{S.23})$$

$$\Sigma_{xx}^* = \hat{\Sigma}_{xx}, \quad (\text{S.24})$$

$$s^* = \arg \min_{s \in [0,1]} \tilde{\mathcal{L}}(s), \quad (\text{S.25})$$

$$\tilde{\Sigma}_{xy}^* = \hat{\tilde{\Sigma}}_{xy}(s^*), \quad (\text{S.26})$$

$$\mu_\eta^* = \hat{\mu}_\eta(s^*), \quad (\text{S.27})$$

$$\sigma_y^{2*} = \hat{\sigma}_y^2(s^*), \quad (\text{S.28})$$

where

$$\hat{\mu}_x = \frac{z'_x \mathbf{1}}{T}, \quad (\text{S.29})$$

$$\hat{\Sigma}_{xx} = \frac{\hat{d}'_x \hat{d}_x}{T}, \quad (\text{S.30})$$

$$\hat{\tilde{\Sigma}}_{xy} = \left(I_{m \times m} - \frac{(\hat{d}'_x Q^{-1} \hat{d}_x)^{-1} \hat{d}'_x Q^{-1} \mathbf{1} (\hat{d}'_x Q^{-1} \mathbf{1})'}{\mathbf{1}' Q^{-1} \mathbf{1}} \right)^{-1} \quad (\text{S.31})$$

$$(\hat{d}'_x Q^{-1} \hat{d}_x)^{-1} \hat{d}'_x Q^{-1} \left(z_y - \frac{z'_y Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}} \mathbf{1} \right),$$

$$\hat{\mu}_\eta = \frac{(z_y - \hat{d}_x \hat{\tilde{\Sigma}}_{xy})' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}, \quad (\text{S.32})$$

$$\hat{\sigma}_y^2 = \frac{(\hat{d}_y - \hat{d}_x \hat{\tilde{\Sigma}}_{xy})' Q^{-1} (\hat{d}_y - \hat{d}_x \hat{\tilde{\Sigma}}_{xy})}{T}, \quad (\text{S.33})$$

$$\tilde{\mathcal{L}} = \frac{\ln(|Q|)}{T} + \ln(\hat{\sigma}_y^2), \quad (\text{S.34})$$

where $\hat{d}_x = z_x - \mathbf{1} \hat{\mu}'_x$ and $\hat{d}_y = z_y - \mathbf{1} \hat{\mu}_\eta$.

This section focuses exclusively on minimizing the function $\tilde{\mathcal{L}}$ using numerical methods that do not require derivative information. For this purpose, the golden section search is used, which is one such method [90].

To achieve the desired computational efficiency, it is necessary to perform the calculations in Equations (S.29)–(S.34) in time proportional to T (if derivative-based minimization methods were used, it would also be necessary to compute the derivative of $\tilde{\mathcal{L}}$ efficiently).

S.2.5. Recursive Computations

The main obstacle to efficiently computing θ^* according to Equations (S.23)–(S.28) is the need to invert the $T \times T$ matrix Q , which appears in expressions of the form $u'Q^{-1}v$. Here, u and v may be $T \times 1$ vectors or $T \times m$ matrices.

For simplicity, here all computations are performed assuming u and v are $T \times 1$ vectors. In this case, $u'Q^{-1}v$ is a scalar, and the vector or matrix cases consist of such scalar elements.

Compared to direct matrix operations, computational efficiency can be immediately improved by analytically expressing the elements of Q^{-1} [51].

Further improvements, enabling the desired linear complexity concerning T , can be achieved by expressing $u'Q^{-1}v$ recursively, where each iteration requires a constant number of elementary operations (i.e., constant time).

In addition to time complexity, it is also important that all expressions used are well-defined and numerically stable.

A key quantity here is $c_T = \frac{1}{T} \frac{([u]_{1:T})'([Q]_{1:T,1:T})^{-1}[v]_{1:T}}{q_T}$, where $q_T = \left[([Q]_{1:T,1:T})^{-1}\right]_{T,T}$ is used for the case $s = 0$, and division by T is used for normalization.

Thus, instead of computing terms of the form $u'Q^{-1}v$ in the estimators defined by Equations (S.31)–(S.34), we compute terms of the form c_T , since $\frac{1}{T} \frac{1}{q_T}$ cancels out in the estimators $\hat{\Sigma}_{xy}$ and $\hat{\mu}_\eta$, and using c_T in the estimators $\hat{\sigma}_y^2$ and $\tilde{\mathcal{L}}$ enables robust expressions, as they are then computed as:

$$\hat{\sigma}_y^2 = q_T g = s \frac{f_T}{f_{T+1}} g, \quad (\text{S.35})$$

$$\tilde{\mathcal{L}} = h + \ln(g), \quad (\text{S.36})$$

here

$$h = \frac{\ln(|Q|)}{T} + \ln(q_T) = \ln\left(\frac{f_T}{(f_{T+1})^{1-\frac{1}{T}}}\right), \quad (\text{S.37})$$

$$g = \frac{\hat{\sigma}_y^2}{q_T}, \quad (\text{S.38})$$

where g is c_T with $u = v = \hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy}$, and $f_T = \sum_{i=0}^{T-1} s^{2i}$ is computed recursively:

$$f_T = f_{T-1} + s^{2(T-1)}. \quad (\text{S.39})$$

In turn, the quantity c_T has the following recursive expression:

$$c_T = \left(1 - \frac{1}{T}\right) \frac{f_{T-1}f_{T+1}}{f_T^2} c_{T-1} + \frac{a_T b_T}{T}, \quad (\text{S.40})$$

where $a_T = \frac{([u]_{1:T})'[(Q)_{1:T,1:T}]^{-1}}{q_T}$ and $b_T = \frac{([v]_{1:T})'[(Q)_{1:T,1:T}]^{-1}}{q_T}$ are also computed recursively:

$$a_T = s \frac{f_{T-1}}{f_T} a_{T-1} + [u]_T, \quad b_T = s \frac{f_{T-1}}{f_T} b_{T-1} + [v]_T. \quad (\text{S.41})$$

The initial values for all recursions, namely a_1 , b_1 , c_1 , and f_1 , are:

$$a_1 = [u]_1, \quad b_1 = [v]_1, \quad c_1 = [u]_1, [v]_1, \quad f_1 = 1. \quad (\text{S.42})$$

Equations (S.35)–(S.42) enable the computation of θ^* in time linear in the sample size T .

Finally, based on Equation (S.35), s cancels out in the term $\frac{\hat{\sigma}_y^2}{s}$, which is needed to compute the estimate of Σ_{yy} (see Equation (S.11)), yielding:

$$\frac{\hat{\sigma}_y^2}{s} = \frac{f_T}{f_{T+1}} g. \quad (\text{S.43})$$

S.2.6. Estimation Algorithm

The obtained results are summarized in the following pseudocode, which is the main goal of this section.

Algorithm S.1 Computationally efficient maximum likelihood parameter estimation for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence, given a fixed dataset.

Input: Observed data sample z_t , $t = 1, \dots, T$

Output: Estimates $\hat{\mu}$, $\hat{\Sigma}$, $\hat{\sigma}_y^2$

- 1: Compute $\hat{\mu}_x$ and $\hat{\Sigma}_{xx}$ according to Equations (S.29)–(S.30)
 - 2: Find $\hat{s}$ by solving $\arg \min_{s \in [0,1]} \tilde{\mathcal{L}}(s)$ using a derivative-free method on Equation (S.36), with each term of the form $u'Q^{-1}v$ replaced by c_T , and each c_T computed via simultaneous evaluation (i.e., as contained in a single `for` loop) of Equations (S.39)–(S.42)
-

-
- 3: Compute $\hat{\Sigma}_{xy}(\hat{s})$, $\hat{\mu}_\eta(\hat{s})$, and $\frac{\hat{\sigma}_y^2(\hat{s})}{\hat{s}}$ according to Equations (S.31)–(S.32) and Equation (S.43), with each $u'Q^{-1}v$ term replaced by c_T , and each c_T computed via simultaneous evaluation of Equations (S.39)–(S.42)
 - 4: Compute $\hat{\Sigma}_{xy}$ using Equation (S.10) (with $\hat{\Sigma}_{xx}$ and $\hat{\Sigma}_{xy}$), and compute $\hat{\Sigma}_{yy}$ using Equation (S.11) (with $\hat{s}$, $\hat{\Sigma}_{xy}$, $\hat{\Sigma}_{xy}$, and $\frac{\hat{\sigma}_y^2}{\hat{s}}$)
 - 5: Construct $\hat{\mu}$ from $\hat{\mu}_x$ and $\hat{\mu}_\eta$; construct $\hat{\Sigma}$ from $\hat{\Sigma}_{xx}$, $\hat{\Sigma}_{xy}$, and $\hat{\Sigma}_{yy}$; assign $\hat{\sigma}_y^2$ the result of $\hat{s} \cdot \frac{\hat{\sigma}_y^2}{\hat{s}}$
-

S.2.7. Empirical Results

Algorithm S.1 was verified empirically. To this end, several experiments were conducted with the goal of:

- verifying the convergence of its estimates as T increases;
- comparing its estimates with those of the traditional MLM (i.e., obtained by directly minimizing the likelihood function defined by Equation (S.9)) and empirical estimates (i.e., using the inherent method for mean and covariance estimation);
- comparing its execution time with the execution times of algorithms implementing these competing methods.

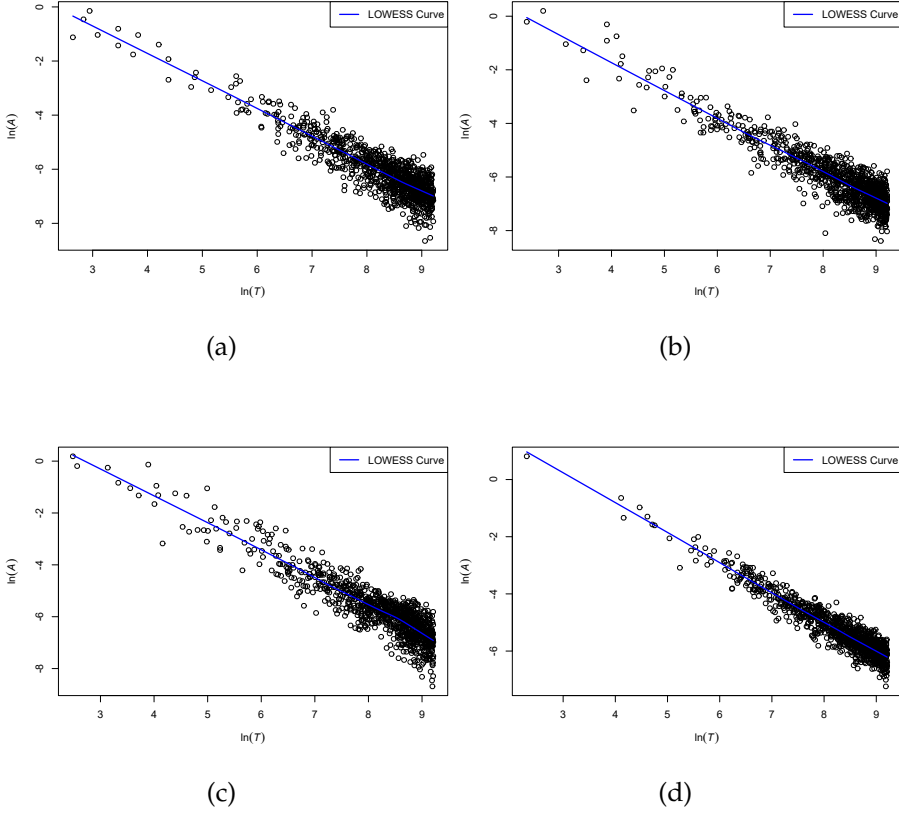
The experiments were conducted using the R statistical package [92], specifically the `EMLI` package [29], as well as custom code for interfacing with `EMLI` functions, introducing competing algorithms, and presenting results.

In the convergence experiments, specific values of m , k , L , σ , and μ were first fixed. Then, using the `generate_data` function from the `EMLI` package, 1,000 data samples were generated, with sizes denoted by T_i , $i = 1, \dots, 1,000$, where each T_i was assigned the result of the operation `as.integer(runif(1, 10, 10000))`.

The `estimate_parameters` function from the `EMLI` package was applied to these data samples to obtain parameter estimates, using a convergence tolerance threshold of 10^{-5} . The outputs were then passed to the `evaluate_estimates` function from the `EMLI` package to assess the accuracy of the estimates. The output of this function for a sample of size T_i is denoted by A_i , $i = 1, \dots, 1,000$.

For each fixed parameter set m , k , L , σ , and μ , a scatter plot was created where each point corresponds to a pair $(\ln(T_i), \ln(A_i))$. In all scatter plots, a blue LOWESS curve [21] was also drawn to smooth the results of individual experiments.

In total, experiments were conducted with six different parameter sets m , k , L , σ , and μ . The results, along with selected model-related values, are presented in Figure S.1 as well as in the accompanying information.



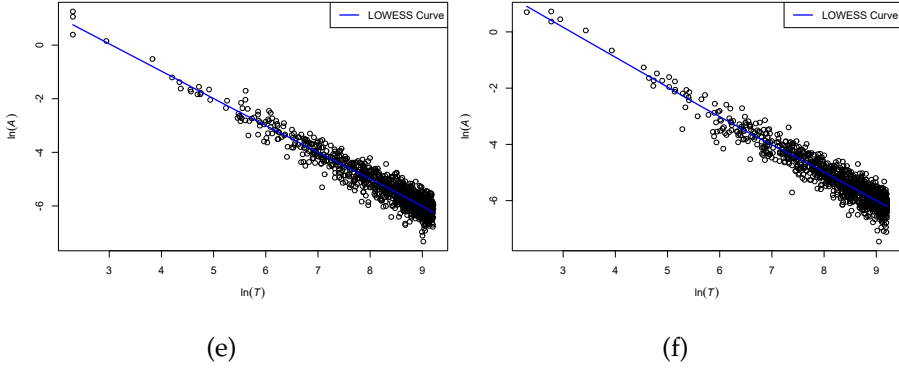


Figure S.1: Verification of the convergence of Algorithm S.1 under different model parameters: (a) parameter set No. 1; (b) parameter set No. 2; (c) parameter set No. 3; (d) parameter set No. 4; (e) parameter set No. 5; (f) parameter set No. 6.

The parameter sets used in the convergence experiments are listed below:

$$\begin{aligned}
 1. \quad m = 2, k = 1, L = \begin{pmatrix} 5.784996 \\ 4.489983 \\ -9.822689 \end{pmatrix}, \sigma = \begin{pmatrix} 5.7348710 \\ 7.3034459 \\ 4.2458612 \\ 0.7684022 \end{pmatrix}, \mu = \\
 \begin{pmatrix} 0.6637255 \\ -4.2957524 \\ 7.2816166 \end{pmatrix} \text{ (corresponding } s = 0.009926739); \\
 2. \quad m = 2, k = 1, L = \begin{pmatrix} 7.194264 \\ -8.883094 \\ -2.453192 \end{pmatrix}, \sigma = \begin{pmatrix} 0.07254785 \\ 4.00563476 \\ 5.11295875 \\ 7.20232401 \end{pmatrix}, \mu = \\
 \begin{pmatrix} -0.08266602 \\ -1.70045560 \\ 1.44687459 \end{pmatrix} \text{ (corresponding } s = 0.4986795); \\
 3. \quad m = 2, k = 1, L = \begin{pmatrix} -8.359644 \\ 3.495408 \\ -8.493300 \end{pmatrix}, \sigma = \begin{pmatrix} 0.63484701 \\ 0.01266988 \\ 0.01098159 \\ 8.59915832 \end{pmatrix}, \mu =
 \end{aligned}$$

$$\begin{pmatrix} -8.086132 \\ -8.308135 \\ 3.147155 \end{pmatrix} \text{ (corresponding } s = 0.99621);$$

$$4. \ m = 4, k = 1, L = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (corresponding } s = 0.3510004);$$

$$5. \ m = 4, k = 2, L = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (corresponding } s = 0.3479092);$$

$$6. \ m = 4, k = 4, L = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}, \sigma = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mu = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ (corresponding } s = 0.3461196).$$

Figure S.1 presents several experiments covering various model and parameter choices, resulting in different model identification and parameter s scenarios. As the sample size T increases, a clear trend of decreasing average estimation error (approaching zero) is observed in all cases, empirically confirming the convergence of Algorithm S.1.

Next, relative quality experiments are considered. For these experiments, the following values were fixed: $m = 2, k = 1, L = \mathbf{1}_{3 \times 1}, \sigma_{x_1} = \sigma_{x_2} = \sigma_{\eta} = 1$, and $\mu = \mathbf{0}_{3 \times 1}$. For selected T , 1,000 values of s were generated, denoted by $s_i, i = 1, \dots, 1,000$, where each s_i was assigned the result of the operation `runif(1, 0, 1)`.

For the obtained s values, the corresponding σ_y were calculated and the `generate_data` function from the `EMLI` package was used. Then,

the generated data were passed to both the `estimate_parameters` function from the `EMLI` package (using a convergence tolerance threshold of 10^{-5}) and a custom function implementing Algorithm S.2, which is a computationally simple MLM alternative [57].

Algorithm S.2 Empirical parameter estimation for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence, given a fixed dataset.

Input: Observed data sample $z_t, t = 1, \dots, T$

Output: Estimates $\hat{\mu}, \hat{\Sigma}, \hat{\sigma}_y^2$

- 1: $\hat{\mu} \leftarrow \frac{1}{T} \sum_{t=1}^T z_t$
 - 2: $\hat{\Sigma} \leftarrow \frac{1}{T-1} \sum_{t=1}^T (z_t - \hat{\mu})(z_t - \hat{\mu})'$
 - 3: $\hat{\sigma}_y^2 \leftarrow - \left[\frac{1}{T-2} \sum_{t=1}^{T-1} \left(z_t - \left(\hat{\mu} - \frac{z_T}{T} \right) \frac{T}{T-1} \right) \left(z_{t+1} - \left(\hat{\mu} - \frac{z_1}{T} \right) \frac{T}{T-1} \right)' \right]_{(m+1), (m+1)}$
-

Finally, the estimates obtained using both Algorithm S.1 and Algorithm S.2 were passed to the `evaluate_estimates` function from the `EMLI` package, with outputs denoted by $A_{1,i}$ and $A_{2,i}, i = 1, \dots, 1,000$, respectively.

For each selected value of T , a scatter plot was created where each point corresponds to the pair $\left(s_i, \frac{A_{1,i}}{A_{1,i} + A_{2,i}} \right)$. Each scatter plot also includes a blue `LOWESS` curve that smooths the results of individual experiments, and a red reference line at the 0.5 level, indicating equal accuracy of parameter estimates from both algorithms. Values below this line indicate higher accuracy of Algorithm S.1, whereas values above it indicate higher accuracy of Algorithm S.2.

Experiments were conducted for six different values of T . The results, along with the selected T values, are presented in Figure S.2.

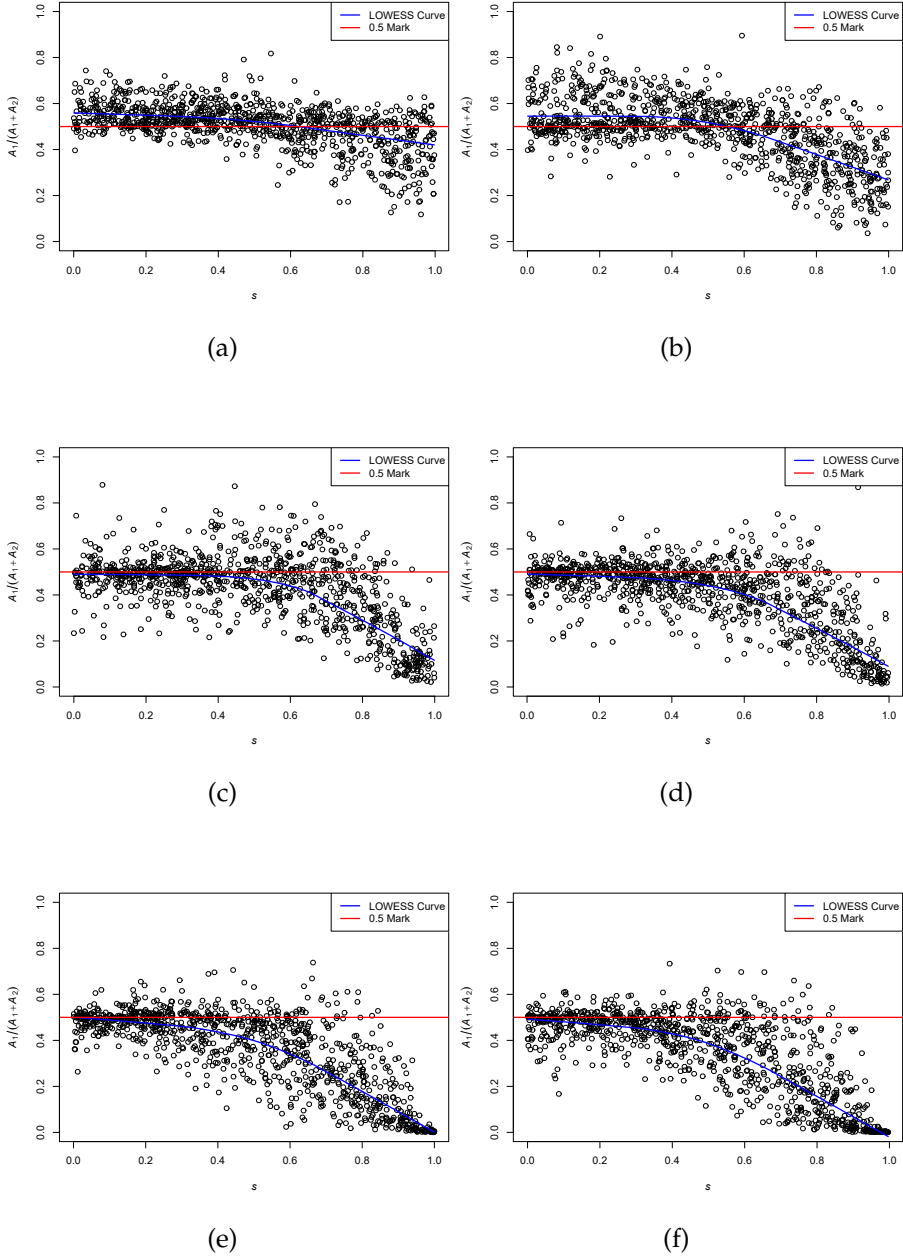


Figure S.2: Comparison of Algorithm S.1 with Algorithm S.2 for different sample sizes: (a) $T = 10$; (b) $T = 20$; (c) $T = 50$; (d) $T = 100$; (e) $T = 1,000$; (f) $T = 10,000$.

Figure S.2 shows that as the value of parameter s increases, Algorithm S.1 becomes, on average, more accurate than Algorithm S.2, and

this effect strengthens with increasing sample size T . For sufficiently large T (≥ 50), Algorithm S.1 is at least as accurate as Algorithm S.2 for all values of s (both algorithms provide equivalent accuracy when s is close to 0). However, for small T values and small to moderate s , Algorithm S.1 underperforms compared to Algorithm S.2 (neglecting the fact that for small samples, Algorithm S.2 tends to produce nonsensical estimates).

Accordingly, the estimates produced by Algorithm S.1 were compared with those obtained through direct minimization of the likelihood function defined by Equation (S.9). This minimization used the R function `optim` from the `stats` package, applying the optimization setting `method = "BFGS"`. The experiments showed that, when aligning the convergence tolerance thresholds of both algorithms and setting the `maxit` parameter in `optim` to ensure convergence, the estimates from both algorithms are equivalent.

Finally, runtime experiments were conducted. These experiments used the R function `microbenchmark` from the `microbenchmark` package, which allows precise measurement of operation durations. For several selected values of T , a single data sample was generated, and for each algorithm, parameter estimates were computed 1,000 times to determine the average runtime. The results are presented in Figure S.3.

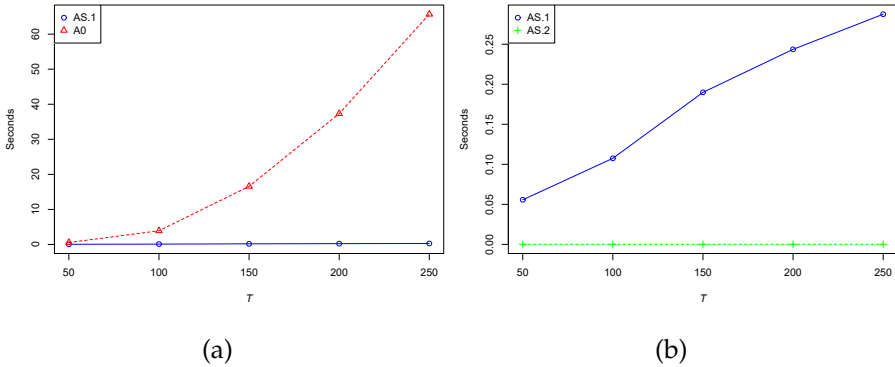


Figure S.3: Comparison of average algorithm runtimes for different sample sizes: (a) Algorithm S.1 (AS.1) versus direct minimization of the likelihood function defined by Equation (S.9) (A0); (b) Algorithm S.1 (AS.1) versus Algorithm S.2 (AS.2).

Figure S.3 shows that the developed Algorithm S.1 exhibits linear

complexity concerning T , significantly accelerating the traditional MLM, which has cubic complexity in T . On the other hand, Algorithm S.1 does not match the speed of the empirical estimator, which is also linear in T but has a significantly smaller constant.

S.2.8. Section Conclusions

- The univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence is suitable for revealing SET principles and initiating their application.
- The parameter estimation algorithm for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence, developed using SET, exhibits linear complexity concerning the sample size T and provides maximum likelihood parameter estimates. This allows it to be characterized as an efficient algorithm that combines the advantages of traditional MLM (optimal estimates) and empirical estimation (fast execution).

S.3. Adapting the Algorithm for Real-Time Parameter Estimation

This section, based on the article [28], analyzes the previously discussed parameter estimation problem in the context of real-time application.

S.3.1. Likelihood Function

The likelihood function is the same as that considered in Section S.2:

$$\begin{aligned} \mathcal{L} \left(s, \Sigma_{xx}, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta \right) &= \frac{1}{T} \left(\ln \left((|\Sigma_{xx}| \sigma_y^2)^T |Q| \right) \right. \\ &\quad \left. + \text{tr} \left(d_x \Sigma_{xx}^{-1} d'_x \right) \right. \\ &\quad \left. + \frac{1}{\sigma_y^2} \left(\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-1} \left(d_y - d_x \tilde{\Sigma}_{xy} \right) \right) \right). \end{aligned} \quad (\text{S.44})$$

Borrowing the results from Section S.2, the explicit estimators of the parameters μ_x , Σ_{xx} , $\tilde{\Sigma}_{xy}$, μ_η , and σ_y^2 , expressed in terms of s , are as

follows:

$$\hat{\mu}_x = \frac{z'_x \mathbf{1}}{T}, \quad (\text{S.45})$$

$$\hat{\Sigma}_{xx} = \frac{\hat{d}'_x \hat{d}_x}{T}, \quad (\text{S.46})$$

$$\hat{\Sigma}_{xy} = \left(\hat{d}'_x Q^{-1} \hat{d}_x - \frac{\hat{d}'_x Q^{-1} \mathbf{1} \left(\hat{d}'_x Q^{-1} \mathbf{1} \right)'}{\mathbf{1}' Q^{-1} \mathbf{1}} \right)^{-1} \hat{d}'_x Q^{-1} \left(z_y - \frac{z'_y Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}} \right), \quad (\text{S.47})$$

$$\hat{\mu}_\eta = \frac{\left(z_y - \hat{d}_x \hat{\Sigma}_{xy} \right)' Q^{-1} \mathbf{1}}{\mathbf{1}' Q^{-1} \mathbf{1}}, \quad (\text{S.48})$$

$$\hat{\sigma}_y^2 = \frac{\left(\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy} \right)' Q^{-1} \left(\hat{d}_y - \hat{d}_x \hat{\Sigma}_{xy} \right)}{T}. \quad (\text{S.49})$$

Structurally, the estimators defined by Equations (S.45)–(S.49) are suitable for use in the algorithm being developed for real-time parameter estimation. However, in practice, their computation must be performed recursively, as was done in Algorithm S.1, which was designed for parameter estimation with a fixed dataset.

Nevertheless, real-time estimation of the parameter s requires a fundamentally different approach than the one used in Section S.2.

S.3.2. Partial Derivative

Specifically, this approach involves the derivative of the likelihood function defined by Equation (S.44) with respect to the variable s :

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial s} &= \frac{s^2 - 1}{s^2} \left(\text{tr} \left(\frac{Q^{-1}}{T} \right) - \frac{1}{\sigma_y^2} \frac{\left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-2} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{T} \right) \\ &= \frac{s^2 - 1}{s^2} \left(\frac{\frac{s^2 + 1}{2s} - \left(1 + \frac{1}{T} \right) [Q^{-1}]_{T,T}}{2 \left(\frac{s^2 - 1}{2s} \right)^2} \right. \\ &\quad \left. - \frac{[Q^{-1}]_{T,T} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)' Q^{-2} \left(d_y - d_x \tilde{\Sigma}_{xy} \right)}{\sigma_y^2 T [Q^{-1}]_{T,T}} \right) \\ &= \left(1 + \frac{1}{T} \right) \left(1 - \frac{s^2 + 1}{s^2 - 1} \right) h, \end{aligned}$$

where

$$h = [Q^{-1}]_{T,T} - \frac{T}{T+1} \left(\frac{\frac{s^2+1}{2s} \sigma_y^2}{[Q^{-1}]_{T,T}} - 2 \left(\frac{s^2-1}{2s} \right)^2 \frac{(d_y - d_x \tilde{\Sigma}_{xy})' Q^{-2} (d_y - d_x \tilde{\Sigma}_{xy})}{T [Q^{-1}]_{T,T}} \right) \frac{\sigma_y^2}{[Q^{-1}]_{T,T}}.$$

Here, the equation $h = 0$ becomes relevant, as it serves as the basis for obtaining a real-time estimator for the parameter s . By replacing the variable σ_y^2 in h with its estimator (see Equation (S.49)) and keeping the other variables unchanged, we consider $\tilde{h} = \tilde{h}(s, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta)$ instead of $h = h(s, \tilde{\Sigma}_{xy}, \sigma_y^2, \mu_x, \mu_\eta)$. This step allows the term involving Q^{-2} in $\frac{\partial \mathcal{L}}{\partial s}$ to be combined with a similar term involving Q^{-1} into a unified expression, forming a robust ratio in the proposed estimator for s .

Having defined

$$a_T(\alpha, \beta) = \frac{([\alpha]_{1:T})' ([Q]_{1:T,1:T})^{-1} [\beta]_{1:T}}{T \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

$$b_T(\alpha, \beta) = \frac{([\alpha]_{1:T})' \left(\frac{s^2+1}{2s} ([Q]_{1:T,1:T})^{-1} - 2 \left(\frac{s^2-1}{2s} \right)^2 ([Q]_{1:T,1:T})^{-2} \right) [\beta]_{1:T}}{T \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

we obtain the following expression for $\tilde{h}$:

$$\tilde{h} = [Q^{-1}]_{T,T} - \frac{T}{T+1} \frac{b_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})}{a_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})}. \quad (\text{S.50})$$

To ensure a smooth alternation between parameter estimations, the proposed algorithm computes the terms $a_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})$ and $b_T(d_y - d_x \tilde{\Sigma}_{xy}, d_y - d_x \tilde{\Sigma}_{xy})$ in Equation (S.50) using atomic components that depend only on s :

$$a_{1,T} = a_T(\mathbf{1}, \mathbf{1}), \quad b_{1,T} = b_T(\mathbf{1}, \mathbf{1}), \quad (\text{S.51})$$

$$a_{2,T} = a_T(z_x, \mathbf{1}), \quad b_{2,T} = b_T(z_x, \mathbf{1}), \quad (\text{S.52})$$

$$a_{3,T} = a_T(z_x, z_x), \quad b_{3,T} = b_T(z_x, z_x), \quad (\text{S.53})$$

$$a_{4,T} = a_T(z_y, \mathbf{1}), \quad b_{4,T} = b_T(z_y, \mathbf{1}), \quad (\text{S.54})$$

$$a_{5,T} = a_T(z_y, z_y), \quad b_{5,T} = b_T(z_y, z_y), \quad (\text{S.55})$$

$$a_{6,T} = a_T(z_x, z_y), \quad b_{6,T} = b_T(z_x, z_y). \quad (\text{S.56})$$

Having defined

$$\begin{aligned} f_T \left(\gamma, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta \right) &= \gamma_{1,T} \mu_\eta^2 - 2\gamma_{4,T} \mu_\eta + \gamma_{5,T} \\ &\quad + 2 \left(\mu_\eta \gamma'_{2,T} - \gamma'_{6,T} + (\gamma_{4,T} - \mu_\eta \gamma_{1,T}) \mu'_x \right) \tilde{\Sigma}_{xy} \\ &\quad + \tilde{\Sigma}'_{xy} \left(\gamma_{3,T} + (\mu_x \gamma_{1,T} - 2\gamma_{2,T}) \mu'_x \right) \tilde{\Sigma}_{xy}, \end{aligned}$$

we obtain the final expression for $\tilde{h}$:

$$\tilde{h} = [Q^{-1}]_{T,T} - \frac{T}{T+1} \frac{f_T \left(b, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta \right)}{f_T \left(a, \tilde{\Sigma}_{xy}, \mu_x, \mu_\eta \right)}. \quad (\text{S.57})$$

S.3.3. Recursive Computations

In the proposed algorithm, the terms defined by Equations (S.51)–(S.56) are subject to recursive computation, as the real-time estimated parameters are expressed in terms of them.

Having defined

$$\begin{aligned} r_T^{(a)}(\gamma, \alpha) &= \left(1 - \frac{1}{T} \right) k_{3,T} \gamma + \frac{1}{T} \alpha, \\ r_T^{(b)}(\gamma, \alpha, \beta) &= \left(1 - \frac{1}{T} \right) k_{3,T} \gamma + \frac{1}{T} (k_{4,T} \alpha - k_{5,T} \beta), \end{aligned}$$

where

$$k_{3,T} = \frac{\left[([Q]_{1:(T-1), 1:(T-1)})^{-1} \right]_{T-1, T-1}}{\left[([Q]_{1:T, 1:T})^{-1} \right]_{T, T}}, \quad (\text{S.58})$$

$$k_{4,T} = \frac{1}{2} \left(\frac{s^2 + 1}{s} - \left(\frac{s^2 - 1}{s} \right)^2 \frac{\left[([Q]_{1:T, 1:T})^{-2} \right]_{T, T}}{\left[([Q]_{1:T, 1:T})^{-1} \right]_{T, T}} \right), \quad (\text{S.59})$$

$$k_{5,T} = \left(\frac{s^2 - 1}{s} \right)^2 \left[([Q]_{1:(T-1), 1:(T-1)})^{-2} \right]_{T-1, T-1}, \quad (\text{S.60})$$

the corresponding recursions are as follows:

$$a_{1,T} = r_T^{(a)}(a_{1,T-1}, c_{1,T}^2), \quad b_{1,T} = r_T^{(b)}(b_{1,T-1}, c_{1,T}^2, d_{1,T-1} c_{1,T}), \quad (\text{S.61})$$

$$a_{2,T} = r_T^{(a)}(a_{2,T-1}, c_{2,T}c_{1,T}), \quad b_{2,T} = r_T^{(b)}\left(b_{2,T-1}, c_{2,T}c_{1,T}, \frac{d_{1,T-1}c_{2,T} + d_{2,T-1}c_{1,T}}{2}\right), \quad (\text{S.62})$$

$$a_{3,T} = r_T^{(a)}(a_{3,T-1}, c_{2,T}c'_{2,T}), \quad b_{3,T} = r_T^{(b)}\left(b_{3,T-1}, c_{2,T}c'_{2,T}, \frac{d_{2,T-1}c'_{2,T} + c_{2,T}d'_{2,T-1}}{2}\right), \quad (\text{S.63})$$

$$a_{4,T} = r_T^{(a)}(a_{4,T-1}, c_{3,T}c_{1,T}), \quad b_{4,T} = r_T^{(b)}\left(b_{4,T-1}, c_{3,T}c_{1,T}, \frac{d_{3,T-1}c_{1,T} + d_{1,T-1}c_{3,T}}{2}\right), \quad (\text{S.64})$$

$$a_{5,T} = r_T^{(a)}(a_{5,T-1}, c_{3,T}^2), \quad b_{5,T} = r_T^{(b)}(b_{5,T-1}, c_{3,T}^2, d_{3,T-1}c_{3,T}), \quad (\text{S.65})$$

$$a_{6,T} = r_T^{(a)}(a_{6,T-1}, c_{2,T}c_{3,T}), \quad b_{6,T} = r_T^{(b)}\left(b_{6,T-1}, c_{2,T}c_{3,T}, \frac{d_{2,T-1}c_{3,T} + d_{3,T-1}c_{2,T}}{2}\right). \quad (\text{S.66})$$

Having defined

$$c_T(\alpha) = \frac{([\alpha]_{1:T})' \left[([Q]_{1:T,1:T})^{-1} \right]_{,T}}{\left[([Q]_{1:T,1:T})^{-1} \right]_{T,T}},$$

$$d_T(\alpha) = \frac{([\alpha]_{1:T})' \left[([Q]_{1:T,1:T})^{-2} \right]_{,T}}{\left[([Q]_{1:T,1:T})^{-2} \right]_{T,T}},$$

the c and d terms are as follows:

$$c_{1,T} = c_T(\mathbf{1}), \quad d_{1,T} = d_T(\mathbf{1}), \quad (\text{S.67})$$

$$c_{2,T} = c_T(z_x), \quad d_{2,T} = d_T(z_x), \quad (\text{S.68})$$

$$c_{3,T} = c_T(z_y), \quad d_{3,T} = d_T(z_y). \quad (\text{S.69})$$

Initial values (i.e., when $T = 1$) for the recursions defined by Equations (S.61)–(S.66) are as follows:

$$a_{1,1} = 1, \quad b_{1,1} = a_{1,1} \frac{2s}{s^2 + 1}, \quad (\text{S.70})$$

$$a_{2,1} = [z'_x]_{,1}, \quad b_{2,1} = a_{2,1} \frac{2s}{s^2 + 1}, \quad (\text{S.71})$$

$$a_{3,1} = [z'_x]_{,1} \left([z'_x]_{,1} \right)', \quad b_{3,1} = a_{3,1} \frac{2s}{s^2 + 1}, \quad (\text{S.72})$$

$$a_{4,1} = [z_y]_{1,}, \quad b_{4,1} = a_{4,1} \frac{2s}{s^2 + 1}, \quad (\text{S.73})$$

$$a_{5,1} = \left([z_y]_{1,} \right)^2, \quad b_{5,1} = a_{5,1} \frac{2s}{s^2 + 1}, \quad (\text{S.74})$$

$$a_{6,1} = [z'_x]_{,1} [z_y]_{1,}, \quad b_{6,1} = a_{6,1} \frac{2s}{s^2 + 1}. \quad (\text{S.75})$$

The secondary terms defined by Equations (S.67)–(S.69) must also be computed recursively.

Having defined

$$\begin{aligned} r_T^{(c)}(\gamma, \alpha) &= k_{1,T}\gamma + \alpha, \\ r_T^{(d)}(\gamma, \alpha) &= k_{2,T}\gamma + \alpha, \end{aligned}$$

where

$$k_{1,T} = \left[([Q]_{1:(T-1),1:(T-1)})^{-1} \right]_{T-1,T-1}, \quad (\text{S.76})$$

$$k_{2,T} = \left[([Q]_{1:T,1:T})^{-1} \right]_{T,T} \frac{\left[([Q]_{1:(T-1),1:(T-1)})^{-2} \right]_{T-1,T-1}}{\left[([Q]_{1:T,1:T})^{-2} \right]_{T,T}}, \quad (\text{S.77})$$

the corresponding recursions are as follows:

$$c_{1,T} = r_T^{(c)}(c_{1,T-1}, 1), \quad d_{1,T} = r_T^{(d)}(d_{1,T-1}, c_{1,T}), \quad (\text{S.78})$$

$$c_{2,T} = r_T^{(c)}\left(c_{2,T-1}, [z'_x]_{,T}\right), \quad d_{2,T} = r_T^{(d)}(d_{2,T-1}, c_{2,T}), \quad (\text{S.79})$$

$$c_{3,T} = r_T^{(c)}\left(c_{3,T-1}, [z_y]_{T,}\right), \quad d_{3,T} = r_T^{(d)}(d_{3,T-1}, c_{3,T}). \quad (\text{S.80})$$

The initial values (i.e., when $T = 1$) for the recursions defined by Equations (S.78)–(S.80) are as follows:

$$c_{1,1} = 1, \quad d_{1,1} = c_{1,1}, \quad (\text{S.81})$$

$$c_{2,1} = [z'_x]_{,1}, \quad d_{2,1} = c_{2,1}, \quad (\text{S.82})$$

$$c_{3,1} = [z_y]_{1,}, \quad d_{3,1} = c_{3,1}. \quad (\text{S.83})$$

S.3.4. Asymptotic Approximations

In the presented recursive computations, the quantities defined by Equations (S.58)–(S.60) and Equations (S.76)–(S.77) still pose computational efficiency challenges, as they involve inverting the matrix Q and are

undefined at $s = 0$. However, it can be observed that they have the following alternative expressions:

$$k_{1,T} = \begin{cases} 0, & s = 0 \\ s \frac{1-s^{2(T-1)}}{1-s^{2T}}, & 0 < s < 1, \\ 1 - \frac{1}{T}, & s = 1 \end{cases} \quad (\text{S.84})$$

$$k_{2,T} = \begin{cases} 0, & s = 0 \\ s \frac{1-s^{2(T+1)}}{1-s^{2T}} \frac{1-s^{2(2T-1)} - (1-s^2)s^{2(T-1)}(2T-1)}{1-s^{2(2T+1)} - (1-s^2)s^{2T}(2T+1)}, & 0 < s < 1, \\ \left(1 - \frac{1}{T}\right) \frac{2T-1}{2T+1}, & s = 1 \end{cases} \quad (\text{S.85})$$

$$k_{3,T} = \begin{cases} 1, & s = 0 \\ \frac{(1-s^{2(T-1)})(1-s^{2(T+1)})}{(1-s^{2T})^2}, & 0 < s < 1, \\ 1 - \frac{1}{T^2}, & s = 1 \end{cases} \quad (\text{S.86})$$

$$k_{4,T} = \begin{cases} 0, & s = 0 \\ s + \frac{1-s^2}{s} \left(T \frac{s^{2T}}{1-s^{2T}} - (T+1) \frac{s^{2(T+1)}}{1-s^{2(T+1)}} \right), & 0 < s < 1, \\ 1, & s = 1 \end{cases} \quad (\text{S.87})$$

$$k_{5,T} = \begin{cases} 1, & s = 0 \\ (1-s^2) \frac{1-s^{2(2T-1)} - s^{2(T-1)}(1-s^2)(2T-1)}{(1-s^{2T})^2}, & 0 < s < 1, \\ 0, & s = 1 \end{cases} \quad (\text{S.88})$$

here the boundary cases ($s = 0$ and $s = 1$) are derived from the corresponding interior case ($0 < s < 1$) by a continuous approach.

The quantities k defined by Equations (S.84)–(S.88) no longer require matrix inversion and are defined for all values of the parameter s . However, since MLM properties emerge only in large samples and

$$\lim_{T \rightarrow \infty} k_{1,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{2,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{3,T} = 1,$$

$$\lim_{T \rightarrow \infty} k_{4,T} = s,$$

$$\lim_{T \rightarrow \infty} k_{5,T} = 1 - s^2$$

converge sufficiently fast, the proposed algorithm inserts the limits of k as T approaches infinity, i.e., instead of k , the following approximations

$\tilde{k}$ are used:

$$\tilde{k}_{1,T} = s \approx k_{1,T}, \quad (\text{S.89})$$

$$\tilde{k}_{2,T} = s \approx k_{2,T}, \quad (\text{S.90})$$

$$\tilde{k}_{3,T} = 1 \approx k_{3,T}, \quad (\text{S.91})$$

$$\tilde{k}_{4,T} = s \approx k_{4,T}, \quad (\text{S.92})$$

$$\tilde{k}_{5,T} = 1 - s^2 \approx k_{5,T}. \quad (\text{S.93})$$

This enables the estimation of parameter s by moving s to the left-hand side of the equation $0 = \tilde{h}$, since $[Q^{-1}]_{T,T}$ (see Equation (S.57)) is approximated by s . Moreover, this also reduces the number of elementary operations in the recursive computations of the quantities defined by Equations (S.51)–(S.56) and improves their stability, all achieved at the cost of practically negligible loss in accuracy.

S.3.5. Maximum Likelihood Estimators

Specifically, the following estimator for parameter s is obtained:

$$\hat{s} = \frac{T}{T+1} \frac{f_T \left(b, \hat{\Sigma}_{xy}, \hat{\mu}_x, \hat{\mu}_\eta \right)}{f_T \left(a, \hat{\Sigma}_{xy}, \hat{\mu}_x, \hat{\mu}_\eta \right)}.$$

In real-time parameter estimation, the sequence of estimates is then obtained as:

$$\hat{s}_{T+1} = \frac{T}{T+1} \frac{f_T \left(b, \hat{\Sigma}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T} \right)}{f_T \left(a, \hat{\Sigma}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T} \right)}, \quad (\text{S.94})$$

where $\hat{\Sigma}_{xy,T}$, $\hat{\mu}_{x,T}$, and $\hat{\mu}_{\eta,T}$ are defined below along with $\hat{\sigma}_{y,T}^2$ and $\hat{\Sigma}_{xx,T}$.

The estimators $\hat{\Sigma}_{xy}$, $\hat{\mu}_\eta$, and $\hat{\sigma}_y^2$, given by Equations (S.47)–(S.49), respectively, can also be expressed in terms of the recursively computed

quantities defined by Equations (S.51)–(S.56):

$$\begin{aligned} \hat{\Sigma}_{xy,T} = & \left(a_{3,T} - a_{2,T} \hat{\mu}'_{x,T} - \hat{\mu}_{x,T} a_{2,T} + \hat{\mu}_{x,T} a_{1,T} \hat{\mu}'_{x,T} \right. \\ & \left. - \frac{(a_{2,T} - \hat{\mu}_{x,T} a_{1,T})(a_{2,T} - \hat{\mu}_{x,T} a_{1,T})'}{a_{1,T}} \right)^{-1} \left(a_{6,T} - \hat{\mu}_{x,T} a_{4,T} \right. \\ & \left. - \frac{a_{4,T}}{a_{1,T}} (a_{2,T} - \hat{\mu}_{x,T} a_{1,T}) \right), \end{aligned} \quad (\text{S.95})$$

$$\hat{\mu}_{\eta,T} = \frac{a_{4,T} - \hat{\Sigma}'_{xy,T} (a_{2,T} - \hat{\mu}_{x,T} a_{1,T})}{a_{1,T}}, \quad (\text{S.96})$$

$$\hat{\sigma}_{y,T}^2 = sf_T \left(a, \hat{\Sigma}_{xy,T}, \hat{\mu}_{x,T}, \hat{\mu}_{\eta,T} \right). \quad (\text{S.97})$$

Meanwhile, the estimators $\hat{\mu}_x$ and $\hat{\Sigma}_{xx}$ (see Equations (S.45)–(S.46)) do not depend on s and can be easily computed recursively:

$$\hat{\mu}_{x,T} = \left(1 - \frac{1}{T} \right) \hat{\mu}_{x,T-1} + \frac{[z'_x]_T}{T}, \quad (\text{S.98})$$

$$\hat{\Sigma}_{xx,T} = u_T - \hat{\mu}_{x,T} \hat{\mu}'_{x,T}, \quad (\text{S.99})$$

where $u_T = \frac{[z'_x]_{1:T} [z_x]_{1:T}}{T}$ is computed as follows:

$$u_T = \left(1 - \frac{1}{T} \right) u_{T-1} + \frac{[z'_x]_T [z_x]_T}{T}. \quad (\text{S.100})$$

At the initial time point (i.e., when $T = 1$), the recursions defined by Equation (S.98) and Equation (S.100) are initialized as follows:

$$\hat{\mu}_{x,1} = [z'_x]_{,1}, \quad (\text{S.101})$$

$$u_1 = [z'_x]_{,1} [z_x]_1. \quad (\text{S.102})$$

S.3.6. Estimation Algorithm

The material of this section is summarized by presenting the pseudocode of the constructed algorithm.

Algorithm S.3 Real-time maximum likelihood parameter estimation for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence.

Input: Observed data sample $z_t, t = 1, \dots, T$, and initial (guess) value of s (i.e., $\hat{s}_1$)

Output: Estimates $\hat{\mu}_t, \hat{\Sigma}_t, \hat{\sigma}_{y,t}^2, t = 1, \dots, T$

- 1: Compute $c_{j,1}, d_{j,1}, j = 1, \dots, 3$, using Equations (S.81)–(S.83), and compute $a_{i,1}, b_{i,1}, i = 1, \dots, 6$, using Equations (S.70)–(S.75) with $s = \hat{s}_1$
 - 2: Compute $\hat{\mu}_{x,1}$ and $\hat{\Sigma}_{xx,1}$ using Equation (S.101) and Equation (S.99) with Equation (S.102), and compute $\hat{\Sigma}_{xy,1}, \hat{\mu}_{\eta,1}$, and $\hat{\sigma}_{y,1}^2$ using Equations (S.95)–(S.97) with $s = \hat{s}_1$
 - 3: Construct $\hat{\mu}_1$ from $\hat{\mu}_{x,1}$ and $\hat{\mu}_{\eta,1}$, and construct $\hat{\Sigma}_1$ from $\hat{\Sigma}_{xx,1}, \hat{\Sigma}_{xy,1}, \hat{s}_1$, and $\frac{\hat{\sigma}_{y,1}^2}{\hat{s}_1}$ using Equations (S.10)–(S.11)
 - 4: **for** $t \leftarrow 2$ to T **do**
 - 5: Compute $\hat{s}_t$ using Equation (S.94)
 - 6: $\hat{s}_t \leftarrow \max(0, \min(\hat{s}_t, 1))$
 - 7: Compute $c_{j,t}, d_{j,t}, j = 1, \dots, 3$, using Equations (S.78)–(S.80), and compute $a_{i,t}, b_{i,t}, i = 1, \dots, 6$, using Equations (S.61)–(S.66). In these computations, use $\tilde{k}_l, l = 1, \dots, 5$, defined by Equations (S.89)–(S.93) (with $s = \hat{s}_t$), instead of $k_l, l = 1, \dots, 5$, defined by Equations (S.84)–(S.88)
 - 8: Compute $\hat{\mu}_{x,t}$ and $\hat{\Sigma}_{xx,t}$ using Equation (S.98) and Equations (S.99)–(S.100), and compute $\hat{\Sigma}_{xy,t}, \hat{\mu}_{\eta,t}$, and $\hat{\sigma}_{y,t}^2$ using Equations (S.95)–(S.97) with $s = \hat{s}_t$
 - 9: Construct $\hat{\mu}_t$ from $\hat{\mu}_{x,t}$ and $\hat{\mu}_{\eta,t}$, and construct $\hat{\Sigma}_t$ from $\hat{\Sigma}_{xx,t}, \hat{\Sigma}_{xy,t}, \hat{s}_t$, and $\frac{\hat{\sigma}_{y,t}^2}{\hat{s}_t}$ using Equations (S.10)–(S.11)
 - 10: **end for**
-

S.3.7. Empirical Results

This subsection presents the empirical verification of Algorithm S.3. For this purpose, several numerical experiments were conducted to assess the convergence of the estimates produced by this algorithm as the number of observation time points increases, and to evaluate their quality by comparing them with the estimates obtained using the

empirical estimation method (see Algorithm S.4), which can be easily computed both in the fixed dataset and real-time settings [9].

All necessary computations were performed using the R statistical package [92].

Algorithm S.4 Real-time empirical parameter estimation for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence.

Input: Observed data sample $z_t, t = 1, \dots, T$

Output: Estimates $\hat{\mu}_t, \hat{\Sigma}_t, \hat{\sigma}_{y,t}^2, t = 3, \dots, T$

```

1:  $\hat{\mu}_3 \leftarrow \frac{z_1 + z_2 + z_3}{3}$ 
2:  $u \leftarrow \frac{z_1 z'_1 + z_2 z'_2 + z_3 z'_3}{2}$ 
3:  $\hat{\Sigma}_3 \leftarrow u - \frac{3}{2} \hat{\mu}_3 \hat{\mu}'_3$ 
4:  $v \leftarrow z_1 z'_2 + z_2 z'_3$ 
5:  $\mu_a \leftarrow \left( \hat{\mu}_3 - \frac{z_1}{3} \right) \frac{3}{2}$ 
6:  $\mu_b \leftarrow \left( \hat{\mu}_3 - \frac{z_3}{3} \right) \frac{3}{2}$ 
7:  $\hat{\sigma}_{y,3}^2 \leftarrow -[(v - 2\mu_b \mu'_a)]_{m+1, m+1}$ 
8: for  $t \leftarrow 4$  to  $T$  do
9:    $\hat{\mu}_t \leftarrow \left(1 - \frac{1}{t}\right) \hat{\mu}_{t-1} + \frac{z_t}{t}$ 
10:   $u \leftarrow \frac{t-2}{t-1} u + \frac{z_t z'_t}{t-1}$ 
11:   $\hat{\Sigma}_t \leftarrow u - \frac{t}{t-1} \hat{\mu}_t \hat{\mu}'_t$ 
12:   $v \leftarrow \frac{t-3}{t-2} v + \frac{z_{t-1} z'_t}{t-2}$ 
13:   $\mu_a \leftarrow \left( \hat{\mu}_t - \frac{z_1}{t} \right) \frac{t}{t-1}$ 
14:   $\mu_b \leftarrow \left( \hat{\mu}_t - \frac{z_t}{t} \right) \frac{t}{t-1}$ 
15:   $\hat{\sigma}_{y,t}^2 \leftarrow -[(v - (t-1)(t-2)\mu_b \mu'_a)]_{m+1, m+1}$ 
16: end for
```

Based on the Monte Carlo method [63], for three different sets of model parameters—each representing a characteristic scenario—100 data samples were generated, each consisting of $T = 100,000$ observations. Without loss of generality, the following values were chosen: $m = 2, k = 1, \mu = 0_{3 \times 1}, L = 0_{3 \times 1}, \sigma_{x_1}^2 = \sigma_{x_2}^2 = \sigma_\eta^2 = 1$.

Then, by varying σ_y^2 , three different parameter sets were obtained, corresponding to different scenarios:

1. Low output noise scenario ("Low s "): $\sigma_y^2 = 0.01$ corresponding to $s = 0.009805$.

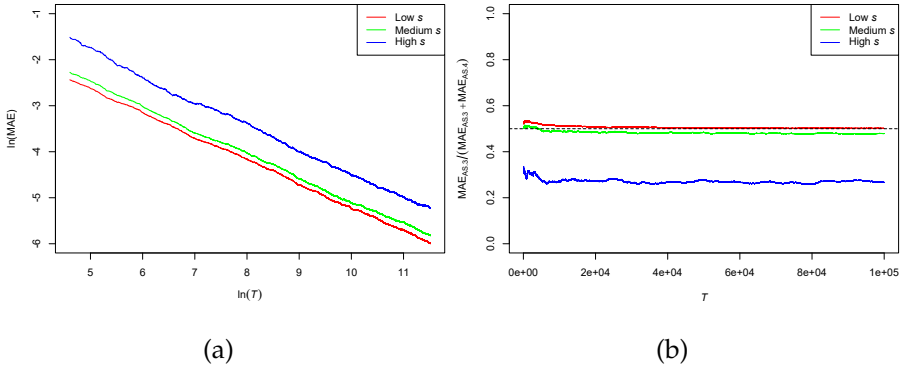
2. Medium output noise scenario ("Medium s "): $\sigma_y^2 = 1$ corresponding to $s = 0.381966$.
3. High output noise scenario ("High s "): $\sigma_y^2 = 100$ corresponding to $s = 0.904875$.

Algorithm S.3 and Algorithm S.4 were run on each of the generated data samples (Algorithm S.3 was initialized with a randomly chosen value of s). After obtaining all estimates, they were compared with the true, i.e., data-generating parameter values. For each combination of the scenario, algorithm, iteration (sample size), and parameter subset, the mean absolute error (MAE) [125] was computed.

The parameters were grouped into three subsets to neutralize the effect of scale:

1. Elements equal to zero (this subset includes μ and the off-diagonal elements of Σ).
2. Elements equal to one (this subset includes the diagonal elements of Σ , excluding $[\Sigma]_{3,3}$, which is adjusted by subtracting $2\sigma_y^2$).
3. The element whose value depends on the scenario (i.e., σ_y^2).

The obtained results are illustrated in Figure S.4.



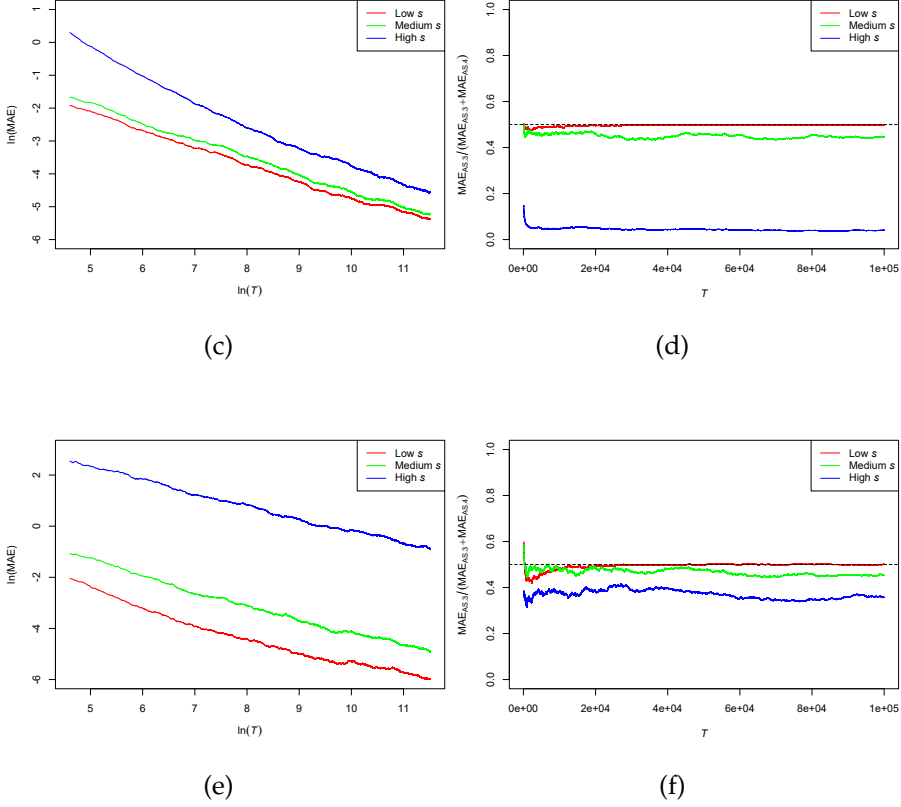


Figure S.4: Empirical verification of Algorithm S.3 (AS.3) (curves plotted from $T = 100$): (a) convergence for parameter subset No. 1; (b) comparison with Algorithm S.4 (AS.4) for parameter subset No. 1; (c) convergence for parameter subset No. 2; (d) comparison with Algorithm S.4 (AS.4) for parameter subset No. 2; (e) convergence for parameter subset No. 3; (f) comparison with Algorithm S.4 (AS.4) for parameter subset No. 3.

Figure S.4a, Figure S.4c, and Figure S.4e show that as the sample size T increases, the MAE approaches zero across all considered scenarios and parameter subsets. This empirically confirms the convergence of Algorithm S.3.

Meanwhile, Figure S.4b, Figure S.4d, and Figure S.4f demonstrate that as the relative weight of output observation noise increases in the model, the advantage of Algorithm S.3 over Algorithm S.4 in terms of estimation quality becomes more pronounced: when the output noise is low, both algorithms are essentially equivalent; with medium noise, a

noticeable difference emerges; with high noise, the difference becomes substantial.

Combining the empirical results with the fact that each iteration of Algorithm S.3 requires a constant amount of computation time (i.e., it does not increase with T), it can be concluded that the MLM-based Algorithm S.3 is a strong competitor to Algorithm S.4, which serves as a benchmark in the context of various heuristic methods. The latter are often computationally simple but achieve this at the cost of estimation quality.

Although Algorithm S.3 requires more elementary operations per iteration than Algorithm S.4, it not only has comparable algorithmic complexity suitable for real-time parameter estimation, but also demonstrates superior reliability in noisy output observation environments.

S.3.8. Section Conclusions

- The results of SET implementation can be used to construct efficient real-time parameter estimation algorithms; however, additional specific steps are also required (constructing efficient real-time parameter estimation algorithms poses more challenges than constructing corresponding algorithms for fixed datasets).
- The MLM- and recursion-based real-time parameter estimation algorithm for the univariate-output, pure-time-series-data CSEM with an additive-noise output measurement equation and assumptions of normality and independence demonstrates competitive estimation quality in the presence of noisy output observations.

S.4. Practical Aspects

Although the construction of parameter estimation algorithms such as Algorithm S.1 and Algorithm S.3 requires specific efforts (which increase with the generality of the model), they are relatively easy to implement programmatically, making their application technically straightforward [29]. On the other hand, the input to such algorithms consists of data interpreted through a specific model (i.e., data serve as input to an algorithm designed to estimate the parameters of a given model), but these components are not always obvious in practical applications. In

other words, parameter estimation, while essential and challenging in its own right, is only one part of the overall modeling process.

This section, based on the article [96], explores aspects relevant to practical applications in the context of COVID-19 pandemic modeling.

S.4.1. Modeling the COVID-19 Pandemic

The COVID-19 pandemic remains a relevant research topic even after its decline [2, 40, 72, 94]. Analysis of officially collected data on infections, deaths, and other related statistics can provide valuable insights for forecasting future pandemics and developing strategies to manage them [30, 86]. However, there is a certain gap between official COVID-19 statistics and models that allow for in-depth analysis of the effectiveness of control measures such as lockdowns [7], face masks [102], COVID-19 passports [119], or mass vaccination [56], raising doubts about whether the available data are sufficient for drawing high-quality conclusions [65]. This gap is examined in this section to demonstrate a possible approach to working with existing data that at least partially mitigates its limiting effects and to highlight the importance of data collection and usage that enables deeper analysis.

The motivation to examine this seemingly unproblematic topic arises from a debate in editorial letters regarding the impact of the COVID-19 passport on virus spread in Lithuania [55, 108, 109]. The initial letter assessed this impact using descriptive statistics and certain common-sense assumptions. However, the opposing side pointed out the shortcomings of such an approach and argued that the conclusions might be misleading due to issues with both the method and the data.

As a matter of fact, when studying the characteristics of pandemic-related processes, researchers often use descriptive statistics, which summarize or describe the characteristics of a dataset by calculating measures of central tendency, variability, and frequency distribution [33]. However, such methods are limited to illustrative purposes and lack the capacity for thorough and reliable analysis of the phenomena of interest. Any form of data aggregation inherently results in information loss, which in turn reduces the model's sensitivity to details that are often crucial, potentially compromising the quality of the results [98].

To achieve lower-level modeling where the studied objects are broken down into smaller conceptual units until fundamental mechanisms

become apparent [103], epidemiology often employs compartmental models (providing the model with detailed data is another condition for enabling low-level modeling). Essentially, compartmental models are a very general modeling technique. In this framework, the population is divided into compartments, such as S, I, R, and D (S stands for *Susceptible*, I for *Infected*, R for *Recovered*, and D for *Deceased*) [32]. Individuals in the population are allowed to move between compartments according to rules defined by the model structure, which in turn reflects the modeling assumptions [84]. Compartmental models are often described using ordinary (deterministic) differential equations, but they can also be implemented stochastically, which is more realistic but analytically more complex [17]. These models are used in epidemiology to predict the spread of infection, the number of infected individuals, the duration of the pandemic, and to estimate various epidemiological parameters such as the reproduction number. They can also be used to assess how public health interventions imposed by authorities may influence the course of a pandemic [16, 133].

Analysis of the available data through the lens of the above-described, daily-resolution (which is the maximum available granularity), four-compartment SIRD model reveals several issues related to the data. Here, compartment I is best represented by the daily numbers of new COVID-19 cases [121]. However, it is important to emphasize that these data are subject to several problems:

1. They are inseparable from testing volumes, reflected by the daily numbers of new COVID-19 tests [82], which vary significantly on a daily basis.
2. Only a portion of the population is tested, so the daily numbers of new COVID-19 cases [121] do not reflect the entire population.
3. Testing can yield false results—a person with a positive test may not be infected, and vice versa.
4. The time between testing and result reporting is variable and essentially unknown.

These issues significantly complicate the determination of the actual daily population flow into compartment I. Moreover, identifying

transitions into compartment R presents a similar challenge, as there is no established practice of testing to confirm recovery.

The situation is more favorable when examining transitions into compartment D, as it is relatively well represented by the daily numbers of new COVID-19 deaths [122]. This is because such data are easily recorded on the actual day of death in full. The main drawback is the potentially incorrect attribution of the cause of death, analogous to the third issue listed for the daily numbers of new COVID-19 cases [121].

In the absence of high-quality data for each modeled compartment, it may be reasonable to perform compartmental modeling using only those compartments for which the data are most reliable. This section argues that, in the case of a SIRD model, compartment D is such a case, and advocates for the use of MLM to estimate the parameters of the SIRD model based solely on COVID-19 deaths, with observation noise incorporated into the model to account for data limitations. This approach is then applied to actual COVID-19 death data from Lithuania to analyze the impact of the COVID-19 passport during the Delta and Omicron waves. Finally, potential improvements to the data are discussed.

S.4.2. Data and Model

We consider a discrete-time deterministic SIRD model. In this model, transitions between compartments are defined by the following equations:

$$S_{t+1} = S_t - aI_tS_t, \quad (\text{S.103})$$

$$I_{t+1} = I_t (1 + aS_t - b - c), \quad (\text{S.104})$$

$$R_{t+1} = R_t + bI_t, \quad (\text{S.105})$$

$$D_{t+1} = D_t + cI_t, \quad (\text{S.106})$$

where $t = 0, \dots, T - 1$, $S_0 = 1 - I_0$, $I_0 = \mu$, $R_0 = 0$, $D_0 = 0$. The coefficients a , b , and c represent the transition probabilities from S to I , from I to R , and from I to D , respectively, while μ denotes the initial proportion of the population that is infected. This value must be greater than zero for the infection dynamics to manifest in the model.

To use this model, it is necessary to estimate its parameters (i.e., a , b , c , and μ) based on data describing compartments S , I , R , and D . One

of the most suitable and desirable methods for this purpose is MLM, which possesses (asymptotically) optimal properties. It is important to note that the chosen model equations can be expressed in terms of the remaining ones, thus giving the model a form suitable for working with only selected compartments (data).

Here, we use daily-resolution COVID-19 death data (i.e., the discrete-time index t corresponds to days) to estimate the model parameters using MLM. The model-related quantity corresponding to the daily numbers of new COVID-19 deaths [122] is defined as follows (expressed per million population and divided by one million):

$$\Delta D_t = D_{t+1} - D_t. \quad (\text{S.107})$$

The time series of ΔD in Lithuania from August 7, 2021, to May 8, 2022, is shown in Figure S.5.

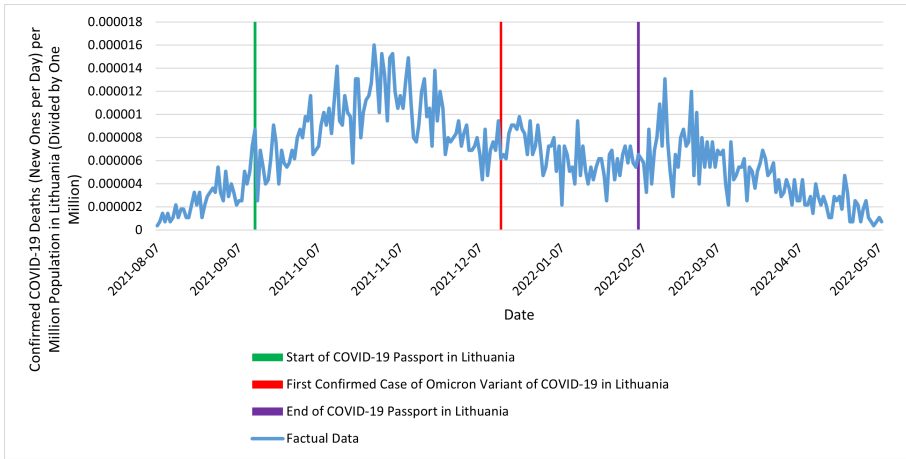


Figure S.5: Daily new COVID-19 deaths per million population in Lithuania (divided by one million) from August 7, 2021, to May 8, 2022.

The period shown in Figure S.5 essentially covers the Delta and Omicron waves in Lithuania. The middle vertical line marks the date when the first Omicron cases were detected (December 15, 2021). The left line marks the introduction of the COVID-19 passport (September 13, 2021), and the right line marks its expiration (February 5, 2022). Throughout the indicated period, ΔD is non-zero, while on the first days outside this range (i.e., August 6, 2021, and May 9, 2022), ΔD

equals zero. This period is analyzed in Subsection S.4.3, which examines the impact of the COVID-19 passport on the spread of the pandemic in Lithuania.

Concerning the term defined by Equation (S.107), the model defined by Equations (S.103)–(S.106) can be expressed as follows:

$$\Delta D_0 = \mu c, \quad (\text{S.108})$$

$$\Delta D_1 = \Delta D_0 \left(1 + a \left(1 - \frac{\Delta D_0}{c} \right) - b - c \right), \quad (\text{S.109})$$

$$\Delta D_{t+1} = \Delta D_t \left(1 + a \left(1 - \frac{\Delta D_t}{c} - \left(\frac{b}{c} - 1 \right) \sum_{j=0}^{t-1} \Delta D_j \right) - b - c \right), \quad (\text{S.110})$$

where $t = 1, \dots, T - 2$.

Considering the limitations of ΔD , this term is augmented with normal multiplicative observation noise (the normal distribution is widely used for modeling errors [134, 135]):

$$Z_t = \Delta D_t (1 + \sigma \epsilon_t), \quad (\text{S.111})$$

where $t = 0, \dots, T - 1$, ϵ_t are observation uncertainties modeled as independent realizations of a standard normal random variable (i.e., with zero mean and unit variance), and σ is the corresponding standard deviation (i.e., the strength of the observation noise).

The model defined by Equations (S.108)–(S.111) forms the mathematical foundation of the proposed method. The next (and final) step is the estimation of the corresponding parameters, which can be efficiently performed using MLM, incorporating specific aspects of SET and a standard function optimization algorithm. Since the mathematical details of these operations are not essential to the main idea of this section, they are not elaborated further.

S.4.3. Results

The following phenomena are analyzed in the case of Lithuania:

1. The impact of the introduction of the COVID-19 passport on the spread of the Delta wave.

2. The impact of the termination of the COVID-19 passport on the spread of the Omicron wave.

For the first phenomenon, the analysis period spans from the beginning of the available data (i.e., August 7, 2021) to December 27, 2021. It is assumed here that the Omicron wave began on December 28, 2021. Although the first cases were officially recorded on December 15, 2021 (and Omicron was likely circulating earlier), this variant only began to overshadow the Delta wave later. Given that at that time the seven-day moving average of daily COVID-19 cases was below 500—indicating the decline of the Delta wave and that Omicron had not yet gained significant momentum—it is assumed that the onset of Omicron coincides with the renewed increase in COVID-19 cases, when this average exceeded the 500 threshold. Essentially, the onset of Omicron could be modeled as a parameter that may also be estimated from the data [10]; for simplicity, it is treated here as predefined.

This selected period was divided into two parts: the period without the COVID-19 passport (i.e., until September 12, 2021) and the period with the COVID-19 passport (i.e., from September 13, 2021). Since the COVID-19 passport can essentially only affect virus transmission (modeled by parameter a), and not recovery or mortality (modeled by parameters b and c , respectively), the only changing model parameter in this case was a . A stepwise trend was applied to its variation, i.e., $a = a_1$ until September 12, 2021, and $a = a_2$ from September 13, 2021 (all other model parameters were assumed constant over time), which corresponds to the assumption that the effect of the COVID-19 passport was immediate. Overall, these modeling choices reflect the assumption that the characteristics of the Delta wave and any significant external factors (except the COVID-19 passport) remained constant during the analyzed period.

The obtained results are presented in Figure S.6. Figure S.6a illustrates the estimated model overlaid on the data, consisting of a deterministic component (i.e., the mean) and a two-standard-deviation confidence band, which contains approximately 95% of the model-generated data. This model is considered a good fit to the actual data, as its mean aligns with the center of the observed data, and nearly all observations fall within the confidence band.

Figure S.6b shows the same as Figure S.6a, but using a hypothetical

model that applies the estimated parameter a_1 throughout the entire modeling horizon. Since the hypothetical model also fits the data well, it can be concluded that the difference between the estimated a_1 and a_2 is not significant. This implies that aside from random variation, a_2 is equivalent to a_1 (see Fig. S.6c, which compares the means of both models; it shows that the estimate of a_2 is lower than that of a_1).

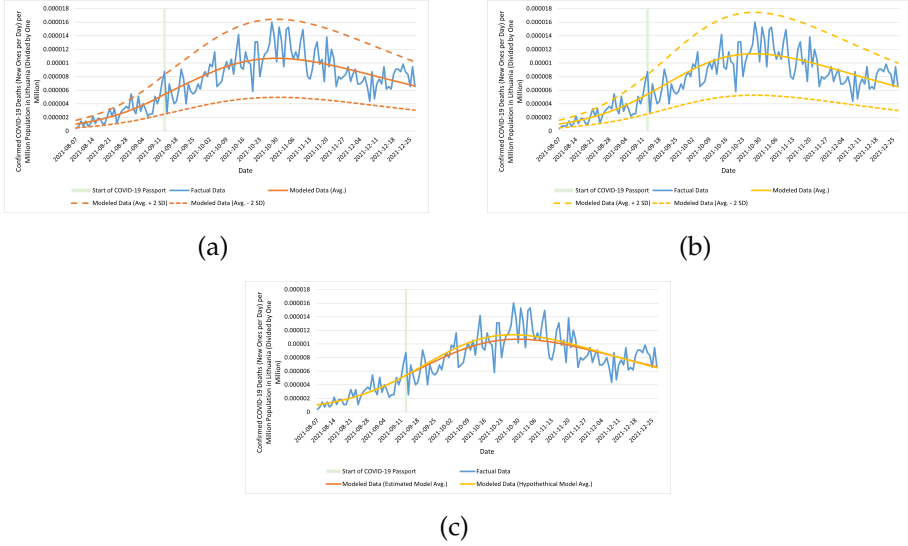


Figure S.6: Modeling the impact of the COVID-19 passport introduction on the spread of the Delta wave in Lithuania: (a) estimated model; (b) hypothetical model; (c) comparison of estimated and hypothetical models.

Thus, there is insufficient evidence to conclude that the COVID-19 passport had an impact on the spread of the Delta wave in Lithuania.

This result differs from the conclusion in [108], where the authors claim that the COVID-19 passport had a suppressive effect on the spread of COVID-19 during the Delta wave in Lithuania. This discrepancy can largely be attributed to different interpretations of the natural course of the pandemic in the applied models.

In [108], it is assumed that factors such as the use of the COVID-19 passport and/or public awareness were necessary for the infection dynamics to eventually peak and decline. The authors present a scenario in which, in the absence of these factors, the number of infections grows

exponentially without bound. However, this interpretation overstates the situation, as such dynamics can primarily be explained by the fact that once a critical portion of the population has been infected, the number of new cases inevitably begins to decline.

In contrast, the compartmental model presented in this work incorporates this fact into its structure (see Fig. S.6b, which illustrates the hypothetical model corresponding to a scenario similar to the exponential model in [108], but reflecting the natural dynamics of a pandemic). This enables a more adequate comparison of different scenarios (see Fig. S.6c).

For the second phenomenon, the analysis was conducted analogously, using the data from December 28, 2021, to May 8, 2022, corresponding to the Omicron wave. In this case, the change point for parameter a is assumed to be February 5, 2022, when the COVID-19 passport expired.

The results are presented in Figure S.7.

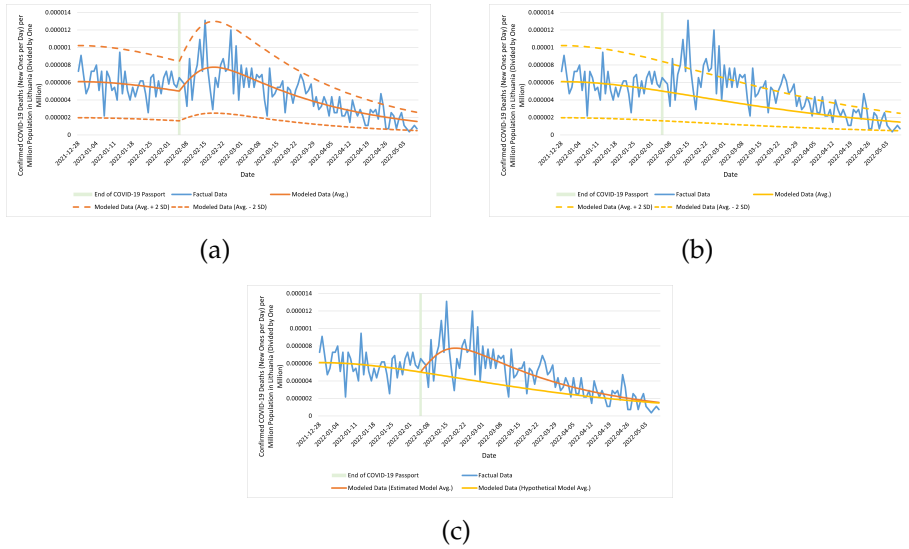


Figure S.7: Modeling the impact of the COVID-19 passport expiration on the spread of the Omicron wave in Lithuania: (a) estimated model; (b) hypothetical model; (c) comparison of estimated and hypothetical models.

Figure S.7a illustrates the estimated model overlaid on the actual

data. Figure S.7b shows the hypothetical model, which does not accurately represent the actual data. The fact that the hypothetical model does not match the data indicates that the difference between the estimated a_1 and a_2 is significant, meaning that a_2 differs from a_1 even beyond random variation (see Fig. S.7c, which compares the means of both models; it shows that the estimate of a_2 is greater than that of a_1).

Nevertheless, this result remains somewhat ambiguous, as the observed increase in deaths may primarily reflect a rise in infections that occurred before the termination of the COVID-19 passport. Therefore, to draw a definitive conclusion, more detailed data would be required.

S.4.4. Discussion

To eliminate modeling obstacles, efforts must first be directed toward improving data quality, as no method can fully compensate for data deficiencies. In this regard, it is worth considering the possibility of linking COVID-19 case data with COVID-19 testing data, which are currently unjustifiably disconnected despite their fundamental interdependence.

This connection could be restored by recording the date of the positive test result at the individual level. Such a procedure essentially requires more meticulous data collection and/or presentation, rather than initiating entirely new processes. It could significantly enhance COVID-19 modeling capabilities, as this relationship is tied to the more accurate identification of the population transitioning into compartment I.

Moreover, an analogous link could be established between COVID-19 deaths and COVID-19 cases, thereby enabling more accurate inference of infection-related outcomes from death data.

S.4.5. Section Conclusions

- Reliable conclusions from practical dynamic system modeling require sufficiently flexible nonlinear models that describe relevant, fundamentally understood laws. However, their implementation (parameter estimation) may be limited by the available data, which provide insufficient information.

General Conclusions

- To fully leverage the potential of dynamic system modeling in the era of big data, efficient parameter estimation algorithms are essential—those capable of combining high estimation accuracy with high computational speed. However, existing solutions often exhibit only one of these properties, especially in the case of more complex models.
- The stepwise estimation technique provides a methodological foundation for developing efficient parameter estimation algorithms for stochastic dynamic models when the dataset is fixed. Based on the maximum likelihood method and recursive computations, this technique makes possible the combination of estimation optimality and computational efficiency, addressing the needs of modern data analysis.
- The univariate-output, pure-time-series-data cumulative structural equation model with an additive-noise output measurement equation and assumptions of normality and independence provides a suitable framework for applying the principles of the stepwise estimation technique and initiating algorithm development using a bottom-up methodology. The resulting parameter estimation algorithm, applied to a fixed dataset, yields maximum likelihood estimates (which are particularly beneficial under intense output observation noise) and has a runtime proportional to the number of observation time points.
- Developing efficient algorithms for real-time parameter estimation poses greater challenges than estimation with fixed datasets. Nevertheless, their construction can also be based on the maximum likelihood method and recursion, and thus can be carried out by appropriately extending the results of the stepwise estimation technique. The resulting real-time parameter estimation algorithm for the univariate-output, pure-time-series-data cumulative structural equation model with an additive-noise output measurement equation and assumptions of normality and independence exhibits estimation properties analogous to those of the corresponding algorithm for fixed datasets.

- Practical dynamic system modeling requires nonlinear models reflecting fundamental laws and detailed data necessary for accurate parameter estimation. As demonstrated by the case of modeling COVID-19 pandemic dynamics, empirical data limitations can significantly reduce modeling capabilities and reduce the reliability of conclusions. For example, publicly available COVID-19 data lack the information needed to link testing, infection, and mortality data at the individual level. Such data are essential for reliably assessing the impact of interventions such as the COVID-19 passport during the pandemic. Therefore, when applying dynamic models in decision-making contexts, it is crucial to ensure the completeness and quality of the data.

Vytautas Dulskis

Stochastinių dinaminių sistemų modelių parametrų efektyvių
vertinimo algoritmų sudarymas ir taikymas

Daktaro disertacija

Gamtos mokslai

Informatika (N 009)

Disertacijos redaktorė: Jorūnė Rimeisytė-Nekrašienė

Vytautas Dulskis

Development and Application of Efficient Algorithms for Parameter
Estimation in Stochastic Dynamic System Models

Doctoral Dissertation

Natural Sciences

Informatics (N 009)

Summary Editor: Zuzana Šiušaitė

Vilniaus universiteto leidykla
Saulėtekio al. 9, III rūmai, LT-10222 Vilnius
El. p. info@leidykla.vu.lt, www.leidykla.vu.lt
bookshop.vu.lt, journals.vu.lt
Tiražas 20 egz.