https://doi.org/10.15388/vu.thesis.27

VILNIAUS UNIVERSITETAS

Jūratė
VAIČIULYTĖ

Paslėptųjų Markovo modelių tyrimas ir taikymas daugiamačių sekų palaipsnei analizei

DAKTARO DISERTACIJA

Gamtos mokslai,
informatika N 009

VILNIUS 2020

Disertacija rengta 2015–2019 metais Vilniaus universitete.

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

SANTRAUKA

Pastaraisiais dešimtmečiais klasikinis paslėptųjų Markovo modelių (PMM) parametrų vertinimo uždavinys buvo modifikuotas. Įvestas papildomas reikalavimas, kad stebėjimai turi būti apdorojami nuosekliai (t. y. palaipsniui), o ne saugomi kompiuterio atmintyje ir apdorojami kaip vientisas rinkinys. Šioje disertacijoje sprendžiami uždaviniai, susiję su paslėptųjų Markovo modelių parametrų vertinimo palaipsniais algoritmais. Nagrinėjami paslėptieji Markovo modeliai, kurių būsenų erdvė yra diskreti, o stebėjimai – tolydieji dydžiai.

Darbo tikslas – paslėptųjų Markovo modelių daugiamačių parametrų vertinimo palaipsnių algoritmų tyrimas ir taikymas srautu gaunamų atsitiktinių sekų analizėje.

Šioje disertacijoje pasiūlyti du palaipsniai paslėptųjų Markovo modelių parametrų vertinimo algoritmai: pirmasis – palaipsnis Gauso PMM parametrų vertinimo algoritmas, o antrasis – palaipsnis Dirichlė PMM parametrų vertinimo algoritmas. Pasiūlyti palaipsniai PMM parametrų vertinimo algoritmai leidžia taupyti skaičiavimo laiką ir kompiuterio atmintį, kadangi nebereikia saugoti visų modelio apmokymo duomenų kompiuterio atmintyje, o skaičiavimo laikas nuoseklios sekų analizės atveju yra tiesinis. Disertacijoje pasiūlyti algoritmai yra sudaryti iš dviejų dalių: pirmoji dalis yra skirta PMM parametrams vertinti, o antroji – stebėjimams atpažinti ir PMM parametrų įverčiams atnaujinti, perskaičiuojant parametrų vertes nepriklausomai nuo jau apdorotos duomenų aibės. Pasiūlytas modelio būsenų perėjimo tikimybių skaičiavimas, kuris skiriasi nuo modifikuotos tiesioginio-atbulinio sklidimo procedūros. Atlikta išsami pasiūlytų algoritmų lyginamoji analizė su klasikiniais algoritmais, sprendžiant daugiamačių stebėjimų klasifikavimo ir klasterizavimo uždavinius.

Disertaciją sudaro įvadas, 4 skyriai, išvados, literatūros sąrašas, autorės publikacijų disertacijos tema sąrašas. Disertacijos apimtis: 128 puslapiai, 23 paveikslai ir 29 lentelės. Literatūros sąrašą sudaro 120 šaltinių. Tyrimų rezultatai publikuoti 4 recenzuojamuose periodiniuose mokslo žurnaluose, pristatyti tarptautinėse ir nacionalinėse konferencijose.

Raktiniai žodžiai: stochastinis procesas, paslėptieji Markovo modeliai, palaipsnis parametrų vertinimas, didžiausio tikėtinumo metodas.

ABSTRACT

In recent decades, the classic task of estimating hidden Markov models (HMM) has been modified. It introduces an additional requirement where observations must be processed sequentially (i.e. recursively) in time rather than after being stored in computer memory and then processed as blocks.

In this thesis, we investigate important problems involving online algorithms for estimating Hidden Markov Model parameters.

The aim of this work is to investigate and apply Hidden Markov Models multivariate parameter estimation algorithms to online analysis of sequences.

In this work, two online algorithms for recursively estimating HMM parameters are proposed. The first is the online Gaussian HMM parameter estimation algorithm. The second is the online Dirichlet HMM parameter estimation algorithm. The proposed online HMM parameter estimation algorithms save computational time and computer memory by eliminating the need to store all training data in computer memory by using sequential data analysis. The algorithms proposed in the work consist of two parts: the first part is dedicated to the training of HMM parameters and the second part is used to recognize the observations and update the HMM parameters by re-estimating the parameters. Calculation of HMM state transition probabilities by changing the classic modified „Forward-Backward“ procedure, is proposed. All algorithms are applied to solve real data analysis tasks.

Detailed comparative analysis of the proposed algorithms was performed, solving the tasks of classification and clustering of multidimensional observations.

Keywords: Stochastic processes, hidden Markov model, online algorithm, recursive parameter estimation, maximum likelihood, EM algorithm.

PADĖKA

Dėkoju darbo vadovui prof. habil. dr. Leonidui Sakalauskui už nuoseklų vadovavimą, vertingas mokslines konsultacijas ir pagalbą ruošiant šį darbą, disertacijos recenzentams doc. dr. Gintautui Tamulevičiui ir prof. dr. Juliui Žilinskui už kritines ir konstruktyvias pastabas tobulinant disertaciją, taip pat Vilniaus universiteto Duomenų mokslo ir skaitmeninių technologijų instituto kolegoms už bendradarbiavimą ir pagalbą rengiant disertaciją.

Nuoširdžiai dėkoju savo artimiesiems ir draugams, palaikiusiems ir motyvavusiems mane judėti į priekį.

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

Jūratė Vaičiulytė

ŽYMĖJIMAI

Simboliai

α – Dirichlė skirstinio parametras;

μ – vidurkių vektorius;

π – pradinio buvimo būsenoje tikimybių vektorius;

σ – kovariacijų matrica;

A – paslėptojo Markovo modelio būsenų perėjimo tikimybių vektorius;

B – paslėptojo Markovo modelio būsenos išvesties tikimybinis skirstinys;

L – tikėtinumo funkcija;

M – stebėjimų dimensijų skaičius;

N – paslėptojo Markovo modelio būsenų skaičius;

O – daugiamatis stebėjimo vektorius;

S – paslėptojo Markovo modelio būsena;

t – laiko momentas;

δ – algoritmo efektyvumo kriterijus, nusakantis vidutinį atstumą nuo modelio parametrų įverčių iki tikrųjų parametrų verčių;

T – stebėjimų imties dydis;

Santrumpos

MVM – matematinės vilties maksimizavimas (angl. *expectation maximization*);

KTR – klaidingai teigiamas rodiklis (angl. *false positive rate*);

PŽA – pavienių žodžių atpažinimas (angl. *isolated word recognition*);

DTM – didžiausio tikėtinumo metodas (angl. *maximum likelihood method*);

TTF – tikimybės tankio funkcija (angl. *probability density function*);

PMM – paslėptasis Markovo modelis (angl. *hidden Markov model*);

P_DPMM_PVA – Palaipsnis Dirichlė PMM parametrų vertinimo algoritmas;

P_GPMM_PVA – Palaipsnis Gauso PMM parametrų vertinimo algoritmas;

SP – standartinė paklaida (angl. *standard error*);

ŽAT – žodžių atpažinimo tikslumas (angl. *word recognition rate*);

Sąvokos

Bendras klasifikavimo teisingumas – vienas iš algoritmo našumo įvertinimo kriterijų (angl. *accuracy*);

„Baum-Welch“ algoritmas – tai ypatingas MVM algoritmo atvejis, taikantis tiesioginio-atbulinio sklidimo (angl. *Forward-Backward*) algoritmą ir naudojamas

nežinomiems paslėptojo Markovo modelio parametrams rasti;

Blokinis metodas – algoritmas, apdorojantis duomenis pablokiui.

Rinkinio algoritmas – algoritmas, be žmogaus įsikišimo apdorojantis iš anksto paruoštus ir į rinkinius sudėtus duomenis (angl. *batch algorithm*);

Pablokiui – rekursinis algoritmas atnaujina modelio parametrų įverčius, apdorojus tam tikro dydžio stebėjimų seką (angl. *block-wise*);

Klaidingai neigiamas įvykio įvertinimas (angl. *false-negative*);

Klaidingai teigiamas įvykio įvertinimas (angl. *false-positive*);

Tiesioginio-atbulinio sklidimo procedūra (angl. *Forward-Backward*);

F-rodiklis – jungia tikslumo rodiklio ir atrinkimo rodiklio reikšmes, taip išreiškiamas algoritmo našumas viena reikšme (angl. *F-score*);

Geometrinis vidurkis – vienas iš algoritmo našumo įvertinimo kriterijų (angl. *GMean*);

Paslėptieji Markovo modeliai – tai statistinis Markovo modelis, kuriame modeliuojama sistema laikoma Markovo grandine su nestebimomis (t. y. paslėptomis) būsenomis.

Tikslumo rodiklis – vienas iš algoritmo našumo įvertinimo kriterijų (angl. *precision*);

Atkuriamumo rodiklis – vienas iš algoritmo našumo įvertinimo kriterijų (angl. *recall*);

Palaipsnis algoritmas – adaptyvus algoritmas, nuolat atnaujinantis modelį pagal srautu gautą informaciją bei mažinantis duomenų saugojimo vietą ir apdorojimo laiką (angl. *recursive, incremental, on-line*);

Palaipsnis įvertinimas – modelio parametrai atnaujinami nuosekliai analizuojant srautu gaunamus duomenis;

Signalas – stebėjimų seka;

Simbolinis metodas – algoritmas, apdorojantis duomenis pasimboliui.

Pasimboliui – rekursinis algoritmas atnaujina modelio parametrų įverčius, apdorojus kiekvieną gautą stebėjimą (angl. *symbol-wise*);

Softmax funkcija – duomenims normalizuoti skirta funkcija, kuri užtikrina, kad stebėjimo vektoriaus elementai yra teigiami, o jų suma lygi 1;

Specifiškumo rodiklis – vienas iš algoritmo našumo įvertinimo kriterijų (angl. *specificity*);

Teisingai neigiamas įvykio įvertinimas (angl. *true-negative*);

Teisingai teigiamas įvykio įvertinimas (angl. *true-positive*);

ILIUSTRACIJŲ SĄRAŠAS

LENTELIŲ SĄRAŠAS

TURINYS

1 ĮVADAS

1.1 Tyrimų sritis ir problemos aktualumas

Svarbus įvairių taikomųjų sričių (tokių kaip kompiuterinio matymo (angl. *computer vision*) programos, šnekos atpažinimo, vaizdų analizė, Edge-AI ir kt.) bruožas yra dideli ir nuolat srautu gaunami duomenų rinkiniai. Juos galima gana efektyviai apdoroti realiu laiku taikant įvairius algoritmus. Pastaraisiais metais ypač sparčiai populiarėja ir plečiasi dirbtinio intelekto sritis, tobulinami mašininio mokymo algoritmai dėl naujų besiformuojančių skaičiavimo iššūkių. Jie susiję su duomenų interpretavimu, mokymusi ir sprendimų priėmimu realiu laiku. Tokio pobūdžio srityse, kai duomenis reikia apdoroti ir panaudoti mokymui realiu laiku, sunku pritaikyti gilaus mokymo ar tradicinio mašininio mokymo metodus, nes jiems reikia didelės statinės mokymo duomenų aibės ir pakankamų apmokymo resursų (giliam mokymui dažnai naudojami grafiniai procesoriai). Adaptyvaus (palaipsnio) mokymo ar mokymo realiu laiku (angl. *unsupervised learning*) metodai tik pradedami tyrinėti dirbtinių neuroninių tinklų metodologijoje [1, 2].

Atsirandančiose realaus laiko sistemose apdorojamus didelius duomenų srautus bandoma modeliuoti kaip stochastinius procesus. Stochastiniai procesai yra plačiai nagrinėjami inžinerijos, gamtos mokslų, socialinių mokslų, verslo ir finansų bei kitose srityse. Stochastinių procesų įvairovė yra didžiulė, apima nepriklausomus ir identiškai pasiskirsčiusius procesus, stacionarius procesus, Gauso procesus, Markovo procesus, paslėptuosius Markovo modelius (PMM) ir kt. Nepaisant plataus stochastinių procesų taikymo, vis dar išlieka daug sudėtingų uždavinių, susijusių su gebėjimu modeliuoti tikrovę. Iš tiesų, būsenų vertinimo [6–8], modelio parametrų vertinimo [9–11] ir sprendimų priėmimo [12–16] uždaviniai ir toliau nagrinėjami signalų apdorojimo, automatinio valdymo ir informacijos teorijos srityse. Įvairiuose dirbtinio intelekto modeliuose kyla panašių uždavinių, todėl išsprendus vienos srities uždavinį, galima daryti išvadas apie kitus modelius.

Pastaraisiais dešimtmečiais iškeltas modifikuotas PMM parametrų vertinimo uždavinys. Jame įvestas papildomas reikalavimas, kad stebėjimai turi būti apdorojami nuosekliai (t. y. palaipsniui), o ne saugomi kompiuterio atmintyje ir apdorojami kaip vientisas rinkinys. Ši palaipsnio PMM parametrų vertinimo formuluotė tapo didelės teorinės ir praktinės reikšmės, kadangi kai kuriose taikomosiose programose (pvz., objektų aptikimo ir stebėjimo [17–19]) skaičiavimo požiūriu neįmanoma saugoti ir apdoroti didelių stebėjimo duomenų partijų (ar rinkinių) [10, 20].

Palaipsnis PMM parametrų vertinimas taip pat svarbus programose, kuriose PMM parametrai gali kisti laike.

„Baum-Welch“ (rinkinio) algoritmas yra skirtas ne palaipsnio PMM parametrų vertinimo uždaviniui spręsti, tačiau juo remiasi daugybė palaipsnių MVM metodų [9,21–24]. Kiti siūlomi PMM parametrų vertinimo metodai remiasi palaipsniais didžiausio tikėtinumo [25,26], prognozavimo klaidų (angl. *prediction error*) metodais [26–28]. Šie palaipsniai PMM parametrų vertinimo būdai įprastai konverguoja į jų tikslo funkcijų lokalius (ne globalius) ekstremumus.

Pastaruoju metu pasiūlyti du nauji PMM būsenų perėjimo parametrų vertinimo metodai vienmačiams Gauso modeliams (su įrodytomis konvergavimo savybėmis) [11, 17], naudojant ergodines (paslėptųjų) Markovo grandinių būsenos procesų ir informacijos teorijų sąvokas. [11, 17] straipsnių autorių siūlomi vertinimo metodai yra neprieštaringi (angl. *consistent*) vertinant PMM būsenų perėjimo parametrus, esant PMM struktūros apribojimo sąlygoms ir turint visas žinias apie PMM stebėjimo procesą. Tačiau bendru nežinomų parametrų atveju – tiek būsenų, tiek stebėjimų procesuose – nuoseklus daugiamačių (nebūtinai Gauso) PMM parametrų vertinimas vis dar yra reikšmingas neišspręstas uždavinys. Šioje disertacijoje pristatoma nauja metodologija, kuria rinkinio algoritmams kuriami palaipsnių algoritmų atitikmenys. Nagrinėjami klasikiniai didžiausio tikėtinumo metodai yra asimptotiškai optimalūs, todėl kitokie algoritmai (pvz, dirbtinių neuroninių tinklų) iš principo negali pateikti geresnių rezultatų. Tad prasminga disertacijoje gautus rezultatus ir pasiekimus pritaikyti adaptyvių (palaipsnių) dirbtinio intelekto metodų kūrimui.

1.2 Tyrimų objektas

Disertacijos tyrimo objektas – palaipsniai algoritmai, skirti diskrečiųjų paslėptųjų Markovo modelių daugiamačiams parametrams vertinti, kai stebėjimai yra tolydieji dydžiai.

1.3 Darbo tikslas ir uždaviniai

Tikslas:

- Diskrečiųjų paslėptųjų Markovo modelių daugiamačių parametrų vertinimo palaipsnių algoritmų kūrimas, jų tyrimas ir taikymas atsitiktinių sekų analizėje, kai stebėjimai yra tolydieji dydžiai.

Uždaviniai:

- Analitiškai apžvelgti atsitiktinių sekų nuoseklioje analizėje taikomus paslėptųjų Markovo modelių parametrų vertinimo metodus.
- Sudaryti palaipsnius paslėptųjų Markovo modelių daugiamačių parametrų vertinimo algoritmus.
- Sukurtus palaipsnius paslėptųjų Markovo modelių daugiamačių parametrų vertinimo algoritmus ištirti statistinio modeliavimo būdu ir palyginti su kitais paslėptųjų Markovo modelių parametrų vertinimo algoritmais.
- Sukurtus palaipsnius paslėptųjų Markovo modelių daugiamačių parametrų vertinimo metodus pritaikyti atsitiktinių sekų nuoseklioje analizėje.

1.4 Tyrimų metodai

Šios disertacijos tyrimas pagrįstas šiais metodais:

- Darbo tikslui pasiekti ir uždaviniams spręsti analizuojami moksliniai palaipsnių PMM parametrų vertinimo algoritmų tyrimai.
- Naudojami informacijos paieškos, sisteminimo, analizės, lyginamosios analizės ir apibendrinimo metodai.
- Sukurtas palaipsnis PMM parametrų vertinimo algoritmas tiriamas Monte-Karlo metodu ir sprendžiant testinius uždavinius.
- Eksperimentinio tyrimo metodu atliekamas stebėjimų apdorojimas ir statistinė tyrimų rezultatų analizė, o gautiems rezultatams įvertinti naudojamas lyginimo ir apibendrinimo metodas.
- Darbe naudojamos algoritmų teorijos, duomenų gavybos, statistinės analizės, atpažinimo teorijos žinios.

1.5 Mokslinis darbo naujumas

Tyrimas yra moksliškai reikšmingas dėl šių priežasčių:

- Disertacijoje sukurti ir eksperimentiškai ištirti daugiamačių stebėjimų palaipsniai klasifikavimo metodai, paremti paslėptaisiais Markovo modeliais.

- Siūlomas daugiamačių Gauso paslėptųjų Markovo modelių būsenų perėjimo tikimybių palaipsnis skaičiavimo metodas, paremtas Čapmano ir Kolmogorovo lygtimi. Šiuo būsenų perėjimo tikimybių skaičiavimo metodu žinomuose palaipsniuose algoritmuose gaunamas didesnis klasifikavimo tikslumas nei įprastine tiesioginio sklidimo (angl. *forward*) procedūra.

- Disertacijoje pateikiamas naujas palaipsnis algoritmas paslėptųjų Markovo modelių parametrams vertinti, kai stebėjimai yra pasiskirstę pagal Dirichlė skirstinį.

- Sudaryti algoritmai pritaikyti pavienių žodžių atpažinimo, užimtumo nustatymo ir pulsarų nustatymo uždaviniuose. Eksperimentų rezultatai patvirtino pasiūlytų algoritmų efektyvumą – algoritmų skaičiavimo laikas sumažėja, o klasifikavimo tikslumas nežymiai pakinta (iki 3 %).

- Palaipsnių algoritmų tyrimai gali būti taikomi kito tipo algoritmų palaipsnių atitikmenims kurti bei taikyti srautu gaunamų duomenų klasifikavimui.

1.6 Praktinė darbo reikšmė

Siūlomi palaipsniai paslėptųjų Markovo modelių daugiamačių parametrų vertinimo algoritmai gali būti naudojami įvairiose daugiamačių duomenų apdorojimo sistemose ir įrankiuose, kuriuose analizuojami duomenys yra stochastinio proceso pobūdžio, o mokymo duomenų saugojimo reikalavimai ribojami:

- palaipsnis Gauso PMM parametrų vertinimo algoritmas pritaikytas pavieniams žodžiams atpažinti, kai fiksuotas šnekos duomenų kiekis yra skirtas apmokyti, o tolimesni šnekos duomenys yra atpažįstami ir naudojami pakartotiniam parametrų vertinimui.

- palaipsnis Dirichlė PMM parametrų vertinimo algoritmas pritaikytas užimtumo nustatymo uždavinyje, kai iš sensorių gautų duomenų reikia nustatyti, ar analizuojama patalpa yra užimta, ar ne.

- palaipsnis Dirichlė PMM parametrų vertinimo algoritmas pritaikytas pulsarų kandidatų nustatymo uždavinyje.

1.7 Ginamieji teiginiai

Disertacijos ginamieji teiginiai yra šie:

- Palaipsnis PMM parametrų vertinimo algoritmas, kai išvesties tikimybinis skirstinys yra Gauso, tiesinio sudėtingumo ir klasifikavimo (stebėjimų atpažinimo) tikslumu prilygsta tradiciniam (rinkinio) „Baum-Welch“ algoritmui.
- Taikant Čapmano ir Kolmogorovo lygtį modelio būsenų perėjimo tikimybėms skaičiuoti modelio parametrai konverguojami geriau nei su tradicine tiesioginio sklidimo (angl. *forward*) procedūra.
- Egzistuoja pakankamas duomenų rinkinys, skirtas pradinei algoritmo aproksimacijai, užtikrinantis algoritmo stabilumą ir neleidžiantis jam konverguoti į išsigimusius lokalius tikėtinumo funkcijos ekstremumus.
- Palaipsnis Gauso ir Dirichlė PMM parametrų vertinimo algoritmai gali būti taikomi kelių klasių klasifikavimo praktiniams uždaviniams, išreiškiamiems stochastiniu procesu ir modeliuojamiems paslėptaisiais Markovo modeliais, spręsti.

1.8 Disertacijos struktūra

Darbą sudaro įvadas, keturi skyriai, išvados, literatūros sąrašas, autorės publikacijų disertacijos tema sąrašas.

Įvade pateikiami tyrimų sritis, objektas, darbo tikslas ir uždaviniai, tyrimų metodai, mokslinis darbo naujumas, praktinė darbo reikšmė, ginamieji teiginiai.

Pirmame skyriuje pateikiamas paslėptųjų Markovo modelių parametrų palaipsnio vertinimo metodų analitinis tyrimas, aptariamas pasirinktos temos aktualumas ir bendra problematika.

Antrame skyriuje sudaromas palaipsnis paslėptųjų Markovo modelių daugiamačių parametrų vertinimo algoritmas, paremtas didžiausio tikėtinumo metodu ir klasikiniu MVM algoritmu, ir pristatomas jo taikymas daugiamačiams stebėjimams, pasiskirsčiusiems pagal Gauso dėsnį, klasterizuoti. Aprašomi su sintetiniais duomenimis atlikti eksperimentai, norint ištirti siūlomo algoritmo savybes.

Trečiame skyriuje sudaromas palaipsnis paslėptųjų Markovo modelių daugiamačių parametrų vertinimo algoritmas, paremtas didžiausio tikėtinumo metodu, kai daugiamačiai stebėjimai yra pasiskirstę pagal Dirichlė skirstinį. Aprašomi su sintetiniais duomenimis atlikti eksperimentai, norint ištirti siūlomo algoritmo savybes.

Ketvirtame skyriuje aprašomas sudarytų Dirichlė ir Gauso paslėptųjų Markovo modelių parametrų palaipsnio vertinimo algoritmų taikymas pavienų žodžių atpažininimo, užimtumo nustatymo ir pulsarų nustatymo uždaviniuose. Aprašomas algoritmų efektyvumo tyrimas ir lyginimas su kitais egzistuojančiais algoritmais.

Disertacijos apimtis: 128 puslapiai, 29 lentelės, 23 iliustracijos. Disertacijoje remtasi 120 literatūros šaltinių.

2 PASLĖPTŲJŲ MARKOVO MODELIŲ PARAMETRŲ PALAIPSNIO VERTINIMO METODŲ ANALITINIS TYRIMAS

Šiame skyriuje pateikiama analitinė paslėptųjų Markovo modelių parametrų palaipsnio vertinimo metodų apžvalga.

Kai kurios šio skyriaus dalys yra publikuotos [29, 30] straipsniuose.

2.1 Paslėptieji Markovo modeliai

Paslėptieji Markovo modeliai (PMM) – tai Markovo modeliai, kai stebėjimas yra atsitiktinė būsenos funkcija. Šis modelis (vadinamas paslėptuoju Markovo modeliu) yra tarsi dvigubas stochastinis procesas, nes paslėptasis procesas gali būti stebimas tik per kitų stochastinių procesų sukurtą stebėjimų seką.

Markovo savybė sako, kad paslėptojo kintamojo s laiko momentu t sąlyginis tikimybinis skirstinys (angl. *conditional probability distribution*), kai duotos paslėptojo kintamojo $s(t)$ reikšmės visais laiko momentais, priklauso tik nuo paslėptojo kintamojo $s(t-1)$ reikšmės. Panašiai – stebimo kintamojo $O(t)$ reikšmė priklauso tik nuo paslėptojo kintamojo $s(t)$ reikšmės.

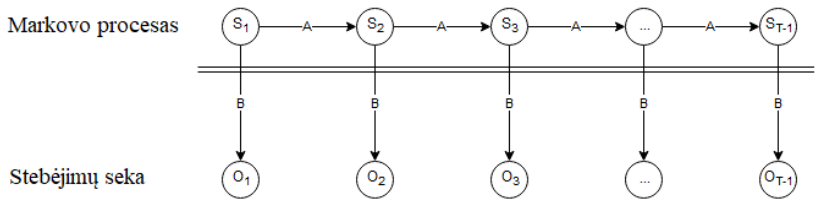
PMM paslėptųjų kintamųjų būsenų erdvė yra diskreti, o stebėjimai yra tolydieji dydžiai (pvz., pasiskirstę pagal Gauso dėsnį). PMM parametrai yra būsenų perėjimo ir išvesties tikimybės.

Būsenų perėjimo tikimybės kontroliuoja, kaip parenkama paslėptoji būsena laiko momentu t, kai duota paslėptoji būsena laiko momentu $t-1$. Paslėptųjų būsenų aibė turi vieną iš N galimų reikšmių, kurios sumodeliuotos pagal kategorinį skirstinį. Tai reiškia, kad kiekvienai iš galimų N būsenų, kurioje laiko momentu t gali būti paslėptasis kintamasis, egzistuoja perėjimo tikimybė iš tos būsenos į kiekvieną iš paslėptojo kintamojo N galimų būsenų laiko momentu $t+1$, kai iš viso yra N^2 perėjimo tikimybių ($N \times N$ perėjimo tikimybių matrica). Kadangi bet kuri perėjimo tikimybė gali būti nustatyta, kai yra žinomos kitos, iš viso yra $N(N-1)$ perėjimo parametrų.

Kiekvienai iš N galimų būsenų priskiriama išvesties tikimybių aibė, kuri valdo stebimo kintamojo skirstinį tam tikru laiko momentu, kai duota paslėptojo kintamojo būsena tuo laiko momentu. Šios aibės dydis priklauso nuo stebimo kintamojo. Pavyzdžiui, jei stebimas kintamasis yra M-matis vektorius, pasiskirstęs pagal daugiamatį Gauso skirstinį, vadinasi, yra M parametrų, valdančių vidurkius,

ir $M(M+1)/2$ parametrų, kontroliuojančių kovariacijų matricą, kai iš viso yra $O(NM^2)$ išvesties parametrų.

Markovo procesas (žr. 1 pav., paslėptas po dviguba ištisine linija) nustatomas pagal esamą būseną ir perėjimo tikimybių matricą **A**. Galima stebėti tik O_i, kuris susijęs su matricos **B** paslėptosiomis Markovo proceso būsenomis [31]. 1 paveikslo rodyklės žymi sąlygines priklausomybes, o S_i reikšmės – paslėptųjų būsenų seką. Procesas generuoja stebėjimą O_i pagal būsenos S_j išvesties tikimybinį skirstinį $B_j(O)$ (2 paveikslas).

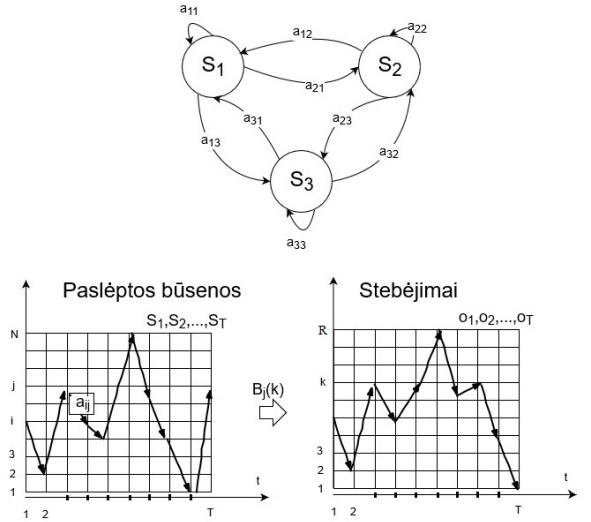


1 pav.: Paslėptieji Markovo modeliai.

Taigi, norint aprašyti PMM, užtenka nusakyti rinkinį $\lambda = (\mathbf{A},\mathbf{B},\pi)$ [32]. Apmokant PMM pagal stebėjimų duomenis O, įvertinami modelio parametrai $\lambda = (\mathbf{A},\mathbf{B})$, t. y. perėjimo iš vienos būsenos į kitą tikimybių matrica ir stebėjimų tikimybės tankio funkcijos kiekvienai modelio būsenai. Modelis aprašomas stebėjimų vektorių tikimybės tankio funkcija, nusakančia konkretaus stebėjimų vektoriaus tikimybę. Tikimybinis skirstinys gali būti įvairių formų, pvz., tolydusis skirstinys, normalusis ir t. t.

PMM parametrų vertinimas dažnai atliekamas taikant didžiausio tikėtinumo įverčio metodą. Kiti įverčiai, tokie kaip maksimali bendra informacija (angl. *maximum mutual information*) ir minimali diskriminantinė informacija (angl. *minimum discrimination information*), taip pat naudojami vertinant PMM parametrus [33]. Pastebėta, kad didžiausio tikėtinumo įverčio taikymas yra paplitęs dėl palankių statistinių savybių – stabilumo ir asimptotinio normalumo (angl. *asymptotic normality*), įrodytų bendromis sąlygomis [34, 35].

Plačiau didžiausio tikėtinumo įvertis nagrinėjamas tolimesniuose šio skyriaus poskyriuose.



2 pav.: Ergodinio trijų būsenų PMM su diskrečiaisiais išvesties stebėjimais (viršuje). Diskretusis PMM su N-paslėptųjų būsenų ir R-skirtingų stebėjimų eina tarp paslėptųjų būsenų s_t ir generuoja stebėjimus o_t (apačioje).

2.2 Didžiausio tikėtinumo įverčiai ir MVM algoritmas

2.2.1 Didžiausio tikėtinumo metodas

Didžiausio tikėtinumo metodas (angl. *maximum likelihood method*, santr. DTM) – metodas, skirtas statistinio modelio parametrams vertinti. Fiksuotai nepriklausomų vienodai pasiskirsčiusių duomenų imčiai DTM leidžia gauti modelio parametrų rinkinių reikšmes, maksimizuojančias tikėtinumo funkciją. Šiuo metodu randamos tokios parametrų reikšmės, su kuriomis gaunamieji rezultatai tampa labiausiai tikėtini duotajam modeliui [36, 37].

Kuo imtis didesnė, tuo labiau tikėtina, kad didžiausio tikėtinumo metodu gauti parametrų įverčiai mažai skirsis nuo tikrųjų parametro reikšmių. Esant gana bendroms sąlygoms [38–40], gauti įverčiai yra:

- pagrįsti (konverguoja pagal tikimybę į nežinomo ir vertinamo parametro reikšmę),
- asimptotiškai normalieji,
- efektyvūs (asimptotiškai turi mažiausią dispersiją tarp visų galimų nežinomo parametro įverčių).

Tarkime, duota n atsitiktinių vektorių $\mathbf{O} = (O_1,O_2,\ldots,O_n)$, kurių skirstinys priklauso nuo fiksuotų nežinomųjų parametrų $\theta = (\theta_1,\ldots,\theta_k)$. Tuomet bet kuriam $O_i \in \mathbf{O}$ individuali tikimybinio tankio funkcija (angl. *probability density function*) yra aprašoma O_i $f(o_i|\theta), i = 1,\ldots,n$.

Atsitiktinio dydžio tankio funkcijų sandauga aprašoma $f(o|\theta) = \prod_{i=1}^{n} f(o_i|\theta_1,\ldots,\theta_k)$, čia $o = (o_1,\ldots,o_n)$ yra stebimi vektorių $O_1,\ldots,O_n$ dydžiai arba reikšmės.

Norint apskaičiuoti didžiausio tikėtinumo įvertį duotam skirstiniui reikia aprašyti jo tikėtinumo funkciją. Tarkime, duoti stebimi dydžiai $\mathbf{O} = o$, tuomet tikėtinumo funkcija $L(\theta|o) = L(\theta_1,\ldots,\theta_k|o_1,\ldots,o_k) = f(o|\theta) = \prod_{i=1}^{n} f(o_i|\theta_1,\ldots,\theta_k)$.

$f(o|\theta)$ atžvilgiu tariama, kad θ yra nežinomas fiksuotas dydis, o o gali įgyti bet kurią reikšmę iš stebėjimų erdvės. Kalbant apie tikėtinumo funkciją $L(\theta|o)$, tariama, kad o yra žinomas fiksuotas dydis, o θ gali įgyti bet kurią reikšmę iš parametrų erdvės Θ.

Didžiausio tikėtinumo įverčiai yra tos parametrų θ reikšmės, su kuriomis funkcija $L(\theta)$ įgyja didžiausią reikšmę, o parametrų įvertis laikomas didžiausio tikėtinumo įverčiu [38].

Tarkime, duotos parametrų reikšmės $\theta',\theta'' \in \Theta$. Norint nustatyti, kurios parametrų reikšmės labiau tikėtinos stebėjimui o, reikia naudotis tikėtinumo funkcija. Jeigu $L(\theta'|o) > L(\theta''|o)$, sakoma, kad stebėjimui o yra labiau tikėtina parametro reikšmė θ' už θ'', kadangi su ja gauta tikėtinumo funkcijos reikšmė yra didesnė.

Sakykime, kad stebėjimui o aprašomas $\hat{\theta}(o) \in argmax_{\theta\in\Theta} L(\theta|o)$. Tuomet $\hat{\theta}(o)$ yra laikomas didžiausio tikėtinumo įverčiu parametrui θ stebint o. Čia $argmax L(\theta|o)$ aprašo aibę visų $\theta \in \Theta$ reikšmių, kurios maksimizuoja $L(\theta|o)$ visoje parametrų erdvėje Θ. Kadangi $L(\theta|o)$ negalima maksimizuoti analitiškai, reikia remtis skaitiniais metodais.

Funkcija $L(\theta)$ dažnai gali būti logaritmuojama skaičiavimams supaprastinti. Jei tikėtinumo funkcija diferencijuojama, tai funkcijos $-\ln L(\theta)$ minimumas ieškomas taip:

- randamos dalinės išvestinės $\frac{\partial L(\theta)}{\partial \theta_j}, j = 1,2,\ldots,n$;
- išvestinės prilyginamos nuliui ir sprendžiama lygčių sistema $\frac{\partial L(\theta)}{\partial \theta_j} = 0$, $j = 1,2,\ldots,n$, su n lygčių ir n nežinomųjų. Ši lygčių sistema dažnai turi vienintelį sprendinį;

- remiantis didžiausio tikėtinumo įverčių asimptotiniu normališkumu apskaičiuojami gautų didžiausio tikėtinumo įverčių pasikliautinieji intervalai. Norint iteratyviai apskaičiuoti parametrų didžiausio tikėtinumo įverčius skaitiniais metodais, reikia parinkti pradines įverčių reikšmes ir atlikti tam tikrą iteracijų kiekį, kol gretimų iteracijų įverčiai pradės skirtis gana mažai.

Matematinės vilties maksimizavimo (angl. *expectation-maximization, EM algorithm*) algoritmas iteratyviai randa statistinio modelio parametrų didžiausio tikėtinumo įverčius, kai modelis priklauso nuo nestebimų latentinių kintamųjų. MVM algoritmas dažnai naudojamas neprižiūrimo mokymosi procese. Jis paremtas tikimybiniu modelio kūrimu su iš dalies stebimais kintamaisiais.

Bendra algoritmo struktūra susideda iš dviejų dalių. Pirmoji dalis yra E-žingsnis, apskaičiuojantis logaritminės tikėtinumo funkcijos sąlyginį vidurkį, o antroji dalis – M-žingsnis, maksimizuojantis E-žingsnyje apskaičiuotą tikėtinumo funkcijos sąlyginį vidurkį.

2.2.2 MVM algoritmas bendru atveju

Latentinio (angl. *latent*, tiesiogiai nestebimo) kintamojo modelis yra statistinis modelis, kurį sudaro dviejų tipų kintamieji: stebimi kintamieji ir latentiniai kintamieji. Stebimi kintamieji yra tie, kuriuos galime išmatuoti ar užfiksuoti, o latentiniai (kartais dar vadinami paslėptaisiais) kintamieji yra tie, kurių negalime tiesiogiai stebėti, tačiau yra susiję su stebimais kintamaisiais.

MVM algoritmas gali būti taikomas didžiausio tikėtinumo įverčiams apskaičiuoti praleistųjų (angl. *missing*) ar neišsamiųjų (angl. *incomplete*) duomenų atveju. Praleistosios reikšmės atsiranda tada, kai reikšmė yra klaidinga, atsitiktinai neužregistruojama, ribojamos (techninės ar fizinės) galimybės arba tyrimo kintamasis tiesiogiai negali būti stebimas. Ši iteratyvi optimizavimo procedūra išsiskiria iš bendrų optimizavimo algoritmų savo santykiniu stabilumu ir mažesniu jautrumu pradinėms reikšmėms parinkti.

Tarkime, visa duomenų aibė yra aprašoma rinkiniu $Z = (O,Y)$. Stebima duomenų dalis yra O (ji gali būti sudaryta iš M-mačių atsitiktinių vektorių, pasiskirsčiusių pagal tam tikrą skirstinį), o nestebima duomenų dalis – Y.

Tegul $f(z|\theta) = f(o,y|\theta)$ aprašo atsitiktinių kintamųjų O ir Y jungtinį pasiskirstymo tankį (angl. *joint pdf*) ir $f(O|\theta) := \int f(o,y|\theta)d\mu_y(y)$ aprašo atsitiktinio kintamojo O marginalinį pasiskirstymo tankį (angl. *marginal pdf*) parametro μ_o atžvilgiu, kur θ yra nežinomas parametras, $\theta \in \Theta \subset \mathbf{R}^k$.

Funkcija $l(\theta|O) := \ln f(O|\hat{\theta})$ vadinama daline logaritminio tikėtinumo funkcija, nes ji yra tik stebimų duomenų dalies O logaritminio tikėtinumo funkcija. Funkcija $l(\theta) = l(\theta|O,Y) = \ln f(o,y|\theta)$ yra visa logaritminio tikėtinumo funkcija (ji yra visų duomenų – tiek stebėtų, tiek ir nestebėtų – logaritminio tikėtinumo funkcija). Čia θ yra nežinomas parametrų vektorius, kuriam norime surasti didžiausio tikėtinumo įverčius.

Nustatoma visa logaritminio tikėtinumo funkcija:

$$Q(\theta;\bar{\theta}) := \mathbf{E}_{\bar{\theta}}\left[\ln f(O,Y|\theta)|O\right] = \mathbf{E}_{\bar{\theta}}\left[l(\theta)|O\right], \bar{\theta},\theta \in \Theta.$$

Pagal Jenseno nelygybę:

$$Q(\theta;\bar{\theta}) - Q(\bar{\theta};\bar{\theta}) = \mathbf{E}_{\bar{\theta}}\left[\ln \frac{f(O,Y|\theta)}{f(O,Y|\bar{\theta})}|O\right] \leq \ln \mathbf{E}\left[\frac{f(O,Y|\theta)}{f(O,Y|\bar{\theta})}|O\right] = l(\theta|O) - l(\bar{\theta}|O).$$

Ir

$$Q(\theta,\bar{\theta}) - Q(\bar{\theta},\bar{\theta}) \leq l(\theta) - l(\bar{\theta}).$$

Apibrėžiama

$$\theta^* = \arg\max_{\theta\in\Theta} Q(\theta,\bar{\theta}).$$

Tuomet

$$0 \leq Q(\theta^*,\bar{\theta}) - Q(\bar{\theta},\bar{\theta}) \leq l(\theta^*|O) - l(\bar{\theta}|O).$$

Į šią nelygybę įstačius didžiausio tikėtinumo įvertį (DTI) $\hat{\theta} := \bar{\theta}_{DTI}$ vietoje $\bar{\theta}$,

$$\hat{\theta}_{DTI} = \arg\max_{\theta\in\Theta} l(\hat{\theta}|O),$$

gaunama nelygybė

$$0 \leq Q(\theta^*,\hat{\theta}) - Q(\hat{\theta},\hat{\theta}) \leq l(\theta^*|O) - l(\hat{\theta}|O) \leq 0.$$

Taigi $Q(\theta^*,\hat{\theta}) = Q(\hat{\theta},\hat{\theta})$. Nesunku pastebėti, kad θ^* yra stacionarusis nagrinėjamo iteracinio proceso taškas, o iteracijos $\theta^0 = \bar{\theta}, \theta^1 = \theta^*$ ir t. t. konverguoja į $\hat{\theta}_{DTI}$.

MVM algoritmas siekia rasti didžiausio tikėtinumo įverčius iteratyviai vykdydamas šiuos žingsnius:

- **Tikėtinumo žingsnis (E-žingsnis)**: nestebimi duomenys įvertinami remiantis stebėjimo duomenimis ir turimais modelio parametro įverčiais.

Apskaičiuojama logaritminės tikėtinumo funkcijos reikšmė $l(\theta|O,Y)$, remiantis stebimais duomenimis O, su turimais parametrų įverčiais $\hat{\theta}$.

$$Q(\theta,\bar{\theta}) := E_{\bar{\theta}}\left[l(\theta)|O\right].$$

- **Maksimizavimo žingsnis (M-žingsnis)**: tikėtinumo funkcijos reikšmė maksimizuojama darant prielaidą, kad nestebimi duomenys yra žinomi. Reikia rasti parametro θ įverčius, su kuriais maksimizuojama funkcija:

$$Q(\theta,\bar{\theta}) \rightarrow \max_{\theta\in\Theta}.$$

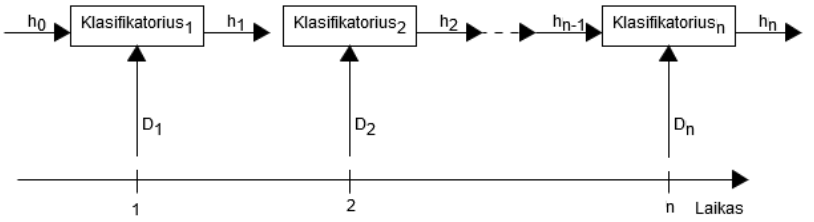
Abu žingsniai kartojami, kol pasiekiamas funkcijos maksimumas.

2.3 Palaipsnis algoritmas ir jo taikymai

PMM parametrų vertinimas jau beveik pusę amžiaus yra svarbus tyrimų objektas, kadangi šiuos taiko daugelis disciplinų, tokių kaip statistika [41], mašininis mokymasis [42, 43], signalų ir vaizdų apdorojimas [44], signalų sekimas [45, 46], prognozavimas [47], ir kt. Standartiniai PMM parametrų vertinimo metodai yra grindžiami rinkinio (angl. *batch*) mokymu, naudojant tikėtinumo-maksimizavimo metodus [24], tokius kaip „Baum-Welch“ algoritmas [21] ar skaitinio optimizavimo metodai. Šiais atvejais PMM parametrai yra vertinami atlikus kelias mokymo iteracijas, kol tikslo funkcija (pvz., didžiausio tikėtinumo funkcija) maksimizuojama. Nors mašininio mokymosi sritis pastaruoju metu skatina domėtis Bajeso metodais, pagal kuriuos parametrai laikomi atsitiktiniais kintamaisiais, dauguma PMM parametrų vertinimo metodų paremti ne Bajeso metodais (parametrai laikomi deterministiniais nežinomaisiais). Naujausi ne Bajeso PMM parametrų vertinimo metodai skirti taikomosioms programoms, kai pradinė stebėjimų duomenų aibė yra ribota, o papildomi stebėjimai atsiranda nuosekliai laike.

Iš jų yra metodai, pagrįsti MVM, skaitiniu optimizavimu ir palaipsniu vertinimu, kuriuose daroma prielaida, kad stebėjimai gaunami kaip duomenų srautas. Kai kurie iš šių metodų skirti atnaujinti PMM parametrus pasimboliui (angl. *symbol-wise*), o kiti atnaujina parametrus pablokiui (angl. *block-wise*). Palaipsniai simbolinio mokymosi metodai (dar vadinami rekursiniais arba nuosekliais vertinimo metodais) sukurti situacijoms, kai mokymui skirti stebėjimai gaunami po vieną laike, o PMM parametrai pakartotinai vertinami stebint ir apdorojant kiekvieną naują

stebėjimą. Palaipsniai blokinio mokymosi metodai yra skirti situacijoms, kai mokymo stebėjimai yra suskirstyti į blokus (viena ar kelios stebėjimų sekos), o PMM parametrai pakartotinai vertinami tik apdorojus kiekvieną naują stebėjimų sekos bloką. 3 paveikslėlyje pavaizduotas bendras palaipsnio mokymosi scenarijus, kai duomenų blokai yra naudojami atnaujinti klasifikatoriaus parametrams per tam tikrą laiko tarpą. Tegul $D_1, D_2, \ldots, D_n$ yra mokymo duomenų blokai, prieinami klasifikatoriui diskrečiaisiais laiko momentais $t_1, t_2, \ldots, t_n$. Klasifikatorius pradedamas nuo pradinės hipotezės h_0, kuri atitinka pirmiau turėtas žinias. Tuomet h_0 yra atnaujinama į h_1 remiantis duomenimis D_0, o h_1 yra atnaujinamas į h_2 remiantis duomenimis D_2 ir t. t. Bet kuriuo atveju PMM parametrai atnaujinami apdorojus naujus mokymo duomenis, nereikalaujant prieigos prie anksčiau apdorotų duomenų ir galbūt nepažeidžiant anksčiau įgytų žinių [22, 48].



3 pav.: Bendras palaipsnio mokymosi scenarijus.

Pagrindinis palaipsnių metodų taikymo inkrementiniam mokymuisi privalumas yra gebėjimas išlaikyti aukštą stebėjimų apdorojimo lygį, nedidinant kompiuterio atminties resursų poreikio, nes nereikia saugoti duomenų iš ankstesnių mokymo etapų. Palaipsnis mokymas taip pat sutrumpina laiką, reikalingą apdoroti ir išmokti naujus duomenis, kadangi mokymas atliekamas tik su naujomis mokymo sekomis, o ne su visais sukauptais duomenimis.

Šiose programose paprastai pageidaujama atnaujinti parametrų įverčius naudojant palaipsnius (arba rekursinius) PMM parametrų vertinimo metodus, kurie iškart apdoroja gautus stebėjimus, tad jų nereikia saugoti kompiuterio atmintyje ar kelis kartus pakartotinai apdoroti. Šie palaipsniai PMM parametrų vertinimo metodai skiriasi nuo labiausiai išplėtotų rinkinio PMM parametrų vertinimo metodų, kurie atnaujina parametrų įverčius tik tada, kai stebėjimai išsaugomi kompiuterio atmintyje. Rinkinio algoritmui numatoma, kad modelio parametrams vertinti yra prieinamas fiksuoto dydžio mokymo stebėjimų rinkinys. Naujo gauto mokymų duomenų rinkinio šis algoritmas negali pritaikyti PMM, jei PMM iš naujo neapmokomas

duomenimis, gautais agregavus naują duomenų rinkinį su jau turėtais duomenimis. Kitu atveju prarandamos jau turėtos žinios.

Daugelio mokslinių straipsnių autoriai išnagrinėjo tik bendruosius tikimybinių modelių teorinius požymius [49], algoritmai iš esmės yra scheminiai [9, 10, 50] ir skaitiniai eksperimentai pateikiami su vienmačiais duomenimis [3, 22]. [9] straipsnyje eksperimentai atliekami su jame siūlomu algoritmu, kai ir stebėjimai, ir modelio parametrai yra vienmačiai (skaitiniu modeliavimo būdu algoritmas įvertintas su dviejų būsenų PMM); [24] pristatė palaipsnį MVM algoritmą, naudodamasis modeliavimu su vienmačiu, diskrečiuoju, dviejų būsenų PMM modeliu. Nors dažnai pateikiamos teorinės palaipsnių algoritmų koncepcinės diagramos, šių algoritmų konvergavimas nėra tiriamas, tad jų vertė lieka neaiški [51].

Pagal [52] straipsnyje atliktą apžvalgą, esami PMM parametrų vertinimo metodai gali būti suskirstyti į tris grupes pagal savo tikslo funkcijas. Pagrindinės identifikuotos metodų grupės yra:

- metodai, pagrįsti tikėtinumo tikslo (angl. *likelihood objective*) funkcijomis [9, 22, 24–27, 53–55];
- metodai, pagrįsti prognozavimo klaidos tikslo funkcijomis [26–28];
- metodai, pagrįsti modelio divergencijos funkcijomis [56–58];

Kiekvienoje iš šių grupių yra įvairūs metodai, taikantys skirtingas optimizavimo technikas ir turintys kitokius teorinius konvergavimo rezultatus. Toliau aptariami metodai su tikėtinumo tikslo funkcijomis.

2.3.1 Minimalios modelio divergencijos metodai

Minimalios modelio divergencijos (angl. *Minimum Model Divergence*) metodų atveju tikslo funkciją sudaro logaritminio tikėtinumo maksimizavimas ir parametrų divergencijos minimizavimas, naudojant kai kurias entropijos priemones. Šia tema blokinei signalų analizei pasiūlyta dvigubų sąnaudų funkcija (angl. dual cost function), maksimizuojanti logaritminį tikėtinumą, kartu minimizuojanti PMM parametrų divergavimą, naudojant eksponentinę gradiento optimizavimo sistemą [56–58], o po to praplėsta ir simbolinei signalų analizei [59].

Blokinei analizei, remiantis eksponentiniu gradiento metodu, [57, 58] straipsnio autoriai diskrečiajam PMM pasiūlė tikslo funkciją, kuri minimizuoja

divergenciją tarp senųjų ir naujųjų PMM parametrų įverčių, taikant fiksuoto mokymosi greičio koeficientą. Šiuo atveju modelio parametrai atnaujinami apdorojus kiekvieną stebėjimų seką.

Simbolinei analizei Gargas ir Warmuthas [59] išplėtė Singerio ir Warmutho [57, 58] blokinei analizei skirtą modelio parametrų vertinimo algoritmą. Modelio divergencija priklauso nuo neigiamos logaritminio tikėtinumo didėjimo reikšmės, t. y. kiekvieno naujo stebėjimo logaritminio tikėtinumo, atsižvelgiant į visus ankstesnius stebėjimus. Kiekvienu laiko momentu apdorojant stebėjimus atliekamas optimizavimas, kurį galima atskirti į panašius parametrų atnaujinimus: būsenų perėjimo tikimybėms ir būsenų išvesties tikimybiniam skirstiniui.

2.3.2 Didžiausio tikėtinumo metodai

Populiariausi palaipsniai (ir rinkinio) PMM parametrų vertinimo metodai yra pagrįsti didžiausio tikėtinumo metodu. Įvairiuose literatūros šaltiniuose pasiūlytos didžiausio tikėtinumo metodo modifikacijos, taikant skirtingas skaitinio optimizavimo technikas.

Geros asimptotinės didžiausio tikėtinumo įverčio savybės paskatino daugelį bandymų praplėsti šį metodą palaipsniam mokymuisi. Tiksliau tariant, netiesioginis logaritminės tikėtinumo funkcijos maksimizavimas per visą logaritminį tikėtinumą buvo daugelio palaipsnių mokymosi algoritmų pagrindas.

Svarbi palaipsnių MVM algoritmų savybė yra lankstumas [60]. Nėra jokių apribojimų, kaip duomenys skirstomi į blokus – blokas gali būti vienas stebėjimas ar kelios stebėjimų sekos. Be to, modelio parametrus galima atnaujinti pritaikius įvairias duomenų apdorojimo schemas. Kaip to rezultatas, literatūroje pasiūlyti metodai išplėtė šį algoritmą ir pritaikė PMM parametrų palaipsniam vertinimui. Dauguma pasiūlytų metodų pradedami tankio inicializavimu nuliu ir tęsiami nuosekliai apdorojant mokymo duomenis (pagal simbolius / stebėjimus ar pagal blokus) ir atnaujinant PMM parametrus po kiekvieno stebėjimo ar bloko.

[60] straipsnio autoriai pasiūlė inkrementinę MVM algoritmo versiją, kad paspartintų konvergavimą baigtinio mokymo duomenų rinkinio atveju. Darant prielaidą, kad fiksuotas mokymo duomenų rinkinys padalijamas į kelis blokus, kiekviena šio algoritmo iteracija atlieka dalinį E-žingsnį (pasirinktam blokui), tada atnaujina modelio parametrus (M-žingsnis), kol pasiekiamas konvergavimo kriterijus. Patobulinta algoritmo versija yra susijusi su greitesniu naujos informacijos panaudojimu, kai atnaujinant parametrus nereikia laukti, kol apdorojami visi duomenys

[60, 61]. Daugelis šio algoritmo versijų taikomos palaipsniam blokiniam [62–64] arba simboliniam [54, 65, 66] mokymui. Tiesioginis logaritminės tikėtinumo funkcijos maksimizavimas pirmą kartą pasiūlytas rinkinio algoritmo atveju, naudojant gradiento nusileidimo algoritmą [67], o paskui kvazi-Niutono algoritmą [68] greitesniam konvergavimui. Taip pat Titteringtonas [69] ir Weinsteinas [70] nepriklausomų duomenų apdorojimui sukūrė simbolinį metodą, siekiantį kiekvienu laiko momentu nuosekliai optimizuoti visą duomenų tikėtinumą.

Pavyzdžiui, rinkinio „Baum-Welch“ algoritmas, realizuotas su PMM tiesioginio-atbulinio sklidimo (angl. *Forward-Backward*) algoritmu ir tikėtinumo maksimizavimu, įvertina modelio parametrus, lokaliai maksimizuojančius tikėtinumo tikslo funkciją [20, 32]. [9, 24] pasiūlė palaipsnius MVM algoritmus, kurie taiko skaitinio glotninimo (angl. *smoothing*) metodą. [10] pasiūlė fiksuoto intervalo glotninimu pagrįstą „Baum-Welch“ metodą. Jie taip pat pristatė eksponentinį užmiršimo faktorių (angl. *exponential forgetting factor*), kuriuo siekiama sumažinti senesnių PMM parametrų įtaką skaičiavimams, nustatant fiksuotą mokymo greitį.

[65] straipsnio autoriai pasiūlė naudoti prognozuojamą būsenos tankį kaip geresnę fiksuoto intervalo glotninimo aproksimaciją. Modelio parametrai nuosekliai atnaujinami, taikant Stengerio [3] rekursijos formules diskrečiajam PMM. Digalakis [62] pristatė palaipsnį MVM algoritmą, skirtą atnaujinti tolydaus PMM parametrus automatiniam šnekos atpažinimui, ir parodė greitesnį konvergavimą bei didesnį atpažinimo tikslumą už tradicinių algoritmų. Panašiai, Mizuno [63] pasiūlė panašų algoritmą automatiniam šnekos atpažinimui, naudojantį diskrečiuosius PMM.

Stengeris [66] „Baum-Welch“ algoritme pagal filtruojamos būsenos tankį aproksimavo nustatytą fiksuoto intervalo glotninimą, siekdamas atnaujinti tolydaus PMM parametrus ir taikydamas modeliuojamoje vaizdo sekų analizėje. Modelio būsenos tankis rekursyviai apskaičiuojamas nepriklausomai nuo apdorojamos sekos ilgio. Taip modelio parametrai atnaujinami kiekvienu laiko momentu.

Šių didžiausio tikėtinumo metodais paremtų parametrų vertinimo uždavinių analitiniai sprendimai yra sudėtingai pasiekiami paslėptuosiuose Markovo modeliuose [20, 32]. Todėl siūlomi įvairūs tikėtinumo metodai, paremti įvairiais skaitiniais optimizavimo metodais. Pavyzdžiui, rinkinio „Baum-Welch“ algoritmas naudoja PMM tiesioginio-atbulinio sklidimo ir MVM procedūras ieškodamas parametrų įverčių, kurie lokaliai maksimizuotų tikėtinumo funkciją [20, 32, 71]. „Baum-Welch“ algoritmas sėkmingai sprendžia PMM parametrų vertinimo uždavinį ir

paskatino keletą bandymų sukurti palaipsnius MVM algoritmus PMM parametrams vertinti.

Pagrindinis uždavinys kuriant PMM palaipsnius MVM algoritmus – apskaičiuoti reikalingas duomenų statistikas be tiesioginio-atbulinio sklidimo procedūros. [53] straipsnyje atbulinio sklidimo (angl. *backward*) procedūra praleista ir pasiūlytame palaipsnio PMM parametrų vertinimo metode realizuota tik tiesioginio sklidimo procedūra. Paskui sukurta sudėtinga palaipsnė baigtinės atminties aproksimacija tiesioginio-atbulinio sklidimo procedūrai [22] ir naudojama siūlant palaipsnį PMM parametrų vertinimą.

Visai neseniai [72] ir [24] pasiūlė palaipsnius MVM algoritmus, kurie naudoja skaitinę palaipsnę glotninimo procedūrą [51], pakeičiančią tiesioginio-atbulinio sklidimo procedūrą. Nors visi šie metodai (ypač palaipsniai MVM metodai [72] ir [24]) yra labai panašūs į rinkinio „Baum-Welch“ algoritmą, jų konvergavimo savybės yra prastai išnagrinėtos [72].

Kaip alternatyvą palaipsniams MVM algoritmams PMM parametrams vertinti kai kurie autoriai pasiūlė palaipsnius parametrų vertinimo algoritmus, pagrįstus didžiausio tikėtinumo metodu, kuriame tikėtinumo funkcija optimizuojama naudojant stochastinius gradiento metodus [25, 26, 73], Niutono (angl. *Newton*) metodus [44], spektrinį mokymąsi [74], momentais grįstus metodus [75] ir kt. Taip pat sukurti nauji PMM parametrų mokymosi metodai, naudojantys neneigiamas matricų faktorizacijas (angl. *nonnegative matrix factorization*) [11, 76, 77]. [26] darbe įrodyta, kad palaipsniai didžiausio tikėtinumo metodai konverguoja į tikėtinumo tikslo funkcijos lokalų maksimumą (taip pat nustatytos stipriosios ir silpnosios konvergavimo į maksimumą normos). Panašūs lokalūs konvergavimo rezultatai, esant negriežtoms sąlygoms, nustatyti ir straipsnyje [25].

Dauguma siūlomų metodų gerai atspindi klasikinį „Baum-Welch“ algoritmą, tačiau jų stabilumas, sudėtingumas ir konvergavimo savybės yra prastai ištirtos.

2.3.3 Numatomos klaidos metodai

Numatomos klaidos metodai (angl. *Prediction Error Methods*), kaip ir palaipsniai didžiausio tikėtinumo metodai, yra skirti PMM parametrams vertinti realiu laiku ir grindžiami stochastiniais gradiento metodais [26–28]. Užuot maksimizavę tikėtinumo tikslo funkciją (kaip daroma palaipsnių didžiausio tikėtinumo metodų atveju), numatomos klaidos metodai siekia minimizuoti skirtumą tarp prognozuojamų PMM stebėjimų ir tikrųjų stebėjimų (ar PMM būsenų ir tikrųjų PMM būsenų

įverčių) [26–28]. Iš tikrųjų [26, 27] darbuose anksčiau pasiūlyti metodai naudoja tikslo funkcijas, susijusias su numatomomis klaidomis tarp stebimų ir prognozuojamų PMM stebėjimų, o [28] darbe vėliau pasiūlyti metodai atsižvelgia į numatomą klaidą tarp įvertintų ir prognozuojamų būsenų.

Daugumai numatomų klaidų metodų nustatytos lokalaus konvergavimo savybės [26, 28], išskyrus [27] darbą. Svarbu pažymėti, kad šie metodai turi vienus iš didžiausių teorinių (lokalaus) konvergavimo greičių PMM parametrų palaipsnio įvertinimo uždavinyje [26, 28].

Stebėjimų ar PMM būsenų numatomos klaidos minimizavimas suteikia alternatyvias tikslo funkcijas, kurios pasiūlytos taikant PMM signalams apdoroti. Jas sudaro PMM generuojamų stebėjimų numatomos [22, 26] arba PMM filtruotos būsenos [28] paklaidos matavimas ir atnaujintų PMM parametrų įverčių pateikimas su kiekvienu nauju stebėjimu.

Rekursinė numatoma klaida (angl. *Recursive Prediction Error*, santr. RNK) pirmiausia pasiūlyta taikyti tolydžiojo diapazono Gauso-Markovo (angl. *Continuous Range Gauss-Markov*) procesui [78] ir bendram palaipsniam stochastinio gradiento algoritmui [79]. RNK taiko išplėstą mažiausių kvadratų metodo principą ne tik tiesinėms, bet ir netiesinėms funkcijoms. Ji nustato mažiausią numatomos klaidos kainos funkcijos lokalę (angl. *locale*) ir pateikia atnaujintus modelio parametrų įverčius su kiekvienu nauju stebėjimu. RNK suformuluota atsižvelgiant į mažiausią numatomos klaidos dispersiją, remiantis tuo laiko momentu turimu geriausiu modelio įverčiu.

Fordas ir Moore [28] pasiūlė RNK modelį, kuriame naudojami filtruotos būsenos įverčiai, tačiau tik tada, kai žinomos būsenų perėjimo tikimybės. Šie modeliai paskui išplėsti į rekursinį būsenų numatomos klaidos (angl. Recursive State Prediction Error, santr. RBNK) algoritmą [28], kai būsenų perėjimo tikimybės įvertinamos pagal nuosekliai gaunamus duomenis. RBNK lokalaus konvergavimo savybės analizė pateikiama naudojant paprastųjų diferencialinių lygčių metodą, sukurtą RNK metodams. LeGlandas ir Mevelis [26] taip pat įrodė rekursinio sąlyginio mažiausių kvadratų įverčio (angl. *Recursive Conditioned Least Squares Estimator*), kuris apibendrina RNK metodą, konvergavimą [27].

Būsenų numatomos klaidos metodai [28] darbuose parodė greitesnį konvergavimą už stebėjimų numatomos klaidos metodus, pasiūlytus [26, 27] darbuose (nors tai lėmė didesnes sudėtingų skaičiavimų sąnaudas [7]). Deja, daugumos prognozavimo klaidų metodų tikslo funkcijose gali būti daug lokalių (ne globalių) optimumų, trukdančių gauti gerus parametrų įverčius.

2.4 Skyriaus apibendrinimas

Atlikus palaipsnių PMM parametrų vertinimo algoritmų analizę apibendrinama:

- Literatūroje pasiūlyta įvairių palaipsnių PMM parametrų vertinimo metodų. Daugelyje straipsnių aprašyti tikėtinumo ir numatomos klaidos metodų konvergavimo rezultatai rodo, kad metodai gali konverguoti į lokalų ekstremumą, kuris skiriasi nuo tikrųjų (nežinomų) PMM parametrų. Pastebėta, kad savo lankstumu iš visų išsiskiria didžiausiu tikėtinumu paremti palaipsniai algoritmai.
- Daugelio mokslinių straipsnių autoriai išnagrinėjo tik bendruosius tikimybinių modelių teorinius požymius, paskelbti algoritmai iš esmės yra scheminiai, o skaitiniai eksperimentai pateikiami su vienmačiais duomenimis. Nors dažnai pateikiamos teorinės palaipsnių algoritmų koncepcinės diagramos, šių algoritmų konvergavimas nėra tiriamas ir jų vertė lieka neaiški.

3 PASLĖPTIEJI MARKOVO MODELIAI SU GAUSO PASISKIRSTYMAIS

Šiame skyriuje aprašomas pasiūlytas palaipsnis PMM parametrų vertinimo algoritmas, skirtas vertinti tolydžius stebėjimus, pasiskirsčiusius pagal Gauso dėsnį. Sukurtas algoritmas paremtas didžiausio tikėtinumo metodu, MVM algoritmu ir tiesioginio sklidimo procedūra. Jis reikalauja tik fiksuoto operacijų skaičiaus kiekviename žingsnyje, o žingsnis atitinka vieną stebėjimą. Modelio mokymas atliekamas su fiksuoto dydžio pradiniu duomenų rinkiniu, o paskesnis parametrų atnaujinimas vyksta su kiekvienu naujai gautu stebėjimu, šių nesaugant kompiuterio atmintyje. Pasiūlytasis palaipsnis algoritmas palygintas su klasikiniu rinkinio PMM parametrų vertinimo algoritmu. Ištirtos palaipsnio algoritmo efektyvumo kriterijaus savybė ir jos priklausomybė nuo pradinio mokymo duomenų rinkinio dydžio.

Kai kurios šio skyriaus dalys yra publikuotos [29, 80].

3.1 Paslėptųjų Markovo modelių matematinis modelis

Nagrinėjamame PMM paslėptųjų kintamųjų būsenų erdvė yra diskrečioji, stebėjimai yra tolydieji dydžiai, pasiskirstę pagal Gauso dėsnį. PMM parametrai yra būsenų perėjimo tikimybės ir būsenos išvesties tikimybės.

Kuriamas PMM matematinis modelis aprašomas nustatytais parametrais:

- T – stebėjimų sekos ilgis,
- N – PMM būsenų skaičius,
- **A** – būsenų perėjimų tikimybių matrica

$$\mathbf{A} = \begin{bmatrix} a_{11} & \dots & a_{1N} \\ \vdots & \dots & \vdots \\ a_{N1} & \dots & a_{NN} \end{bmatrix}, \tag{1}$$

- pradinio buvimo būsenoje tikimybių vektorius

$$\pi = \begin{bmatrix} \pi_1 & \dots & \pi_N \end{bmatrix}^T, \tag{2}$$

- tikimybės tankio funkcija (TTF)

$$N\left(\mu_s, \sigma_s\right) = \frac{1}{\sqrt{\left(2\pi\right)^M \left|\sigma_s\right|}} e^{-\frac{1}{2}\left(\mathbf{o}-\mu_s\right)^T \sigma_s^{-1}\left(\mathbf{o}-\mu_s\right)}. \tag{3}$$

Stebėjimus aprašo normalieji atsitiktiniai dydžiai su vidurkiais μ_s ir kovariacijų matricomis σ_s, $1 \leq S \leq N$:

$$\mu_s = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_M \end{bmatrix}, \sigma_s = \begin{bmatrix} \sigma_{11} & \dots & \sigma_{1M} \\ \vdots & \dots & \vdots \\ \sigma_{M1} & \dots & \sigma_{MM} \end{bmatrix},$$

čia M – elementų (dimensijų) skaičius.

Būsenos parametrų skirstinys gali būti normalusis arba kelių Gauso dėsnių mišinys. Skirtingų būsenų S_i ir S_j parametrai μ ir σ yra skirtingi, kai $i \neq j$. Pastebėsime, kad prielaida, jog kiekvienos būsenos skirstinys yra normalusis, o ne kelių Gauso dydžių mišinys, nemažina bendrumo. Iš tikrųjų, jeigu kurioje nors PMM būsenoje yra stebimas Gauso skirstinio mišinys su svoriais c_1,c_2, . . . ,c_l, šią grandinę galima pakeisti jai ekvivalenčia, pakeitus minėtą mazgą l kitu. Jeigu grandinėje yra kelios būsenos, kuriose stebimi signalai turi tuos pačius parametrus (μ ir σ), būsenas galima sujungti į vieną. Taigi dėl paprastumo laikysime, kad $l = 1$.

3.2 Palaipsnis Gauso PMM parametrų vertinimas didžiausio tikėtinumo metodu

PMM parametrai gali būti įvertinami iteratyviu būdu. Modelio parametrams įvertinti dažnai naudojamas didžiausio tikėtinumo metodas [81, 82]. Didžiausio tikėtinumo metodas aptartas tolydžiųjų atsitiktinių dydžių atveju.

Tarkime, stebime atsitiktinį dydį $\mathbf{o}$, kurio tankis $b\left(\mathbf{o}\right)$ priklauso nuo nežinomų parametrų. Stebėjimo tikimybių tankis užrašomas tokiu pavidalu:

$$\sum_{j=1}^{N} \pi_j b_j\left(\mathbf{o}\right), \tag{4}$$

čia π_j yra buvimo j-toje būsenoje tikimybė, o stebėjimo j-oje būsenoje tikimybė

yra aprašoma daugiamačiu Gauso dėsniu

$$b_j(\mathbf{o}) = N(\mu_j, \sigma_j). \tag{5}$$

Įveskime logaritminę tikėtinumo funkciją, aprašančią stebėjimą atskiroje būsenoje

$$l(\mathbf{o}, \mu, \sigma, \pi) = \frac{(\mathbf{o}-\mu)^T \sigma^{-1} (\mathbf{o}-\mu)}{2} + \ln\left(\frac{\sqrt{|\sigma|}}{\pi}\right), \tag{6}$$

į kurią įtraukta buvimo būsenoje tikimybė π. Pirmiausia, aprašoma

$$L(\mathbf{o}, \mu, \sigma, \pi) = \sum_{i=1}^{N} e^{-l(\mathbf{o}, \mu_i, \sigma_i, \pi)}. \tag{7}$$

Skirstinio parametrų įverčiai turi maksimizuoti logaritminę tikėtinumo funkciją:

$$l(\mathbf{o}, \mu, \sigma, \pi) \to \max_{\mathbf{o}, \mu, \sigma, \pi}. \tag{8}$$

Tuomet apskaičiuojamos $\ln L$ išvestinės pagal μ ir σ

$$(L_{\mu_i})' = \frac{e^{-l(\mathbf{o}, \mu_i, \sigma_i, \pi)} (\mathbf{o}-\mu_i) \sigma_i^{-1}}{\sum_i e^{-l(\mathbf{o}, \mu_i, \sigma_i, \pi)}}, \tag{9}$$

$$(L_{\sigma_i})' = \frac{e^{-l(\mathbf{o}, \mu_i, \sigma_i, \pi)} \left(\sigma^{-1} (\mathbf{o}-\mu)(\mathbf{o}-\mu)^T \sigma^{-1} - \sigma^{-1}\right)}{\sum_i e^{-l(\mathbf{o}, \mu_i, \sigma_i, \pi)}}. \tag{10}$$

Rastos išvestinės prilyginamos nuliui $(\ln L_{\mu_i})' = 0$ ir $(\ln L_{\sigma_i})' = 0$, gautos lygtys sprendžiamos μ ir σ atžvilgiu.

Šiame PMM modelyje vidurkių ir dispersijų įverčiams maksimizuoti panaudotas MVM algoritmas. MVM algoritmas atliekamas šiais žingsniais [21, 60, 82]:

- E-žingsnis: logaritminės tikėtinumo funkcijos sąlyginio vidurkio apskaičiavimas:
$$L\left(\theta^i\right) = E\left[logL\left(S^i|\theta\right)\right].$$
- M-žingsnis: tikėtinumo funkcijos sąlyginio vidurkio maksimizavimas:
$$\theta^{i+1} \to \max_{\theta} L\left(\theta^i\right).$$

Panaudojus didžiausio tikėtinumo metodą išvestos formulės parametrams įvertinti pagal sumas:

$$\bar{\mu_j} = \frac{\sum_{t=1}^{T} \left(\gamma_t\left(j\right) \cdot \mathbf{o}_t\right)}{\sum_{t=1}^{T} \gamma_t\left(j\right)}, \tag{11}$$

$$\bar{\sigma_j} = \frac{\sum_{t=1}^{T} \gamma_t\left(j\right) \cdot \left(\mathbf{o}_t - \bar{\mu_j}\right)\left(\mathbf{o}_t - \bar{\mu_j}\right)^T}{\sum_{t=1}^{T} \gamma_t\left(j\right)}, \tag{12}$$

$$\gamma_t\left(j\right) = \frac{e^{-l(\mathbf{o}_t, \bar{\mu_j}, \bar{\sigma_j}, \pi)}}{\sum_{i=1}^{N} e^{-l(\mathbf{o}_t, \bar{\mu_i}, \bar{\sigma_i}, \pi)}}, \tag{13}$$

$$\bar{a_{ij}} = \frac{\text{tikėtinas perėjimų skaičius iš būsenos } S_i \text{ į būseną } S_j}{\text{tikėtinas skaičius perėjimų iš būsenos } S_i}. \tag{14}$$

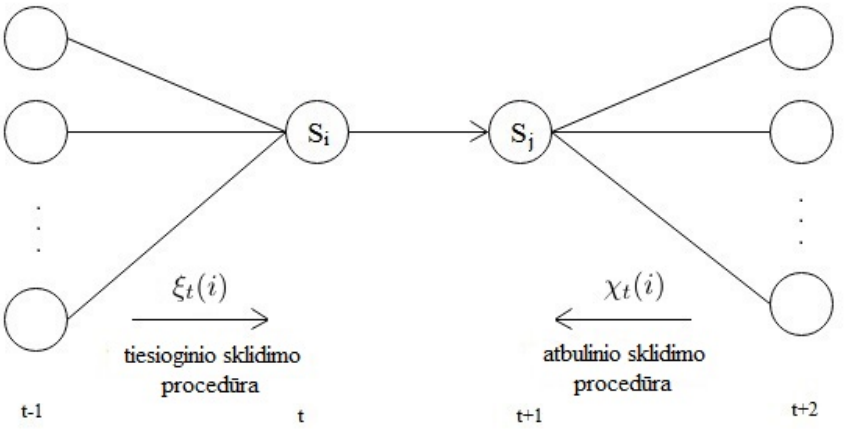
Tuomet $\bar{a_{ij}}$ įvertinimas atliekamas naudojant tiesioginio-atbulinio sklidimo procedūrą [32].

Tiesioginio-atbulinio sklidimo procedūros tikslas – surasti paslėptųjų būsenų sąlyginį skirstinį, atsižvelgiant į turimus duomenis (stebėjimus).

Tiesioginio-atbulinio sklidimo procedūra (žr. 4 pav.) yra PMM taikomas algoritmas, kuris apskaičiuoja visų paslėptųjų būsenų kintamųjų posteriorines ribas (angl. *posterior marginals*), gavęs stebėjimo seką $\mathbf{o}_1, \ldots, \mathbf{o}_T$, t. y. visiems paslėptųjų būsenų kintamiesiems $S_t \in \{S_1, \ldots, S_T\}$ apskaičiuoja skirstinį $P(S_t \mid \mathbf{o}_{1:T})$. Algoritmas naudoja dinaminio programavimo principą, kad efektyviai apskaičiuotų reikiamas vertes ir gautų posteriorinį ribinį skirstinį dviem sklidimais. Pirmasis tiesioginis sklidimas (angl. *forward*) eina laiku į priekį, o antrasis – eina laiku atgal (angl. *backward*).

Tiesioginio sklidimo procedūra – tai rekursinis algoritmas, skirtas apskaičiuoti $\xi_t(i)$ stebėjimo sekai, kurios ilgis t didėja. Pirmiausia, vienos stebėjimų sekos tikimybės apskaičiuojamos kaip pradinės i-tosios būsenos tikimybės ir duoto stebėjimo $\mathbf{o}_1$ generavimo i-joje būsenoje tikimybės sandauga. Tuomet taikoma rekursinė formulė. Tarkime, kad apskaičiavome $\xi_t(i)$ kai kuriems t. Norėdami apskaičiuoti $\xi_{t+1}(j)$, kiekvieną $\xi_t(i)$ turime padauginti iš atitinkamos perėjimo iš i-tosios būsenos į j-tąją būseną tikimybės, sudėti visų būsenų sandaugos rezultatus ir padauginti rezultatą iš stebėjimo $\mathbf{o}_{t+1}$ generavimo tikimybės. Kartodami procesą galiausiai galėsime apskaičiuoti $\xi_T(i)$ reikšmes, o susumavę visų būsenų reikšmes, gausime reikiamą tikimybę.

Simetriškas tiesioginio sklidimo kintamajam yra atbulinio sklidimo kintamasis $\chi_t(i)$, atitinkantis sąlyginę dalinės stebėjimo sekos nuo $\mathbf{o}_{t+1}$ iki pabaigos tikimybę,



4 pav.: Tiesioginio-atbulinio sklidimo (angl. *forward-backward*) procedūra PMM parametrų vertinime.

kurią sukuria nuo i-tosios būsenos prasidedančios visos būsenų sekos. Atbulinio sklidimo procedūra rekursyviai apskaičiuoja atbulinio sklidimo kintamuosius, einančius atgal išilgai stebėjimo sekos (nuo pabaigos iki pradžios).

Paprastai yra sunku realizuoti klasikinę tiesioginio-atbulinio sklidimo procedūrą palaipsniuose algoritmuose. Dėl to dažnai šios procedūros atbulinio sklidimo dalis yra praleidžiama arba taikomi kiti glotninimo metodai būsenų perėjimo tikimybėms skaičiuoti. Šiame darbe siūlome PMM būsenų perėjimo tikimybių vektorių apskaičiuoti integruojant Čapmano ir Kolmogorovo lygtį. Ši lygtis leidžia gauti tikimybių vektorių, nusakantį modelio buvimo būsenoje tikimybę laiko momentu t, jei laiko momentu $t-1$ modelis buvo būsenoje i [81, 83]:

$$\pi_t = \mathbf{A} \cdot \pi_{t-1}. \tag{15}$$

Pasiūlymo prasmė ir reikšmingumas tiriamas 5.1.2 skyrelyje.

Žymime būsenų perėjimo tikimybes kaip

$$\beta_t^i = \frac{1}{t} \sum_{i=1}^{t} \omega_t^i,$$

čia koeficientai

$$\omega_t^i = \frac{e^{-l(\mathbf{o}_t, \hat{\mu}_i, \hat{\sigma}_i, \pi_i)}}{\gamma_t},$$

$$\gamma_t = \sum_{i=1}^{N} e^{-l(\mathbf{o}_t, \hat{\mu}_i, \hat{\sigma}_i, \pi_i)}.$$

Siūlymas. Vidurkių vektoriaus ir kovariacijų matricos vertinimas (11) ir (12) gali būti atliekamas pateiktomis rekursinėmis formulėmis:

$$\mu_t^i = \mu_{t-1}^i + \frac{\left(\mathbf{o}_t - \mu_{t-1}^i\right)}{t} \cdot \frac{\omega_t^i}{\beta_t^i}, \tag{16}$$

$$\sigma_t^i = \left(\frac{\beta_{t-1}^i \cdot (t-1)}{\beta_t^i \cdot t}\right) \cdot \left(\sigma_{t-1}^i + \frac{\left(\mathbf{o}_t - \mu_{t-1}^i\right)\left(\mathbf{o}_t - \mu_{t-1}^i\right)^T}{t} \cdot \frac{\omega_t^i}{\beta_t^i}\right), \tag{17}$$

$$\beta_t^i = \beta_{t-1}^i + \frac{1}{t}\left(\omega_t^i + \beta_{t-1}^i\right). \tag{18}$$

Įrodymas. Iš tiesų nesunku įsitikinti, kad:

$$\beta_t^i = \beta_{t-1}^i + \frac{1}{t}\left(\omega_t^i + \beta_{t-1}^i\right) = \frac{1}{t-1}\sum_{i=1}^{t-1}\omega_t^i + \frac{1}{t}\left(\omega_t^i - \frac{1}{t-1}\sum_{i=1}^{t-1}\omega_t^i\right) = \frac{1}{t}\sum_{i=1}^{t}\omega_t^i. \tag{19}$$

Panašiai (16) ir (17) formulės atitinkamai seka iš (11) ir (12) formulių.

(16), (17), (18) formulės gali būti taikomos pradedant nuo tam tikros pradinės aproksimacijos. Palaipsnis MVM algoritmas, kitaip – palaipsnis Gauso PMM parametrų vertinimo algoritmas (santr. P_GPMM_PVA), susideda iš dviejų pagrindinių dalių. Pirma dalis yra modelio apmokymas, per kurį vykdomas pradinis parametrų vertinimas, kai žinomi tik stebėjimų vektoriai, taikant P_GPMM_PVA ir pasirinkus nedidelę fiksuoto dydžio stebėjimo imtį ((16), (17), (18) formulės). Šioje pradinio apmokymo stadijoje γ_t ir ω_t^i įverčiai apskaičiuojami naudojant fiksuotus pradinius vidurkio vektorių μ ir kovariacijų matricų σ reikšmes – $\hat{\mu}$ ir $\hat{\sigma}$ atitinkamai. Gavus pradinius įverčius atpažinimas ir pakartotinis modelio parametrų vertinimas toliau atliekamas nuosekliai, stebint srautu gaunamus duomenis ((16), (17), (18) formulės). Čia γ_t ir ω_t^i reikšmės apskaičiuojamos naudojant vidurkių vektorių μ ir kovariacijų matricų σ reikšmes, gautas pirmesnio žingsnio metu (t. y. μ_{t-1}^i ir σ_{t-1}^i atitinkamai).

Algoritmas yra stabdomas, kai paklaida yra mažesnė arba lygi santykiui Δ:

$$\bar{L} = \frac{\sum_{i=1}^{N} L_i}{N},$$

$$D^2 = \frac{1}{N}\sum_{i=1}^{N}(L_i - \bar{L})^2,$$

tuomet

$$\Delta \sim \frac{\sqrt{D}}{\sqrt{N}} \leq \epsilon.$$

Pradiniai įverčiai šiam algoritmui reikalingi norint užtikrinti jo stabilumą ir išvengti konvergavimo į išsigimusius lokaliuosius tikėtinumo funkcijos ekstremumus. Siūlomas palaipsnis algoritmas (P_GPMM_PVA) (1 algoritmas), kuris naudojamas tiek pradiniame modelio parametrų vertinime (2 algoritmas), tiek pakartotiniame parametrų vertinime (3 algoritmas), sudarytas iš (16), (17), (18) formulių ir apskaičiuoja parametrų įverčius kiekvienam duotam stebėjimui, nenaudodamas prieš tai apdorotų stebėjimų. Nesunku pastebėti, kad nuoseklios stebėjimų analizės atveju siūlomo P_GPMM_PVA algoritmo laiko sudėtingumas yra $O(n)$.

Reikia atkreipti dėmesį, kad sudaryto rinkinio MVM algoritmo sudėtingumas yra antrosios eilės. Iš tikrųjų nesunku pastebėti, kad esant fiksuotai imčiai didžiausio tikėtinumo įverčiams gauti reikės atlikti skaičiavimus, kurių apimtis yra proporcinga fiksuotam imties tūriui. Tokiu būdu parametrų vertinimą atliekant su kiekvienu stebėjimu, bendras reikalingų operacijų skaičius bus antros eilės. Stebint procesą (kurio parametrus norime įvertinti) reikalingų vertinimui operacijų skaičius labai didės, todėl vertinimas realiu laiku taps nebeįmanomas. Šiam uždaviniui spręsti išvestos rekursinės formulės PMM parametrams vertinti. Šios formulės neturi begalinių sumų ir yra išreikštos vidurkiais, kurie iteracijų skaičiui didėjant konverguoja į baigtines vertes. Šiuo atveju PMM modelio parametrai atnaujinami su kiekvienu gautu nauju stebėjimu, o ankstesnė mokymo aibė neįsimenama. Kiekvienoje P_GPMM_PVA algoritmo iteracijoje atliekamas baigtinis operacijų skaičius, priklausantis polinomiškai (antruoju laipsniu) nuo modelio parametrų skaičiaus, bet nepriklausantis nuo atliktų iteracijų skaičiaus. Šiuo atveju sukurtojo algoritmo sudėtingumas priklauso netiesiškai nuo atliktų iteracijų skaičiaus, t. y. $O(T)$. Klasikinis MVM algoritmas reikalauja kiekvienoje iteracijoje apdoroti visą turimą stebėjimų imtį, todėl algoritmo sudėtingumas tampa kvadratinis, t. y. $O(T^2)$.

1 algoritmas Kito žingsnio parametrų reikšmių apskaičiavimas (16)–(18) formulėmis.

procedure PARAMETER ESTIMATION(O, i, s, M, D, B)
    Įvestis: O – stebėjimas, i – stebėjimo eilė, s – tikėtinumo funkcijos rezultatas, M – vidurkių vektoriai, D – kovariacijų matricos, B – buvimo būsenoje tikimybių vektorius
    Išvestis: kito žingsnio parametrų reikšmės: $tempM$ – vidurkių vektoriai, $tempD$ – kovariacijų matricos, $tempB$ – buvimo būsenoje tikimybių vektorius
    $tfsum = 0$
    for $j = 1$ To N **do**
        $tfsum = tfsum + s_j$
    end for
    for $j = 1$ To N **do**
        $sk = \frac{s_j}{tfsum}$
    end for
    for $j = 1$ To N **do**
        $tempB_j = B_j + \frac{1}{i} \cdot (sk_j - B_j)$
        $tempM_j = M_j + \frac{O - M_j}{i} \cdot \frac{sk_j}{tempB_j}$
        $tempD_j = \frac{B_j \cdot (i-1)}{tempB_j \cdot i} \cdot \frac{D_j + (O - M_j)(O - M_j)^T}{i} \cdot \frac{sk_j}{tempB_j}$
    end for
    Grąžinti: $tempM$, $tempD$, $tempB$
end procedure

2 algoritmas P_GPMM_PVA algoritmo dalis: PMM parametrų vertinimas su fiksuota stebėjimų imtimi, kai gaunami pradiniai parametrų įverčiai.

procedure RECURSIVEEM(O, x, d, p, T, N, dim)

Įvestis: O – stebėjimų masyvas, x_n^u – pradiniai vidurkių vektoriai, d_n^u – pradinės kovariacijų matricos, $A_{n,n}$ – būsenų perėjimo tikimybių matrica, p_n^t – pradinio buvimo būsenoje tikimybių vektorius, T – stebėjimų skaičius, N – būsenų skaičius, dim – vektorių dimensijų skaičius, ϵ – stabdymo kriterijus, $1 \leq n \leq N, t = 0$.

Išvestis: vertinamų parametrų reikšmės: $\bar{x^t}$ – vidurkių vektoriai, $\bar{d^t}$ – kovariacijų matricos, $\bar{b^t}$ – buvimo būsenoje tikimybių vektorius, $1 \leq t \leq T$.

$tfsum = 0, u = 0, t = 0$

for $n = 1$ To N **do**

$\bar{x}_n^t = O_0, \bar{d}_n^t = emptymatrix[dim \times dim]$

$tempb_n = \exp(-l(O_0,x_n^u,d_n^u,p_n^t))$

end for

for $n = 1$ To N **do**

$b_n^u = \frac{tempb_n}{\sum_{n=1}^{N} tempb_n}$

end for

$u = 1$

while $(|x^u - x^{u-1}| \geq \epsilon$ and $|d^u - d^{u-1}| \geq \epsilon)$ **do**

for $t = 1$ To T **do**

$p^t = A^T \cdot p^{t-1}$

for $n = 1$ To N **do**

$s_n = \exp -l(O_t, x_n^{u-1},d_n^{u-1},p_n^t)$

end for

$[\bar{x}_t, \bar{d}_t, \bar{b}_t]$=PARAMETERESTIMATION($O_t$,$t$,$s$,$\bar{x}^{t-1}$,$\bar{d}^{t-1}$,$\bar{b}^{t-1}$)

end for

$[x^u, d^u, p^u] = [\bar{x}^T, \bar{d}^T, \bar{b}^T]$

$u = u + 1$

end while

Grąžinti: $\bar{x}, \bar{d}, \bar{b}$

end procedure

3 algoritmas P_GPMM_PVA algoritmo dalis: atpažinimas ir parametrų vertinimas atliekamas nuosekliai analizuojant srautu gaunamus duomenis, kai pradiniai parametrų įverčiai gauti su 2 algoritmu.

procedure RECURSIVEREESTIMATION(O, x, d, p, T, N, dim)
    Įvestis: O – stebėjimų masyvas, x_n – pradiniai vidurkių vektoriai, d_n – pradinės kovariacijų matricos, $A_{n,n}$ – būsenų perėjimo tikimybių matrica, p_n^t – pradinio buvimo būsenoje tikimybių vektorius, T – stebėjimų skaičius, N – būsenų skaičius, dim – vektorių dimensijų skaičius, $1 \leq n \leq N, t = 0$.
    Išvestis: vertinamų parametrų reikšmės: $\bar{x^t}$ – vidurkių vektoriai, $\bar{d^t}$ – kovariacijų matricos, $\bar{b^t}$ – buvimo būsenoje tikimybių vektorius, $1 \leq t \leq T$.
    $[x, d, b] = RECURSIVEEM(O,x,d,p,T,N,dim)$
    for $t = 1$ To T **do**
        $p^t = A^T \cdot p^{t-1}$
        for $n = 1$ To N **do**
            $s_n = \exp -l(O_t, x_n,d_n,p_n^t)$
        end for
        $[\bar{x}_t, \bar{d}_t, \bar{b}_t]$ = PARAMETERESTIMATION($O_t,t,s,\bar{x}^{t-1},\bar{d}^{t-1},\bar{b}^{t-1}$)
    end for
    Grąžinti: $\bar{x}_T, \bar{d}_T, \bar{b}_T$
end procedure

3.3 Eksperimentų rezultatai

Šiame skyrelyje pateikti skaičiavimo rezultatai gauti realizavus rinkinio ir P_GPMM_PVA algoritmus PMM parametrams vertinti.

Tarkime, kad yra duotas tam tikras mokymo duomenų rinkinys modelio parametrams vertinti pagal „Baum-Welch“ algoritmą. Jei gauname naują duomenų rinkinį modelio parametrams įvertinti, šį duomenų rinkinį galime sujungti su pirmiau naudotu duomenų rinkiniu. Tada skaičiavimai atliekami iš naujo su visu agreguotų duomenų rinkiniu. Modelio parametrus galime vertinti statiškai arba dinamiškai. Skirtumas tarp statinių ir dinaminių metodų priklauso nuo to, kaip elgiamės su žinomomis parametrų reikšmėmis. Parametrų vertinimas yra statinis, jei modelio parametrai vertinami naudojant pakeistą duomenų rinkinį, o visos ankstesnės parametrų vertės yra užmirštamos. Parametrų vertinimas yra dinaminis, jei žinomos parametrų reikšmės iš pirmesnio modelio mokymo naudojamos kaip pradinės vertės ir parametrų vertinimas atliekamas su pakeistu duomenų rinkiniu. P_GPMM_PVA algoritmas nepanaikina ankstesnių sukauptų žinių apie modelio parametrus. Taigi rezultatai gauti atlikus eksperimentus, kai modelio parametrai yra vertinami naudojant „Baum-Welch“ algoritmą ir statiniu, ir dinaminiu būdais.

Skaitiniai eksperimentai atlikti su kompiuteriu, turinčiu Intel Core i5 1,6 GHz procesorių ir 6 GB RAM. Algoritmai koduoti su Mathcad 14 versija. Gauti P_GPMM_PVA algoritmo veikimo rezultatai pritaikyti duomenų klasterizavimo uždaviniams.

Duomenų rinkiniai klasterizavimui atsitiktinai generuojami iš dviejų ($N = 2$) daugiamačių Gauso skirstinių, turinčių šias charakteristikas:

- Klasterio Nr. 1 centras: $\begin{bmatrix}500, & 500, & \ldots\end{bmatrix}$, Klasterio Nr. 2 centras: $\begin{bmatrix}600, & 600, & \ldots\end{bmatrix}$;
- Dimensijos: 2, 4, 8 ir 16;
- Standartinis nuokrypis: 20, 30, 40 ir 50;
- Būsenų perėjimo tikimybių matrica: $A = \begin{bmatrix}0{,}5 & 0{,}5\\ 0{,}5 & 0{,}5\end{bmatrix}$;
- Pradinio būsenų pasiskirstymo vektorius: $\pi = \begin{bmatrix}0{,}5 & 0{,}5\end{bmatrix}^T$;

Pradinis parametrų vertinimas atliktas su P_GPMM_PVA algoritmu, naudojant mažą fiksuoto dydžio ($t = 500$) mokymo stebėjimų rinkinį. Modelio parametrai inicializuoti atsitiktinai parenkant pradines jų reikšmes. Eksperimento rezultatai gauti apskaičiavus vidurkius iš atliktų 100 eksperimento pakartojimų. Algoritmo nutraukimo kriterijus fiksuotas visiems eksperimentams ir lygus $\epsilon = 0{,}01$. Jis nusako, kad gretimų iteracijų modelio parametrų įverčiai turi skirtis gana mažai.

3.3.1 Eksperimentų rezultatai: algoritmo skaičiavimo laikas

1 lentelėje pateikti skaičiavimo laiko rezultatai, gauti klasterizuojant sugeneruotą duomenų rinkinį P_GPMM_PVA algoritmu, ir palyginimui – rinkinio MVM algoritmu (statinio parametrų vertinimo atveju).

Pirmajame stulpelyje pateiktas stebėjimų skaičius (T); kiti stulpeliai rodo stebėjimo dimensijas ($M = 2, 4, 8, 16$) ir CPU laiką (Laikas) sekundėmis, gautą dviejų – P_GPMM_PVA (Palaipsnis) ir rinkinio MVM (Rinkinio) – algoritmų.

Šie eksperimentai sutelkti į įgyvendintiems algoritmams reikiamą skaičiavimo laiką, norint apdoroti klasterizavimo duomenų rinkinius. Apskaičiuotas vidutinis algoritmo veikimo laikas (sekundėmis) iš 100 eksperimentų pakartojimų. Laiko stulpelių lyginimas rodo, kad realizuotas P_GPMM_PVA algoritmas turi nuoseklų greičio pranašumą, lyginant su rinkinio MVM algoritmu (žr. 5 pav.). Net kai

1 lentelė: Algoritmų skaičiavimo laikas, gautas apdorojus klasterizavimo duomenų rinkinius.

T=	M = 2		M = 4		M = 8		M = 16		Sant.
	Pal	Rink	Pal	Rink	Pal	Rink	Pal	Rink	
1000	0,08	0,26	0,11	0,27	0,18	0,55	0,38	0,95	0,37
3000	0,13	0,8	0,17	0,79	0,28	1,5	0,58	2,66	0,20
5000	0,19	1,46	0,24	1,41	0,4	2,62	0,79	4,46	0,16
7000	0,24	2,15	0,31	2,02	0,51	3,77	0,98	6,37	0,14
10000	0,33	3,27	0,41	3,12	0,63	5,26	1,29	9,11	0,13

** Pal – palaipsnis algoritmas, Rink – rinkinio algoritmas, Sant. – vidutinio laiko santykis*

stebėjimų dimensijos didėja, P_GPMM_PVA algoritmo skaičiavimo laikas yra 3–9 kartus greitesnis už rinkinio MVM algoritmo.

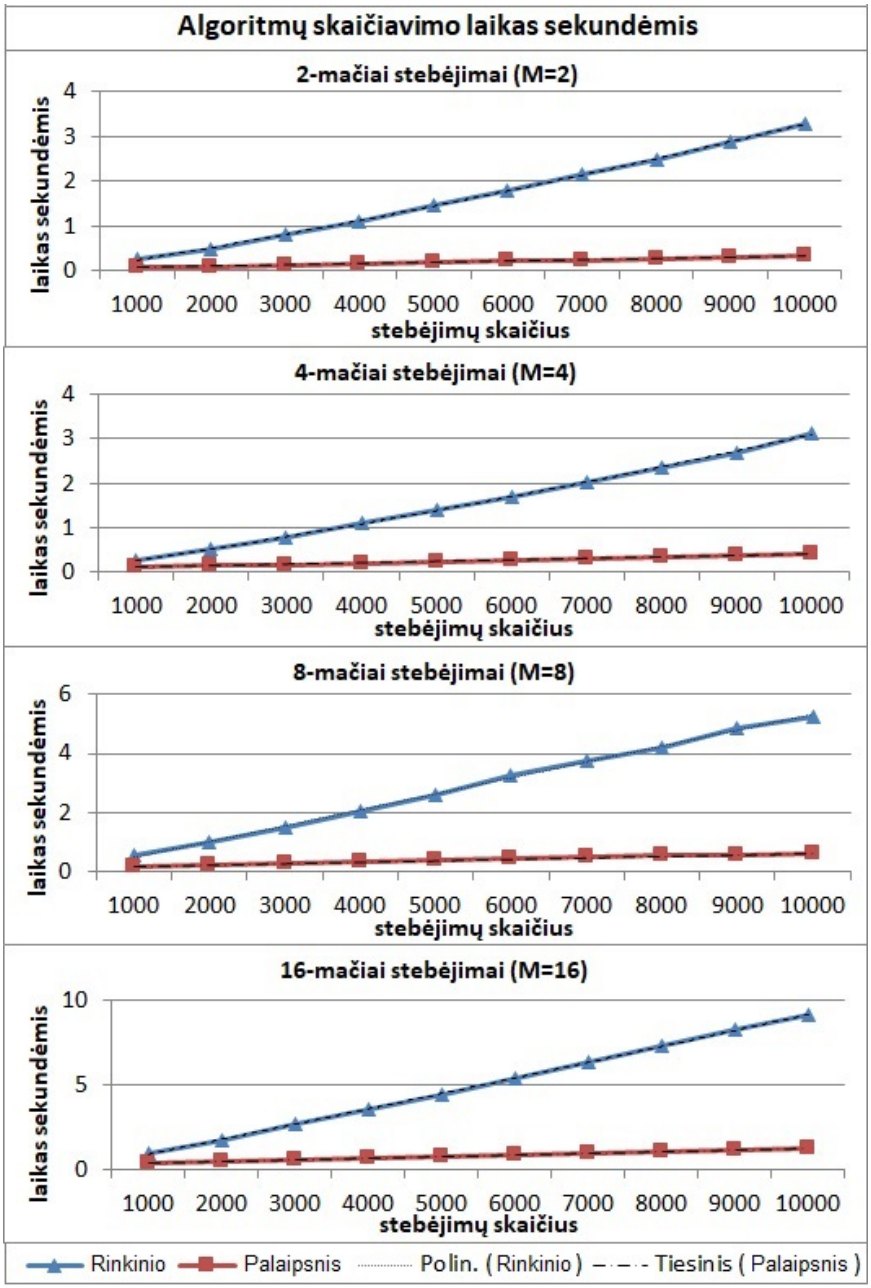
Rinkinio algoritmo dinaminio parametro vertinimo skaičiavimo laiko rezultatai pateikiami 2 lentelėje, kurioje rodomas duomenų rinkinio dydis (T), stebėjimų dimensijos (M) ir rinkinio MVM algoritmo veikimo laikas (Laikas), kai parametrai vertinami dinaminiu būdu. Taip pat rodomas atliktų pakartojimų skaičius (Pakartojimai) su kiekvienu duomenų rinkiniu (kiek kartų tikslo funkcija skaičiuojama iki jos maksimizavimo).

Šio eksperimento rezultatai rodo, kad esant gana mažam stebėjimų dimensijų skaičiui, pakartojimų, atliktų su kiekvienu duomenų rinkiniu, skaičius stabilizuojasi, kai T auga. Vis dėlto, didėjant stebėjimų dimensijoms, labai didėja pakartojimų skaičius, o skaičiavimų laikas, reikalingas tikslo funkcijai maksimizuoti, ilgėja.

2 lentelė: Rinkinio MVM algoritmo skaičiavimo laikas dinaminiam parametrų vertinimui.

T =	M = 2		M = 4		M = 8		M = 16	
	Laikas (s)	Pkart	Laikas (s)	Pkart	Laikas (s)	Pkart	Laikas (s)	Pkart
1000	0,31	6	0,29	5	0,56	7	1,48	9
3000	0,98	2	0,92	2	1,41	2	3,1	2
5000	2,19	2	2,16	2	3,07	2	6,15	2
7000	4,06	2	4,21	2	5,66	2	10,68	2
10000	8,35	2	9,05	2	11,56	2	20,5	2

*Pkart – pakartojimai



5 pav.: Rinkinio MVM (Rinkinio) ir P_GPMM_PVA (Palaipsnis) algoritmų skaičiavimo laikas sekundėmis. Juodos punktyrinės linijos (2-osios eilės polinomas) ir juodos taškinės linijos (tiesinis) atitinkamai vaizduoja palaipsnio ir rinkinio algoritmų tendencijų funkcijas.

3.3.2 Eksperimentų rezultatai: kriterijus δ

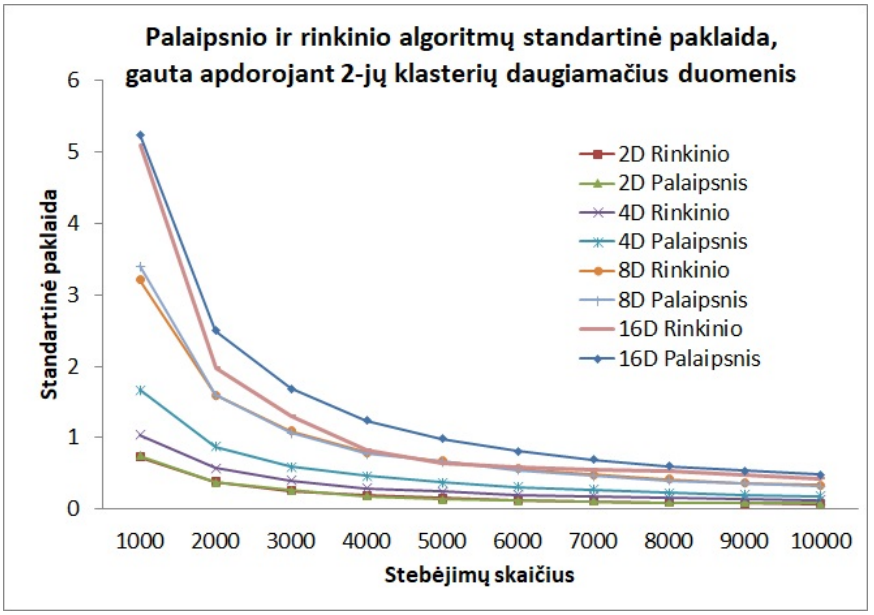
Svarbus PMM parametrų mokymo tikslas yra atkurti originalias šio modelio parametrų vertes. Šiame darbe ištirta, kaip gerai apmokytų modelių parametrų reikšmės artėja prie originalių parametrų reikšmių. Apibrėžiamas algoritmo kriterijus δ, kuris nurodo vidutinį atstumą nuo modelio parametrų įverčių iki tikrųjų parametrų verčių. Ištirta kriterijaus δ priklausomybė nuo algoritmo iteracijų skaičiaus. Empiriniu būdu tiriama hipotezė apie realizuoto P_GPMM_PVA algoritmo konvergavimą į uždavinio (8) sprendimą. Kitaip sakant, tiriama, ar kriterijus δ mažėja, kai didėja algoritmo iteracijų (apdorojamų stebėjimų) skaičius.

Apskaičiuojama PMM modelio parametrų įverčių standartinė paklaida – skirtumas tarp tikrųjų modelio parametrų verčių (parametrų vertės, naudojamos sugeneruoti duomenų rinkinius) ir įvertintų modelio parametrų reikšmių:

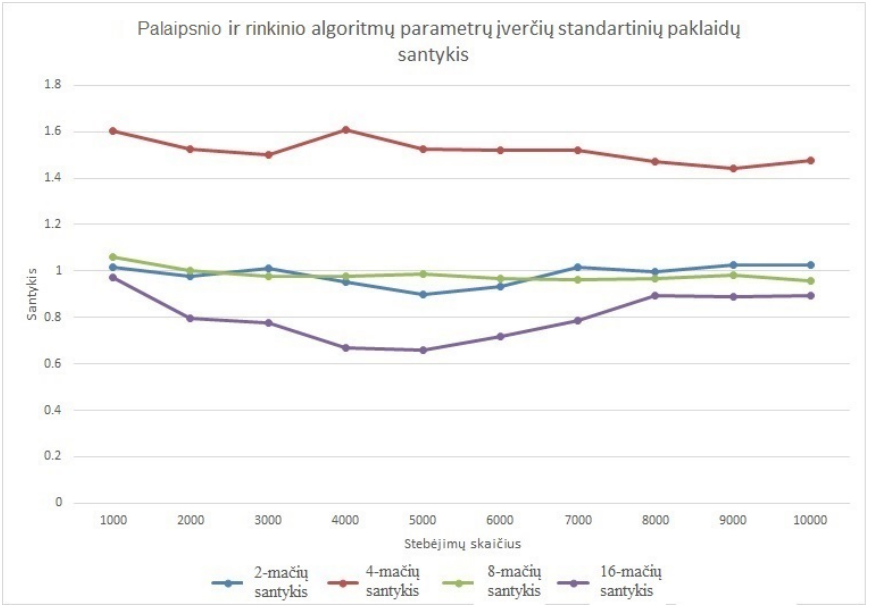
$$SP = \frac{1}{NM} \sum (x - \hat{x})'(x - \hat{x}).$$

6 paveiksle pavaizduoti eksperimentų rezultatai rodo, kad P_GPMM_PVA algoritmo kriterijus δ mažėja, kai didėja apdorotų stebėjimų skaičius (vienas stebėjimas apdorojamas per vieną algoritmo iteraciją), t. y. tiek rinkinio MVM algoritmu, tiek P_GPMM_PVA algoritmu gautos PMM parametrų įverčių standartinės paklaidos mažėja. Galime teigti, kad modelio parametrų įverčiai artėja prie tikrųjų parametrų verčių. Šiuo atveju taip pat matome, kad P_GPMM_PVA algoritmu gauti parametrų įverčiai artėja prie rinkinio algoritmu gautų parametrų įverčių. Kitaip tariant, rinkinio ir palaipsnio algoritmų parametrų įverčių standartinių paklaidų skirtumas yra minimalus.

Rinkinio MVM algoritmu gaunami parametrų įverčiai artėja prie originalių parametrų reikšmių. 7 paveiksle pateiktas abiejų algoritmų standartinių paklaidų santykis. Matyti, kad palaipsnis algoritmas kai kuriais atvejais efektyvumu nenusileidžia rinkinio algoritmui. Keturmačių stebėjimų apdorojimo atveju palaipsnio algoritmo standartinė paklaida yra $1{,}5$ karto didesnė už rinkinio algoritmo. Tačiau dvimačių ir aštuonmačių stebėjimų atveju rinkinio ir palaipsnis algoritmai gauna labai panašius modelio parametrų įverčius, kadangi abiejų algoritmų standartinė paklaida beveik nesiskiria. P_GPMM_PVA algoritmo gauta vidutinė standartinė paklaida trimis procentais (3 %) skiriasi nuo rinkinio MVM algoritmo.



6 pav.: P_GPMM_PVA (palaipsnis) ir rinkinio MVM (rinkinio) algoritmų standartinė paklaida, gauta su daugiamačiais dviejų klasterių duomenimis.



7 pav.: P_GPMM_PVA (palaipsnis) ir rinkinio MVM (rinkinio) algoritmais gautų PMM parametrų įverčių standartinių paklaidų santykis, kai klasterizuojami daugiamačiai duomenys, esantys dviejuose klasteriuose.

6 paveiksle parodyta, kad P_GPMM_PVA algoritmas kriterijaus δ požiūriu iš esmės nesiskiria nuo rinkinio MVM algoritmo. Tiek palaipsnis, tiek rinkinio MVM algoritmas nuosekliai gerina atpažinimo tikslumą, kai didėja mokymo duomenų kiekis. Duomenų sklaida ir dimensijų skaičius turi įtakos modelio parametrų įvertinimui, nes kai didėja T, didėja ir skirtumas tarp standartinių paklaidų, apdorojant 2-mačius ir 16-mačius duomenis.

Rinkinio ir palaipsnio algoritmų būsenos perėjimo tikimybių matricos, gautos parametrų vertinimo metu, pateikiamos 3 lentelėje, kurioje pavaizduotas duomenų rinkinio dydis (T), stebėjimų dimensijų skaičius (M) ir būsenų perėjimo tikimybių matrica.

3 lentelė: Rinkinio MVM ir P_GPMM_PVA algoritmų būsenos perėjimo tikimybių matrica, gauta parametrų vertinimo metu.

M = 2		
T=	Rinkinio MVM	P_GPMM_PVA
1000	$\begin{bmatrix} 0{,}50 & 0{,}50 \\ 0{,}47 & 0{,}53 \end{bmatrix}$	$\begin{bmatrix} 0{,}50 & 0{,}50 \\ 0{,}47 & 0{,}53 \end{bmatrix}$
10000	$\begin{bmatrix} 0{,}51 & 0{,}49 \\ 0{,}50 & 0{,}50 \end{bmatrix}$	$\begin{bmatrix} 0{,}51 & 0{,}49 \\ 0{,}50 & 0{,}50 \end{bmatrix}$

Būsenų perėjimo tikimybių matricų lyginimas rodo, kad realizuotas P_GPMM_PVA algoritmas kriterijaus δ požiūriu nesiskiria nuo rinkinio algoritmo. 3 lentelėje parodyta, kad apdorojus 1000 stebėjimų vektorių, būsenų perėjimo tikimybių matrica gaunama tokia pati tiek palaipsniam, tiek rinkinio algoritmams. Tokie patys rezultatai gauti apdorojus 10000 stebėjimų.

P_GPMM_PVA algoritmu gauti modelio parametrų įverčių paklaida nuo rinkinio MVM algoritmu gautų parametrų įverčių skiriasi iki 3 %, o būsenų perėjimo tikimybių matrica nesiskiria. Šis nedidelis parametrų įverčių skirtumas tarp dviejų algoritmų patvirtina, kad P_GPMM_PVA algoritmas, kaip ir rinkinio MVM algoritmas, yra efektyvus mokant PMM parametrus.

Empiriniu būdu gauti rezultatai patvirtina hipotezę apie algoritmo konvergavimą, t. y. algoritmo kriterijus δ mažėja, kai didėja apdorojamų stebėjimų skaičius. Tačiau ši algoritmo konvergavimo savybė ateityje turėtų būti tiriama teoriškai.

3.3.3 Eksperimentų rezultatai: algoritmo būsenų perėjimo tikimybių skaičiavimo efektyvumas

Šiame darbe atlikti eksperimentai, siekiant ištirti P_GPMM_PVA algoritmo PMM parametrų artėjimo į originalias parametrų reikšmes savybę. Realizuotas P_GPMM_PVA algoritmas PMM parametrų vertinimui lyginamas su [3] straipsnyje aprašytu algoritmu. Pagrindinis šio eksperimento tikslas buvo parodyti būsenų perėjimo tikimybių skaičiavimo pagal Čapmano ir Kolmogorovo lygtį poveikį algoritmo kriterijui δ. [3] straipsnyje aprašytas algoritmas pasirinktas palyginimui, nes jis įgyvendina klasikinę tiesioginio sklidimo procedūrą.

P_GPMM_PVA algoritmo kriterijui δ ištirti skaičiuojama įvertintų PMM parametrų standartinė paklaida kaip skirtumas tarp originalių parametrų reikšmių, naudotų generuoti stebėjimus, ir apskaičiuotų modelio parametrų verčių.

Eksperimentams sugeneruoti trys daugiamačių (3-mačių, 5-mačių, 12-mačių) stebėjimų vektorių duomenų rinkiniai ($T = 800$). Kiekviena PMM būsena modeliuojama 3-mačiais, 5-mačiais ir 12-mačiais vidurkių vektoriais ir kovariacijų matricomis. Stebėjimų sekai modeliuoti pasirinktas 5-ių būsenų PMM, kurio būsenų perėjimo tikimybių matrica nustatyta taip:

$$\mathbf{A} = \begin{bmatrix} 0 & 0{,}8 & 0{,}2 & 0 & 0 \\ 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 \\ 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 \\ 0 & 0 & 0 & 0{,}6 & 0{,}4 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Padinio būsenų pasiskirstymo vektorius nustatytas kaip: $\pi = \begin{bmatrix} 0 & 0{,}8 & 0{,}2 & 0 & 0 \end{bmatrix}^T$.

Algoritmo stabdymo kriterijus nustatytas kaip: $\epsilon = 0{,}01$.

Visi eksperimentai pakartoti 100 kartų.

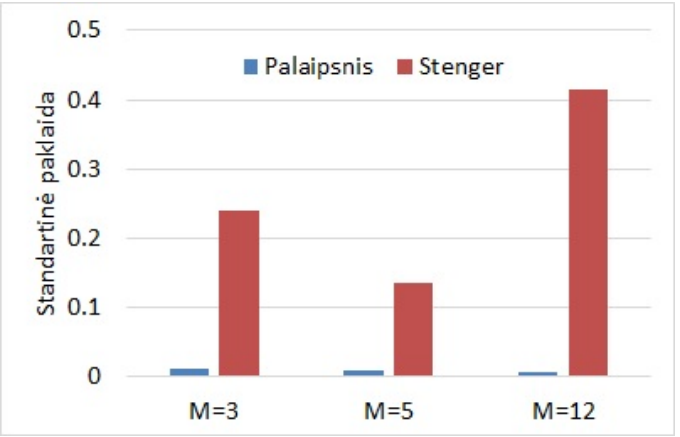
Eksperimentų rezultatai pateikti 4 lentelėje. Jie rodo, kad kai didėja stebėjimų dimensijų skaičius, skirtumas tarp apskaičiuotų parametrų įverčių ir originalių parametrų reikšmių yra gana mažas (tiek P_GPMM_PVA, tiek algoritmo iš [3] atveju). P_GPMM_PVA algoritmu gauta vidutinė modelio parametrų įverčių paklaida yra mažesnė už algoritmo iš [3] visų trijų generuotų duomenų rinkinių atveju (žr. 8 pav.), t. y. algoritmo P_GPMM_PVA kriterijus δ yra mažesnis už [3] algoritmo.

Šis eksperimentas rodo šio siūlomo būsenų perėjimo tikimybių skaičiavimo svarbą vertinant PMM parametrus palaipsniui. Būsenų perėjimo tikimybių skaičiavimas su Čapmano ir Kolmogorovo lygtimi pagerina parametrų įverčių artėjimą prie tikrųjų parametrų verčių, lyginant su algoritmu, kuriame realizuo-

ta tik tiesioginio sklidimo procedūra. 4 lentelėje pateikti rezultatai rodo, kad P_GPMM_PVA algoritmo gauta standartinė parametrų paklaida yra reikšmingai maža.

4 lentelė: Standartinė paklaida apskaičiuota modelio vidurkių vektoriams ir kovariacijų matricoms.

Algoritmas	**Parametrai**	$M = 3$	$M = 5$	$M = 12$
P_GPMM_PVA	μ	0,01	0,01	0,01
	σ	0,02	0,01	0,01
Stenger [3]	μ	0,20	0,17	0,72
	σ	0,28	0,10	0,11
Santykis	μ	0,04	0,04	0,01
	σ	0,06	0,12	0,04



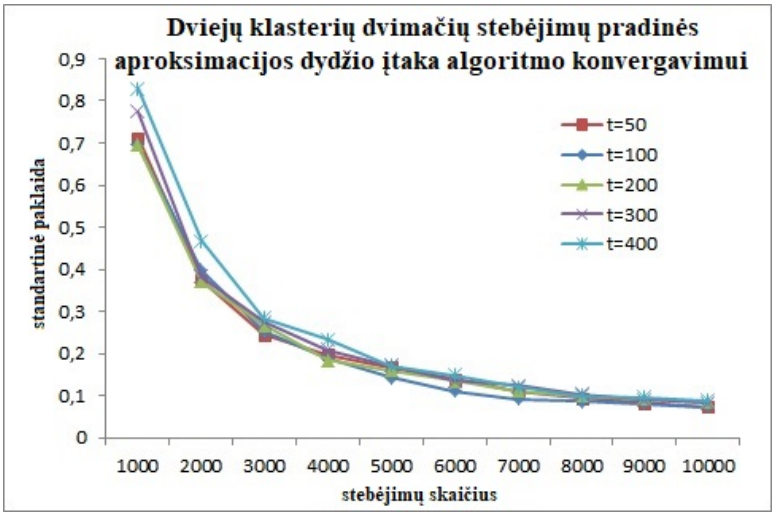
8 pav.: Vidutinė vidurkių vektoriaus μ ir kovariacijų matricos σ paklaida, gauta parametrus vertinant P_GPMM_PVA (palaipsnis) ir Stenger [3] algoritmais, kai stebėjimų vektorių dimensijų skaičius yra $M = 3$, $M = 5$ ir $M = 12$.

3.3.4 Eksperimentų rezultatai: algoritmo pradinės aproksimacijos dydžio analizė

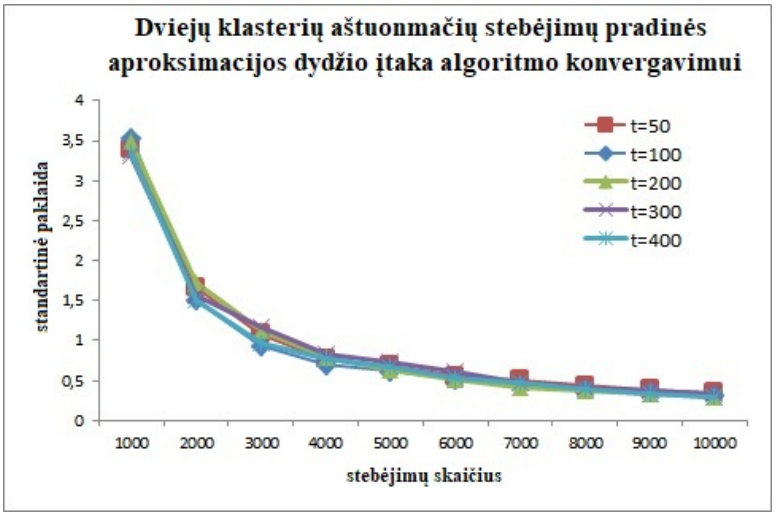
Papildomi eksperimentai atlikti su P_GPMM_PVA algoritmu, siekiant įvertinti pradinės aproksimacijos duomenų rinkinio dydžio įtaką algoritmo kriterijui δ. Eksperimentui sugeneruoti imti keli įvairių dydžių pradinių mokomųjų duomenų rinkiniai (t = 50, 100, 200, 300 ir 400), o skaičiavimai atlikti su dvimačiais ir aštuonmačiais stebėjimais, kurie sudaro du ir penkis klasterius.

Eksperimentų rezultatai rodo, kad pradinio mokymo duomenų rinkinio dydis neturi reikšmingo poveikio parametrų vertinimui su dvimačiais ir aštuonmačiais duomenimis dviejuose klasteriuose (žr. 9 ir 10 pav.). Esant nedideliam klasterių skaičiui, pakanka nedidelio skaičiaus duomenų rinkinio, reikalingo pradinei modelio parametrų aproksimacijai.

9 ir 10 paveiksluose pavaizduoti rezultatai taip pat rodo, kad PMM parametrų vertinimo duomenų rinkinio dydis, skirtas atpažinti ir parametrams vertinti (3 algoritmas), yra svarbus skaičiavimams. Algoritmo kriterijus δ (standartinė parametrų įverčių paklaida) mažėja, kai didėja bendras duomenų rinkinio dydis, net jei pradinės aproksimacijos duomenų rinkinio dydis yra mažas. Todėl parametrų įverčiai su kiekvienu nauju stebėjimu artėja prie originalių parametrų verčių, net jei duomenų kiekis pradiniam parametrų įvertinimui yra nedidelis. Nepaisant to, galima pastebėti, kad klasterių (būsenų) skaičius taip pat yra gana mažas. Tai rodo, kad esant nedideliam klasterių skaičiui, pradiniam mokymui naudojamo duomenų rinkinio dydis taip pat gali būti gana mažas, netgi tada, kai stebėjimo vektorių dimensijų kiekis didėja.

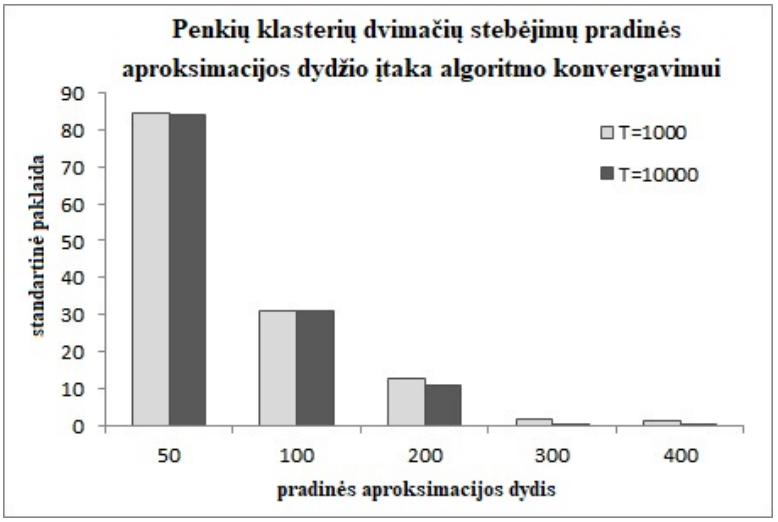


9 pav.: Pradinės aproksimacijos duomenų rinkinio dydžio poveikis algoritmo efektyvumui su dvimačiais duomenimis dviejuose klasteriuose.

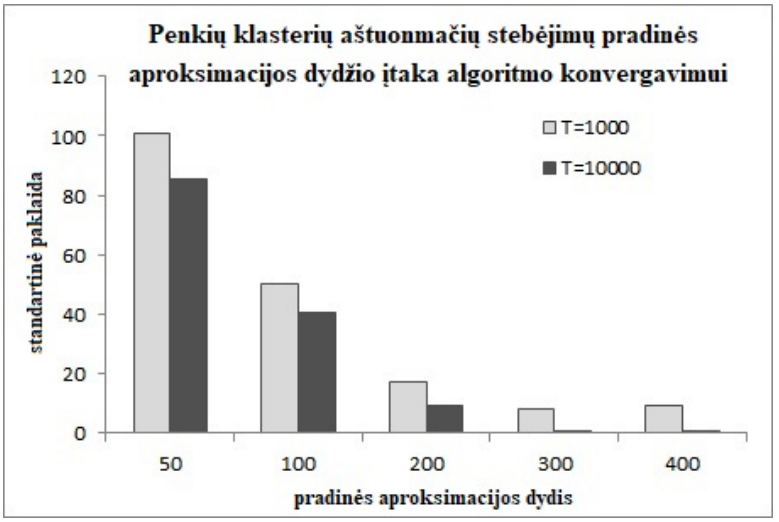


10 pav.: Pradinės aproksimacijos duomenų rinkinio dydžio poveikis algoritmo efektyvumui su aštuonmačiais duomenimis dviejuose klasteriuose.

Pradinio mokymo duomenų rinkinio dydis tampa vis svarbesnis, kai didėja klasterių skaičius (žr. 11 ir 12 pav.). Eksperimentai, atlikti su penkiamačiais ir aštuonmačiais duomenimis, sudarančiais penkis klasterius, rodo, kad skirtumas tarp originalių ir apskaičiuotų parametrų įverčių mažėja, kai didėja pradinis aproksimacijos duomenų rinkinys.



11 pav.: Pradinės aproksimacijos duomenų rinkinio dydžio poveikis algoritmo efektyvumui su dvimačiais duomenimis penkiuose klasteriuose.



12 pav.: Pradinės aproksimacijos duomenų rinkinio dydžio poveikis algoritmo efektyvumui su aštuonmačiais duomenimis penkiuose klasteriuose.

Parametrų įverčiai gali artėti prie lokalių tikėtinumo funkcijos ekstremumų, kurie nėra uždavinio (8) sprendimas, jei pradinis mokymo duomenų rinkinys yra per mažas. Taigi pradinis duomenų rinkinys turi būti tokio dydžio, kad užtikrintų algoritmo stabilumą. Be to, negalime pamiršti bendro stebėjimų rinkinio dydžio, reikalingo efektyviam PMM parametrų vertinimui, reikšmės.

Šis eksperimentas patvirtina P_GPMM_PVA algoritmo privalumus. Vienas iš svarbiausių – skaičiavimo efektyvumas, o kitas – mažas saugojimo atminties poreikis dėl to, kad modelio parametrai atnaujinami palaipsniui.

3.4 Skyriaus apibendrinimas

Remiantis atlikta analize bei gautomis išvadomis pateikiami šie apibendrinimai ir pasiūlymai:

- Šiame darbe sukurtas naujas palaipsnis PMM parametrų vertinimo algoritmas (P_GPMM_PVA), skirtas tolydžiajam PMM su daugiamačiais stebėjimais, pasiskirsčiusiais pagal Gauso dėsnį. Nežinomi PMM parametrai apskaičiuojami rekursinėmis formulėmis pagal didžiausio tikėtinumo metodą. Modelio mokymas atliekamas naudojant fiksuoto dydžio pradinį duomenų rinkinį. Paskui parametrai atnaujinami su kiekvienu nauju stebėjimu, o ankstesnio mokymo duomenų rinkinio saugoti nereikia.

- Kompiuteriniai eksperimentai rodo, kad nuoseklios analizės atveju P_GPMM_PVA algoritmas spartina modelio parametrų įverčių apskaičiavimo greitį. P_GPMM_PVA algoritmas veikia greičiau už rinkinio MVM algoritmą.

- Ištirta palaipsnio algoritmo kriterijaus δ savybė ir jos priklausomybė nuo pradinio mokymo duomenų rinkinio dydžio. Eksperimentų rezultatai parodė, kad pradinio mokymo duomenų rinkinio dydis priklauso nuo modelio būsenų skaičiaus, tačiau parametrų įverčiai artėja prie originalių parametrų reikšmių, jei pakanka pradinės aproksimacijos duomenų.

- Pasiūlyto algoritmo naujumas – rekursinė būsenų perėjimo tikimybės skaičiavimo sistema. Ji keičia klasikinį tiesioginio sklidimo procedūros taikymą – būsenų perėjimo tikimybės apskaičiuojamos pagal Čapmano ir Kolmogorovo lygtį, priešingai nei kiti palaipsniai parametrų vertinimo metodai, kuriuose būsenų perėjimo tikimybės skaičiuojamos naudojant tik klasikinę tiesioginio sklidimo procedūrą. Kompiuterinio modeliavimo būdu atliktų eksperimentų rezultatai parodė, kad siūlomas metodas lemia tikslesnius parametrų įverčius už kitų palaipsnių metodų.

- Nuoseklios analizės atveju P_GPMM_PVA algoritmo sudėtingumas yra tiesinis, priklausomai nuo stebėjimų skaičiaus. Jis reikalauja tik fiksuoto operacijų skaičiaus kiekviename žingsnyje, kai žingsnis atitinka vieną stebėjimą. O klasikinio rinkinio algoritmo PMM parametrams vertinti sudėtingumas yra antrosios eilės.

4 PASLĖPTIEJI MARKOVO MODELIAI SU DIRICHLĖ PASISKIRSTYMAIS

Šiame skyriuje aprašomas pasiūlytas palaipsnis algoritmas, skirtas daugiamačiams Dirichlė PMM parametrams vertinti. Pasiūlytasis palaipsnis algoritmas yra paremtas didžiausio tikėtinumo metodu, MVM algoritmu ir tiesioginio sklidimo procedūra. Jis sudarytas iš dviejų dalių – pradinio modelio apmokymo ir parametrų atnaujinimo. Empiriniu būdu ištirtas pasiūlyto algoritmo efektyvumo kriterijus ir stebėjimų atpažinimo efektyvumas, naudojant kelis klasifikavimo duomenų rinkinius.

Kai kurios šio skyriaus dalys yra publikuotos [30].

4.1 Dirichlė skirstinys

Tegul $\mathbf{O} = (o_1,o_2,\ldots,o_m)$ yra atsitiktinis vektorius, pasiskirstęs pagal Dirichlė skirstinį $Dir(\alpha)$, kur $\alpha = (\alpha_1,\alpha_2,\ldots,\alpha_M)$. Daugiamačio atsitiktinio dydžio pasiskirstymo dėsnis pateikiamas toliau [84]:

$$p(o_1,o_2,\ldots,o_m) = \frac{\Gamma(\alpha_0)}{\prod_{i=1}^{M}\Gamma(\alpha_i)}\prod_{i=1}^{M} o_i^{\alpha_i-1},$$

čia $\alpha_0 = \sum_{i=1}^{m}\alpha_i$, $\alpha_i > 0, \forall i = 1\ldots m$. $\sum_{i=1}^{m-1} o_i < 1$ ir $o_m = 1 - \sum_{i=1}^{m-1} o_i$, kur $0 < o_i < 1, \forall i = 1\ldots m$.

Dirichlė skirstinio vidurkis ir dispersija pateikti toliau [84]:

$$E(o_i) = \frac{\alpha_i}{\alpha_0},$$

$$Var(o_i) = \frac{\alpha_i(\alpha_0-\alpha_i)}{\alpha_0^2(\alpha_0+1)}.$$

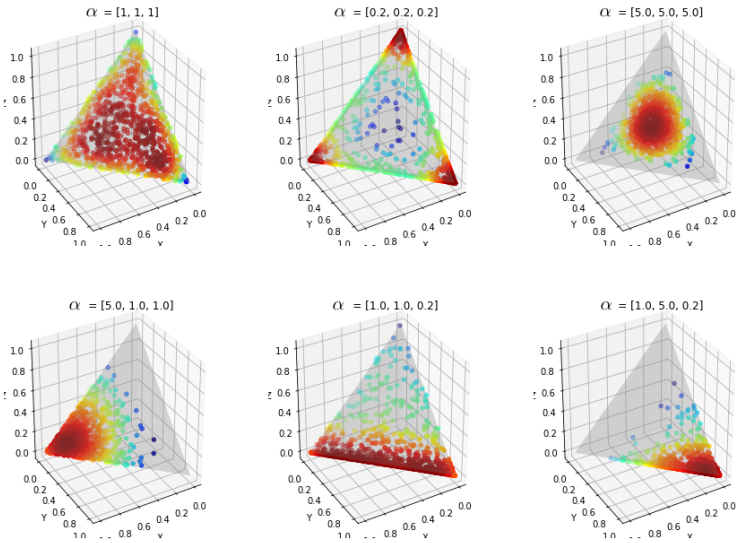
Be to, kovariacija tarp o_i ir o_j yra

$$Cov(o_i,o_j) = \frac{-\alpha_i\alpha_j}{\alpha_0^2(\alpha_0+1)}.$$

Dirichlė skirstinys su parametrų vektoriumi $\alpha = (\alpha_1,\ldots,\alpha_M)$ gali būti aprašytas kaip skirstinys, esantis simplekso viduje $D_m = \left\{(o_1,o_2,\ldots,o_m), \sum_{i=1}^{m-1} o_i < 1\right\}$, apibrėžtas $\mathbb{R}_+^m$. Tai reiškia, kad duomenys turi būti šiame simplekse, t. y. intervale nuo 0 iki 1. Būtina įsitikinti,

kad duomenys visada tenkins šį apribojimą. Paprastas sprendimas – pritaikyti duomenų normalizavimą vektoriams $\mathbf{O} = (o_1, o_2, \ldots, o_m)$, kad šie patektų į simpleksą D_m (žr. 13 pav.) [85]. Gali būti naudojamas *Softmax* normalizavimas, užtikrinantis, kad vektoriaus elementų suma yra lygi 1 ir visi vektoriaus elementai yra teigiami:

$$Softmax_i(\mathbf{O}) = \frac{e^{o_i}}{\sum_j e^{o_j}}. \tag{20}$$



13 pav.: Pagal Dirichlė skirstinį pasiskirstę duomenys, kai parametras α įgyja skirtingas reikšmes.

4.2 Modelio aprašymas ir palaipsnis parametrų atnaujinimas

Šiame darbe pristatomas naujas palaipsnis Dirichlė PMM parametrų vertinimo algoritmas (toliau – P_DPMM_PVA). PMM modeliuoti naudojamas tolydaus tankio daugiamatis išvesties skirstinys. Tolydaus tankio PMM sistemų atveju įprasta naudoti Gauso mišinius kaip išvesties tankio skirstinį. Šiame darbe daroma prielaida, kad PMM būsenų išvesties skirstinys yra Dirichlė.

Algoritmo modeliavimui nustatomi PMM parametrai:

- N yra PMM būsenų skaičius;
- būsenų perėjimo tikimybių matrica $\mathbf{A}$;

- pradinių būsenų tikimybių pasiskirstymo vektorius π;
- Dirichlė skirstinys:

$$Dir(a) = \frac{\Gamma(\sum_{i=1}^{M} \alpha_i)}{\prod_{i=1}^{M} \Gamma(\alpha_i)} (1 - \sum_{i=1}^{M-1} o_i)^{\alpha_M - 1} \prod_{i=1}^{M-1} o_i^{\alpha_i - 1}, \tag{21}$$

čia $\alpha_1, \ldots, \alpha_M$, $\alpha_i > 0$, $M > 2$, $o_1, \ldots, o_M$, $o_i \in (0,1)$ ir $\sum_{i=1}^{M} o_i = 1$.

Esminis PMM parametrų mokymosi uždavinys yra modelyje naudojamo skirstinio parametrų įvertinimas. Atsižvelgiant į daugiamačių stebėjimų vektorių **O** rinkinį, Dirichlė skirstinio (21)) parametrai gali būti įvertinami maksimizuojant duomenų logaritminę tikėtinumo funkciją:

$$\begin{aligned} logDir(\mathbf{O}|\alpha) = \ln\Gamma(\sum_{i=1}^{M} \alpha_i) - \sum_{i=1}^{M} \ln\Gamma(\alpha_i) + \\ + \left[(\alpha_M - 1) \ln (1 - \sum_{i=1}^{M-1} o_i) \right] + \sum_{i=1}^{M-1} \left[(\alpha_i - 1) \ln (o_i) \right]. \end{aligned} \tag{22}$$

Skirstinio parametrų įverčiai turi maksimizuoti logaritminę tikėtinumo funkciją:

$$logDir(\alpha) \to \max_{\alpha}. \tag{23}$$

Stebėjimo tikimybės tankis yra apibrėžiamas taip:

$L(\alpha,\pi) = -\ln \left[\sum_{q=1}^{N} \pi_q Dir(\mathbf{O}|\alpha_q) \right]$, čia π_q yra tikimybė būti būsenoje q.

Apskaičiuodami tikslo funkcijos išvestinę pagal α, gauname:

$$\frac{\partial}{\partial \alpha_j} logDir(O|\alpha) = \Psi\Big(\sum_{i=1}^{M} \alpha_i \Big) - \Psi(\alpha_j) + \ln(o_j), 1 \le q \le M.$$

$\Psi(.)$ yra žinoma kaip *digamma* funkcija.

Tuomet

$$\frac{\partial}{\partial \alpha_j} L(\alpha,\pi) = \frac{\pi^{\langle q \rangle} Dir(O|\alpha_q) \frac{\partial}{\partial \alpha_j} logDir(O|\alpha)}{\sum_{q=1}^{N} \pi^{\langle q \rangle} Dir(O|\alpha_q)}, 1 \le q \le N.$$

Aprašome tikimybę, kad sistema laiko momentu t yra būsenoje q, atsižvelgiant į stebėjimų seką o. Modelis yra

$$\frac{\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}, 1 \le q \le N.$$

Tuomet tikėtinas skaičius kartų, kad sistema pereis iš būsenos q yra lygi

$$\sum_{t=1}^{T}\frac{\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}, 1 \le q \le N.$$

Kadangi visas kiekvienos stebėjimo sekos tikėtinumas yra pagrįstas visų galimų būsenų sekų sumavimu, kiekvienas stebėjimas yra priskiriamas kiekvienai būsenai proporcingai tikimybei, kad modelis yra toje būsenoje, kai stebėjimo vektorius buvo stebimas. Taip PMM *TTF* parametrus galima įvertinti per šių svertinių vidurkių sumas:

$$\widehat{\alpha_i} = \Psi\Big(\sum_{m=1}^{M}\alpha_m^{\langle q\rangle}\Big) + \frac{\frac{1}{T}\sum_{t=1}^{T}\frac{\ln(o_t^{\langle s\rangle})\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}}{\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}}, 1 \le i \le M-1, \quad (24)$$

$$\widehat{\alpha_M} = \Psi\Big(\sum_{m=1}^{M}\alpha_m^{\langle q\rangle}\Big) + \frac{\frac{1}{T}\sum_{t=1}^{T}\frac{\ln(1-\sum_{s=1}^{M-1}o_t^{\langle s\rangle})\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}}{\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_t^{\langle q\rangle} logDir(o_t|\alpha^{\langle q\rangle})}{\sum_{j=1}^{N}\pi_t^{\langle j\rangle} logDir(o_t|\alpha^{\langle j\rangle})}}, \quad (25)$$

čia q yra PMM būsena, $1 \le q \le N$, $1 \le s \le M$, ir $\pi_t^{\langle q\rangle}$ yra tikimybė, kad sistema laiko momentu t bus būsenoje q.

Atkreipkite dėmesį, kad (24) ir (25) formulėse yra naudojamas koeficientas $\pi_t^{\langle q\rangle}$, reiškiantis tikimybę būti būsenoje q laiko momentu t. Šią tikimybę galima lengvai apskaičiuoti naudojant Čapmano ir Kolmogorovo lygtį. Šia lygtimi apskaičiuojama perėjimo tikimybė, kad sistema laiko momentu t bus būsenoje j, jei laiko momentu $t-1$ ji buvo būsenoje i: $\pi_t = \mathbf{A}\cdot\pi_{t-1}$.

(24)-(25) formulės skirtos blokiniam duomenų apdorojimo režimui. Jos gali būti panaudotos išvesti rekursines formules PMM parametrams vertinti.

Jei darysime prielaidą, kad parametro įverčiai labai nekinta, kai gaunamas naujas stebėjimas, o parametras $\alpha_t^{\langle q,s\rangle}$ gali būti aproksimuojamas naudojant ankstesnius

parametrų įverčius $\alpha_{t-1}^{\langle q,s\rangle}$, gaunamos tokios stabilios (angl. *well behaved*) ir lengvai panaudojamos rekursinės parametrų atnaujinimo lygtys:

$$\theta_t^{\langle q\rangle} = \pi_t^{\langle q\rangle} logDir\Big(o_t|\alpha^{\langle q\rangle}\Big), \tag{26}$$

$$\omega_t^{\langle q,s\rangle} = \omega_{t-1}^{\langle q,s\rangle} + \frac{1}{t}\left(\frac{\ln(o_t^{\langle s\rangle})\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \omega_{t-1}^{\langle q,s\rangle}\right), 1 \le q \le N, \tag{27}$$

$$\omega_t^{\langle q,M\rangle} = \omega_{t-1}^{\langle q,M\rangle} + \frac{1}{t}\left(\frac{\ln(1-\sum_{s=1}^{M-1} o_t^{\langle s\rangle})\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \omega_{t-1}^{\langle q,M\rangle}\right), \tag{28}$$

$$\gamma_t^{\langle q\rangle} = \gamma_{t-1}^{\langle q\rangle} + \frac{1}{t}\left(\frac{\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \gamma_{t-1}^{\langle q\rangle}\right), \tag{29}$$

$$\alpha_t^{\langle q,s\rangle} = \Psi^{-1}\left[\Psi\Big(\sum_{i=1}^M \alpha_i^{\langle q\rangle}\Big) + \frac{\omega_t^{\langle q,s\rangle}}{\gamma_t^{\langle q\rangle}}\right], 1 \le s \le M, \tag{30}$$

$\Psi^{-1}(\cdot)$ – yra atvirkštinė *digamma* funkcija.

Įrodymas. Nesunku įsitikinti, kad rekursinės (27)–(29) formulės gali būti gaunamos taip:

$$\begin{aligned}
\gamma_t^{\langle q\rangle} &= \gamma_{t-1}^{\langle q\rangle} + \frac{1}{t}\left(\frac{\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \gamma_{t-1}^{\langle q\rangle}\right) = \\
&= \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}} + \frac{1}{t}\left(\frac{\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}}\right) = \\
&= \frac{1}{t}\sum_{i=1}^{t}\frac{\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}},
\end{aligned}$$

$$\begin{aligned}
\omega_t^{\langle q,s\rangle} &= \omega_{t-1}^{\langle q,s\rangle} + \frac{1}{t}\left(\frac{\ln(o_t^{\langle s\rangle})\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \omega_{t-1}^{\langle q,s\rangle}\right) = \\
&= \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\ln(o_i^{\langle s\rangle})\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}} + \frac{1}{t}\left(\frac{\ln(o_t^{\langle s\rangle})\theta_t^{\langle q\rangle}}{\sum_{j=1}^N \theta_t^{\langle j\rangle}} - \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\ln(o_i^{\langle s\rangle})\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}}\right) = \\
&= \frac{1}{t}\sum_{i=1}^{t}\frac{\ln(o_i^{\langle s\rangle})\theta_i^{\langle q\rangle}}{\sum_{j=1}^N \theta_i^{\langle j\rangle}},
\end{aligned}$$

$$\omega_t^{\langle q,M\rangle} = \omega_{t-1}^{\langle q,M\rangle} + \frac{1}{t}\left(\frac{\ln\left(1-\sum_{s=1}^{M-1} o_t^{\langle s\rangle}\right)\theta_t^{\langle q\rangle}}{\sum_{j=1}^{N}\theta_t^{\langle j\rangle}} - \omega_{t-1}^{\langle q,M\rangle}\right) =$$

$$= \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\ln\left(1-\sum_{s=1}^{M-1} o_i^{\langle s\rangle}\right)\theta_i^{\langle q\rangle}}{\sum_{j=1}^{N}\theta_i^{\langle j\rangle}} +$$

$$+ \frac{1}{t}\left(\frac{\ln\left(o_t^{\langle s\rangle}\right)\theta_t^{\langle q\rangle}}{\sum_{j=1}^{N}\theta_t^{\langle j\rangle}} - \frac{1}{t-1}\sum_{i=1}^{t-1}\frac{\ln\left(1-\sum_{s=1}^{M-1} o_i^{\langle s\rangle}\right)\theta_i^{\langle q\rangle}}{\sum_{j=1}^{N}\theta_i^{\langle j\rangle}}\right) =$$

$$= \frac{1}{t}\sum_{i=1}^{t}\frac{\ln\left(1-\sum_{s=1}^{M-1} o_i^{\langle s\rangle}\right)\theta_i^{\langle q\rangle}}{\sum_{j=1}^{N}\theta_i^{\langle j\rangle}}.$$

PMM parametrų įverčių atnaujinimo algoritmas (P_DPMM_PVA) pateikiamas toliau (4 algoritmas). P_DPMM_PVA algoritmas susideda iš dviejų pagrindinių dalių: modelio mokymo ir modelio parametrų pakartotinio įvertinimo.

Modelio mokymas – pirminis parametrų įvertinimas, atliekamas naudojant nedidelį fiksuoto dydžio mokymo stebėjimų rinkinį ir (26)-(30) formules. Šiame etape $\alpha_t^{\langle q,s\rangle}$ reikšmė apskaičiuojama naudojant fiksuoto taško metodą. Pradiniai palaipsnio algoritmo įverčiai yra būtini siekiant užtikrinti stabilumą ir išvengti konvergavimo į išsigimusius (22) tikėtinumo funkcijos lokalius ekstremumus, kurie nėra uždavinio (23) sprendimas.

Gavus pirminius parametrų įverčius, tolesnis identifikavimas ir parametrų vertinimas gali būti atliekamas stebint procesą realiu laiku. $\alpha_t^{\langle q,s\rangle}$ reikšmės apskaičiuojamos naudojant ankstesnio etapo parametrų reikšmes.

P_DPMM_PVA algoritmo sudėtingumas nuoseklios stebėjimų analizės atveju yra tiesinis, priklausomai nuo stebėjimų skaičiaus. Šis algoritmas, apdorodamas naują stebėjimą, reikalauja tik fiksuoto operacijų skaičiaus kiekvienoje iteracijoje.

4 algoritmas Palaipsnis algoritmas Dirichlė PMM parametrų vertinimui (P_DPMM_PVA).

1: **procedure** PRADINĖ APROKSIMACIJA, MODELIO MOKYMAS:(O_t, $1 \leq t \leq T_1$, pradiniai parametrų įverčiai $\alpha_1^{\langle q,s \rangle}$, $1 \leq s \leq M$, $1 \leq q \leq N$)
2: Maksimizuoti $\alpha_t^{\langle q,s \rangle}$.
3: **while** $\epsilon \geq |\alpha_t - \alpha_{t-1}|$ **do**
4: Normalizuoti stebėjimo vektorių O_t
5: Apskaičiuoti stebėjimo „priklausomybę“ būsenai: $argmax(\theta_t^{\langle q \rangle})$
6: Apskaičiuoti $\omega_t^{\langle q,s \rangle}$, $\gamma_t^{\langle q \rangle}$, $\alpha_t^{\langle q,s \rangle}$
7: **end while**
8: **end procedure**
9: **procedure** PAKARTOTINIS PARAMETRŲ VERTINIMAS:(O_t, $T_1 \leq t \leq T_2$, pradiniai parametrų įverčiai $\alpha_1^{\langle q,s \rangle}$, $1 \leq s \leq M$, $1 \leq q \leq N$)
10: Maksimizuoti $\alpha_t^{\langle q,s \rangle}$
11: **while** $t \leq T_2$ **do**
12: Normalizuoti stebėjimo vektorių O_t
13: Priskirti O klasei, naudojant Bajeso klasifikavimo taisyklę
14: Atnaujinti parametrą $\alpha_t^{\langle q,s \rangle}$
15: **end while**
16: **end procedure**

4.3 Eksperimentų rezultatai

Šiame skyriuje analizuojamas siūlomo P_DPMM_PVA algoritmo efektyvumas, naudojant žinomus daugiamačius klasifikavimo duomenų rinkinius ir sintetinius duomenis. Algoritmo parametrų įverčių artėjimas prie tikrųjų parametrų reikšmių tiriamas naudojant kelis daugiamačius duomenų rinkinius, modeliuojant kelių būsenų PMM. Atliktas P_DPMM_PVA ir P_GPMM_PVA algoritmų lyginimas. Taip pat lygintos Dirichlė PMM ir Gauso PMM modeliavimo galimybės, kai stebėjimai pasiskirstę pagal Gauso ir Dirichlė skirstinius.

4.3.1 Eksperimentų su sintetiniais duomenimis rezultatai ir lyginimas su Gauso PMM

Siekiant palyginti P_DPMM_PVA ir P_GPMM_PVA algoritmų modeliavimo galimybes sugeneruoti du duomenų rinkiniai su stebėjimais, kylančiais iš skirtingų skirstinių. Pirmasis sugeneruotas duomenų rinkinys (A_Dir) sudarytas iš 100

duomenų blokų. Kiekviename bloke yra vienas tūkstantis ($T = 1000$) trimačių stebėjimo vektorių, pasiskirsčiusių pagal Dirichlė skirstinį su nustatytais α parametrais. Antrajame duomenų rinkinyje (B_Gaus) taip pat yra 100 duomenų blokų. Kiekviename bloke yra tūkstantis ($T = 1000$) trimačių stebėjimo vektorių, kurie pasiskirstę pagal Gauso skirstinį su nustatytais būsenų pasiskirstymo parametrais – vidurkiu μ ir kovariacija σ (žr. 5 lentelė).

5 lentelė: Dviejų būsenų PMM parametrai trimačiams Gauso ir Dirichlė duomenų rinkiniams generuoti.

Duomenų rinkinys	**PMM**	**Parametrai**			
		N	**A**	π	B (TTF)
B_Gaus	*Gauso*	2	$\begin{bmatrix} 0{,}5 & 0{,}5 \\ 0{,}6 & 0{,}4 \end{bmatrix}$	$\begin{bmatrix} 0{,}5 \\ 0{,}5 \end{bmatrix}$	$\mu^1 = \begin{bmatrix} 0{,}5 \\ 1{,}5 \\ 3{,}1 \end{bmatrix}, \mu^2 = \begin{bmatrix} 5{,}1 \\ 3{,}8 \\ 2{,}7 \end{bmatrix}, \sigma^1 = \sigma^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
A_Dir	*Dirichlė*	2	$\begin{bmatrix} 0{,}5 & 0{,}5 \\ 0{,}6 & 0{,}4 \end{bmatrix}$	$\begin{bmatrix} 0{,}5 \\ 0{,}5 \end{bmatrix}$	$\alpha^1 = \begin{bmatrix} 0{,}5 \\ 1{,}5 \\ 3{,}1 \end{bmatrix}, \alpha^2 = \begin{bmatrix} 5{,}1 \\ 3{,}8 \\ 2{,}7 \end{bmatrix}$

Sugeneruoti duomenų rinkiniai naudojami palaipsniui įvertinti Dirichlė PMM ir Gauso PMM modelių parametrams (P_DPMM_PVA ir P_GPMM_PVA algoritmais) ir atlikti stebėjimų klasifikavimo (atpažinimo) tikslumo vertinimą. Abiem atvejais nustatytas pradinės aproksimacijos duomenų rinkinio dydis – 200 stebėjimo vektorių. Vidutinis atpažinimo tikslumas apskaičiuotas apdorojus visus 100 duomenų blokus. Kiekvienam PMM nustatytos pradinės inicializacijos vertės pateiktos 6 lentelėje.

Dirichlė PMM parametrų vertinimas ir stebėjimo vektorių klasifikavimas P_DPMM_PVA algoritmu atliktas sunormalizavus stebėjimo vektorius ir naudojant *Softmax* (20) funkciją, kad būtų užtikrintas šių patekimas į intervalą $[0{,}1]$. Algoritmo stabdymo kriterijus nustatytas $\epsilon = 0{,}01$ ir fiksuotas visiems eksperimentams.

7 lentelėje pateikti bendri eksperimento rezultatai rodo skirtingas Dirichlė

6 lentelė: Pradinės dviejų būsenų PMM parametrų vertės, skirtos Gauso ir Dirichlė duomenų rinkiniams apdoroti.

Duomenų rinkinys	**PMM**	**Pradinės modelio parametrų vertės**	
		Būsena #1	Būsena #2
A_Dir	*Gauso*	$\mu^1 = \begin{bmatrix} 0{,}17 \\ 0{,}5 \\ 1{,}03 \end{bmatrix}$, $\sigma^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\mu^2 = \begin{bmatrix} 1{,}7 \\ 1{,}27 \\ 0{,}9 \end{bmatrix}$, $\sigma^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
	Dirichlė	$\alpha^1 = \begin{bmatrix} 0{,}17 & 0{,}5 & 1{,}03 \end{bmatrix}^T$	$\alpha^2 = \begin{bmatrix} 1{,}7 & 1{,}27 & 0{,}9 \end{bmatrix}^T$
B_Gaus	*Gauso*	$\mu^1 = \begin{bmatrix} 0{,}33 \\ 1 \\ 2{,}07 \end{bmatrix}$, $\sigma^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\mu^2 = \begin{bmatrix} 3{,}4 \\ 2{,}53 \\ 1{,}8 \end{bmatrix}$, $\sigma^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
	Dirichlė	$\alpha^1 = \begin{bmatrix} 0{,}33 & 1 & 2{,}07 \end{bmatrix}^T$	$\alpha^2 = \begin{bmatrix} 3{,}4 & 2{,}53 & 1{,}8 \end{bmatrix}^T$

PMM ir Gauso PMM modeliavimo galimybes, kai stebėjimai pasiskirstę pagal skirtingus skirstinius. Apdorojant A_Dir duomenų rinkinį su P_DPMM_PVA, atpažinimo tikslumas siekė 84 %, o naudojant P_GPMM_PVA – reikšmingai mažiau – 58 %. Be to, B_Gaus duomenų rinkinio atpažinimo tikslumas naudojant P_GPMM_PVA yra didesnis kaip 99 %, o naudojant P_DPMM_PVA – 95 %. Jei P_DPMM_PVA naudojamas parametrams vertinti, kai stebėjimai pasiskirstę pagal Dirichlė skirstinį, atpažinimo tikslumas bus didesnis už P_GPMM_PVA, apdorojant tuos pačius stebėjimus. Todėl galima daryti išvadą, kad Gauso PMM taikyti visiems praktiniams uždaviniams negalima, net jei Gauso PMM yra plačiai naudojami įvairioms praktinėms sritims modeliuoti.

7 lentelė: Vidutinis 100 eksperimentinių bandymų pakartojimų atpažinimo tikslumas. Trimačiai duomenys apdorojami 2-ių būsenų PMM.

		Modelis	
		Dirichlė PMM	***Gauso PMM***
Duomenų rinkinys	***B_Gaus***	95,29 %	99,53 %
	A_Dir	84,40 %	58,31 %

4.3.2 Eksperimentai algoritmo kriterijui δ tirti

Eksperimentai atlikti su 3-būsenų trimačiais duomenimis

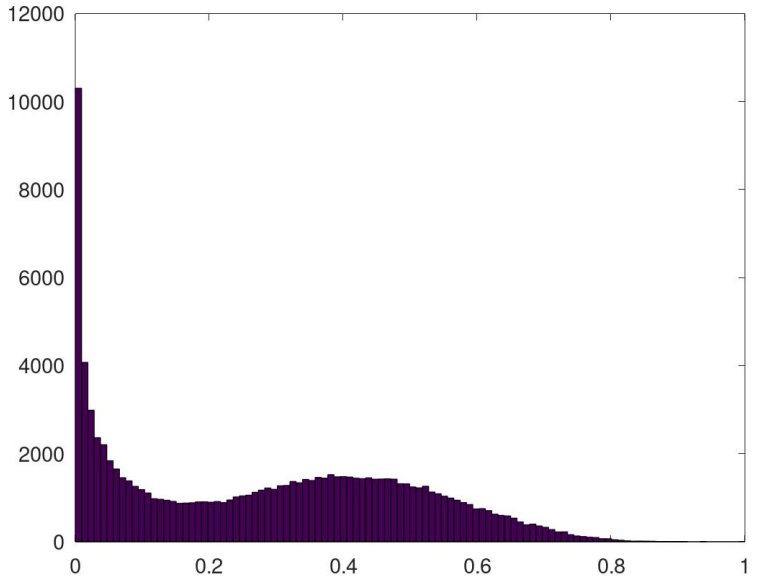
Svarbus parametrų mokymo tikslas – atkurti PMM tikrąsias parametrų vertes. Todėl šiame darbe taip pat tiriama, kaip gerai apmokytų modelių parametrų reikšmės artėja prie tikrųjų parametrų verčių. Apibrėžiamas algoritmo kriterijus δ, kuris nurodo vidutinį atstumą nuo modelio parametrų įverčių iki tikrųjų parametrų verčių. Empiriniu būdu tiriama hipotezė apie realizuoto P_GPMM_PVA algoritmo konvergavimą į uždavinio (23) sprendimą, t. y. algoritmo kriterijaus δ savybę mažėti, kai didėja algoritmo iteracijų (apdorojamų stebėjimų) skaičius.

Norėdami išnagrinėti realizuoto P_DPMM_PVA algoritmo efektyvumą, apskaičiavome stebėjimų atpažinimo tikslumą ir palyginome modelio parametrų įverčius su tikraisiais modelio parametrais.

Trimačiai stebėjimai buvo generuoti iš 3-ių būsenų PMM su šiais parametrais:

$$\mathbf{A} = \begin{bmatrix} 0{,}5 & 0{,}4 & 0{,}1 \\ 0{,}2 & 0{,}6 & 0{,}2 \\ 0{,}2 & 0{,}1 & 0{,}7 \end{bmatrix}, \pi = \begin{bmatrix} 0{,}4 \\ 0{,}3 \\ 0{,}3 \end{bmatrix}, \alpha^1 = \begin{bmatrix} 8 \\ 2 \\ 2 \end{bmatrix}, \alpha^2 = \begin{bmatrix} 80 \\ 80 \\ 80 \end{bmatrix} \text{ ir } \alpha^3 = \begin{bmatrix} 2 \\ 8 \\ 8 \end{bmatrix},$$

Sugeneruotų stebėjimų histograma pavaizduota 14 paveiksle.



14 pav.: Sugeneruotų stebėjimų, pasiskirsčiusių pagal Dirichlė skirstinį, histograma.

8 lentelė: Eksperimentų, atliktų su 3-būsenų PMM ir trimačiais duomenimis, pasiskirsčiusiais pagal Dirichlė skirstinį, rezultatai.

	Parametrų reikšmės			
T=	α^1	α^2	α^3	Atpažinimo tikslumas
Po pradinio mokymo	$\begin{bmatrix}8{,}40\\2{,}03\\2{,}00\end{bmatrix}$	$\begin{bmatrix}67{,}44\\67{,}20\\67{,}75\end{bmatrix}$	$\begin{bmatrix}1{,}93\\7{,}74\\8{,}04\end{bmatrix}$	-
1000	$\begin{bmatrix}8{,}20\\2{,}01\\1{,}98\end{bmatrix}$	$\begin{bmatrix}69{,}05\\68{,}83\\69{,}30\end{bmatrix}$	$\begin{bmatrix}2{,}04\\8{,}19\\8{,}33\end{bmatrix}$	98 % $\pm$0,008
2000	$\begin{bmatrix}8{,}22\\2{,}03\\2{,}01\end{bmatrix}$	$\begin{bmatrix}72{,}17\\71{,}73\\72{,}50\end{bmatrix}$	$\begin{bmatrix}1{,}96\\8{,}05\\8{,}15\end{bmatrix}$	98 % $\pm$0,006
3000	$\begin{bmatrix}8{,}42\\2{,}07\\2{,}06\end{bmatrix}$	$\begin{bmatrix}74{,}42\\74{,}15\\74{,}36\end{bmatrix}$	$\begin{bmatrix}1{,}96\\8{,}00\\8{,}16\end{bmatrix}$	98,1 % $\pm$0,004
4000	$\begin{bmatrix}8{,}48\\2{,}07\\2{,}10\end{bmatrix}$	$\begin{bmatrix}75{,}47\\75{,}42\\75{,}54\end{bmatrix}$	$\begin{bmatrix}1{,}99\\8{,}02\\8{,}14\end{bmatrix}$	98,1 % $\pm$0,004

Eksperimentas atliktas toliau aprašytu būdu. Pradinė aproksimacija (modelio mokymas) atlikta naudojant 1000 stebėjimo vektorių. Po to atliktas pakartotinis parametrų vertinimas ir po kiekvieno 1000 stebėjimų vektorių apskaičiuotas atpažinimo tikslumas. Nustatytas algoritmo užbaigimo kriterijus lygus $\epsilon = 0{,}01$.

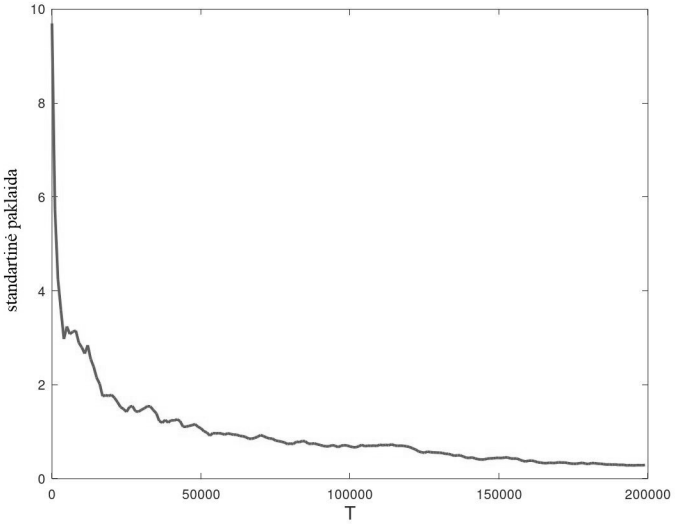
8 lentelėje pavaizduoti rezultatai rodo, kad P_DPMM_PVA algoritmas artėja prie originalių parametrų reikšmių. Didėjant apdorojamų stebėjimų skaičiui P_DPMM_PVA algoritmas išlieka stabilus ir nežymiai gerina atpažinimo tikslumą (apdorojus 1000 stebėjimų vektorių, atpažinimas padidėja 0,1 %). Paskutiniame 8 lentelės stulpelyje taip pat pateiktas atpažinimo tikslumo pasikliautinasis intervalas su 95 % reikšmingumo lygmeniu. Atpažinimo tikslumo (AT) pasikliautinasis intervalas apskaičiuotas pagal formulę:

$$AT \pm 1.96 \cdot \sqrt{\frac{AT \cdot (1 - AT)}{T}} \tag{31}$$

Taip pat apskaičiuota vidutinė standartinė paklaida po kiekvieno stebėjimų bloko apdorojimo. Apskaičiuotų PMM modelio parametrų standartinė paklaida – tai skirtumas tarp tikrųjų parametrų verčių ir modelio parametrų įverčių, gautų

apdorojus nustatytą skaičių stebėjimų.

15 paveiksle pavaizduoti rezultatai rodo, kad realizuotas P_DPMM_PVA algoritmas artėja prie originalių parametrų reikšmių. Po pradinės aproksimacijos modelio parametrų įverčių standartinė paklaida siekė 7, o apdorojus 4000 stebėjimo vektorių sumažėjo iki 2,8. Kitaip tariant, kai didėja T reikšmė, standartinė parametrų įverčių paklaida mažėja. Tai rodo, kad algoritmo kriterijus δ mažėja, kai didėja algoritmo iteracijų skaičius. Mažėjanti standartinė paklaida įrodo, kad P_DPMM_PVA algoritmas artėja prie tikrųjų parametrų verčių ir yra veiksmingas mokantis Dirichlė PMM parametrus.



15 pav.: Modelio parametrų įverčių standartinė paklaida (3-būsenų PMM su trimačiais stebėjimais).

Eksperimentai atlikti su 5-ių būsenų PMM trimačiais duomenimis

Eksperimentai atlikti, norint ištirti algoritmo savybes naudojant 5-būsenų PMM trimačiais stebėjimais. Norėdami išnagrinėti realizuoto P_DPMM_PVA algoritmo kriterijaus δ priklausomybę nuo iteracijų skaičiaus, apskaičiavome stebėjimų atpažinimo tikslumą ir palyginome parametrų įverčius su tikraisiais modelio parametrais. Trimačiai stebėjimai generuoti iš 5-ių būsenų PMM su šiais parametrais:

$$\mathbf{A} = \begin{bmatrix} 0{,}4 & 0{,}2 & 0{,}2 & 0{,}1 & 0{,}1 \\ 0{,}1 & 0{,}4 & 0{,}2 & 0{,}2 & 0{,}1 \\ 0{,}1 & 0{,}3 & 0{,}2 & 0{,}2 & 0{,}2 \\ 0{,}2 & 0{,}1 & 0{,}3 & 0{,}3 & 0{,}1 \\ 0{,}2 & 0{,}2 & 0{,}2 & 0{,}2 & 0{,}2 \end{bmatrix}, \pi = \begin{bmatrix} 0{,}4 \\ 0{,}2 \\ 0{,}2 \\ 0{,}1 \\ 0{,}1 \end{bmatrix}, \alpha^1 = \begin{bmatrix} 8 \\ 2 \\ 2 \end{bmatrix}, \alpha^2 = \begin{bmatrix} 20 \\ 20 \\ 20 \end{bmatrix},$$

$\alpha^3 = \begin{bmatrix} 2 \\ 8 \\ 8 \end{bmatrix}$, $\alpha^4 = \begin{bmatrix} 60 \\ 60 \\ 60 \end{bmatrix}$, ir $\alpha^5 = \begin{bmatrix} 5 \\ 15 \\ 5 \end{bmatrix}$.

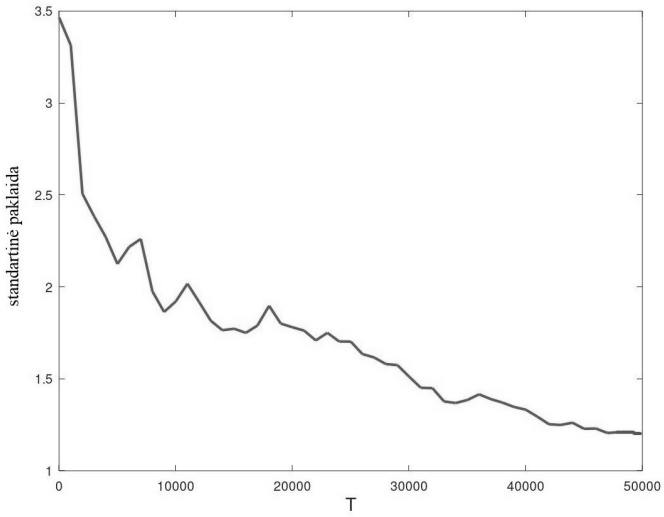
Pradinė aproksimacija P_DPMM_PVA algoritmui atlikta naudojant 1000 stebėjimų. Po pradinės aproksimacijos atliktas pakartotinis parametrų vertinimas ir po kiekvieno tam tikro fiksuoto stebėjimų skaičiaus apskaičiuotas atpažinimo tikslumas su pasikliautinuoju intervalu (31), kurio reikšmingumo lygmuo yra 95 %.

9 lentelėje pateikti rezultatai rodo, kad P_DPMM_PVA algoritmas artėja prie originalių parametrų reikšmių, o atpažinimo tikslumas išlieka stabilus ir nežymiai (iki 0,3 %) didėja po kiekvieno apdoroto stebėjimų bloko. Tai patvirtina, kad algoritmo kriterijus δ mažėja, kai didėja iteracijų skaičius.

9 lentelė: Eksperimentų, atliktų su 5-būsenų PMM ir trimačiais duomenimis, pasiskirsčiusiais pagal Dirichlė skirstinį, rezultatai.

T=	**Parametrų reikšmės**					Atpažinimo tikslumas
	α^1	α^2	α^3	α^4	α^5	
Po pradinio mokymo	$\begin{bmatrix} 8{,}61 \\ 2{,}05 \\ 1{,}92 \end{bmatrix}$	$\begin{bmatrix} 17{,}87 \\ 18{,}18 \\ 18{,}65 \end{bmatrix}$	$\begin{bmatrix} 2{,}28 \\ 9{,}64 \\ 9{,}38 \end{bmatrix}$	$\begin{bmatrix} 57{,}80 \\ 58{,}59 \\ 57{,}90 \end{bmatrix}$	$\begin{bmatrix} 5{,}36 \\ 14{,}80 \\ 4{,}76 \end{bmatrix}$	-
1000	$\begin{bmatrix} 8{,}40 \\ 2{,}04 \\ 1{,}88 \end{bmatrix}$	$\begin{bmatrix} 18{,}10 \\ 18{,}06 \\ 18{,}51 \end{bmatrix}$	$\begin{bmatrix} 2{,}12 \\ 8{,}95 \\ 8{,}72 \end{bmatrix}$	$\begin{bmatrix} 56{,}98 \\ 57{,}34 \\ 57{,}05 \end{bmatrix}$	$\begin{bmatrix} 5{,}30 \\ 15{,}03 \\ 4{,}80 \end{bmatrix}$	77,2 % $\pm 0{,}026$
2000	$\begin{bmatrix} 8{,}44 \\ 2{,}05 \\ 1{,}93 \end{bmatrix}$	$\begin{bmatrix} 18{,}35 \\ 18{,}21 \\ 18{,}62 \end{bmatrix}$	$\begin{bmatrix} 2{,}10 \\ 8{,}98 \\ 8{,}66 \end{bmatrix}$	$\begin{bmatrix} 56{,}85 \\ 57{,}21 \\ 56{,}89 \end{bmatrix}$	$\begin{bmatrix} 5{,}24 \\ 15{,}07 \\ 4{,}84 \end{bmatrix}$	77,5 % $\pm 0{,}018$
5000	$\begin{bmatrix} 8{,}19 \\ 2{,}00 \\ 1{,}95 \end{bmatrix}$	$\begin{bmatrix} 18{,}71 \\ 18{,}54 \\ 18{,}89 \end{bmatrix}$	$\begin{bmatrix} 2{,}01 \\ 8{,}53 \\ 8{,}33 \end{bmatrix}$	$\begin{bmatrix} 56{,}73 \\ 57{,}23 \\ 56{,}57 \end{bmatrix}$	$\begin{bmatrix} 5{,}25 \\ 15{,}19 \\ 4{,}95 \end{bmatrix}$	77,8 % $\pm 0{,}011$
9000	$\begin{bmatrix} 7{,}98 \\ 1{,}97 \\ 1{,}96 \end{bmatrix}$	$\begin{bmatrix} 18{,}46 \\ 18{,}39 \\ 18{,}74 \end{bmatrix}$	$\begin{bmatrix} 2{,}03 \\ 8{,}36 \\ 8{,}23 \end{bmatrix}$	$\begin{bmatrix} 57{,}18 \\ 57{,}78 \\ 57{,}30 \end{bmatrix}$	$\begin{bmatrix} 5{,}13 \\ 15{,}02 \\ 4{,}99 \end{bmatrix}$	77,8 % $\pm 0{,}008$

Iš eksperimentų, atliktų su 5-ių būsenų PMM ir 3-ių būsenų PMM su trimačiais stebėjimais rezultatais, lyginimo aiškėja, kad būsenų skaičius yra svarbus algoritmo efektyvumo veiksnys. Jei būsenų skaičius modelyje didėja, algoritmas lėčiau artėja prie tikrųjų parametrų reikšmių. Taip pat pastebėta, kad algoritmas artėja prie originalių parametrų reikšmių, nes parametrų įverčių standartinės paklaidos mažėja (žr. 16 pav.), o atpažinimo tikslumas apdorojant stebėjimus didėja. Eksperimentai su sintetiniais duomenimis aiškiai rodo P_DPMM_PVA algoritmo efektyvumą.



16 pav.: Modelio parametrų įverčių standartinė paklaida (5-būsenų PMM su trimačiais stebėjimais).

Atlikti eksperimentai patvirtina P_DPMM_PVA algoritmo privalumus. Pagrindiniai iš jų – maži atminties reikalavimai ir skaičiavimo efektyvumas dėl modelio parametrų atnaujinimo palaipsniui.

4.4 Skyriaus apibendrinimas

Remiantis atlikta analize bei gautomis išvadomis pateikiami šie apibendrinimai ir pasiūlymai:

- Šiame darbe pateikiamas algoritmas, skirtas daugiamačių Dirichlė PMM parametrų palaipsniam vertinimui (P_DPMM_PVA). Parametrų įverčiai gaunami apdorojant naujus stebėjimus. Modelio mokymas atliekamas naudojant fiksuoto dydžio pradinį duomenų rinkinį, paskui parametrai atnaujinami su kiekvienu nauju stebėjimu, o pirmesnio mokymo rinkinio saugoti nereikia.

- Kompiuteriniai eksperimentai rodo, kad Dirichlė PMM ir Gauso PMM palaipsnio parametrų vertinimo efektyvumas skiriasi, kai stebėjimai pasiskirstę pagal skirtingus skirstinius. Eksperimentais parodyta, kad kai kuriuose atpažinimo uždaviniuose P_DPMM_PVA algoritmas veikia geriau už P_GPMM_PVA algoritmą.
- Empiriniu būdu ištirta algoritmo kriterijaus δ priklausomybė nuo apdorojamų stebėjimų skaičiaus. Eksperimentų rezultatai rodo, kad kai didėja apdorojamų stebėjimų skaičius, modelio parametrų įverčiai artėja prie tikrųjų parametrų verčių.
- Šiame darbe taip pat ištirtas palaipsnio algoritmo atpažinimo efektyvumas naudojant kelis žinomus klasifikavimo duomenų rinkinius. Kadangi sukurto palaipsnio algoritmo sudėtingumas yra tiesinis, jis gali būti efektyviai taikomas nuoseklaus klasifikavimo ir atpažinimo užduotims, pagrįstoms daugiamačiu Dirichlė PMM modeliu.

5 TAIKYMAI

Šiame skyriuje aprašytas sukurtų palaipsnių PMM parametrų vertinimo algoritmų taikymas keliuose praktiniuose uždaviniuose. Palaipsnis PMM parametrų su Gauso pasiskirstymais vertinimo algoritmas (P_GPMM_PVA) pritaikytas pavienių žodžių atpažinimo uždavinyje. O palaipsnis Dirichlė PMM parametrų vertinimo algoritmas (P_DPMM_PVA) pritaikytas užimtumo nustatymo ir pulsarų nustatymo uždaviniuose. Algoritmų efektyvumas uždavinių sprendime lyginamas tarpusavyje ir su kitais literatūroje aprašytais algoritmais.

Kai kurios šio skyriaus dalys yra publikuotos [30, 86].

5.1 Pavienių žodžių atpažinimas

Automatinis šnekos atpažinimas (AŠA) – sudėtinga daugiapakopė atpažinimo užduotis kompiuterizuotoje šnekos apdorojimo ir atpažinimo sistemoje, kurios tikslas yra klasifikuoti įvesties duomenis į klases pagal tam tikrus požymius. Kitaip tariant, šnekos atpažinimas gali būti apibrėžiamas kaip šnekos transkripcija kompiuteriu [87].AŠA gali būti pritaikytas daugeliui praktinių sričių, pvz., programinės įrangos valdymui [88, 89], numerių rinkimui [90], internetinei paieškai [91, 92] ir kt. Pasiūlyta įvairių atpažinimo metodų, tokių kaip: tiesinės laiko skalės (angl. *linear-time-scaled word-template matching*) [93], paslėptieji Markovo modeliai [32,94,95], gilieji neuroniniai tinklai (angl. *deep neural networks*) [96] ir t. t. Šnekos atpažinimo sistemose plačiai taikomais PMM galima gana tiksliai modeliuoti šnekos signalus.

Tradiciniuose šnekos modeliavimo ir mokymosi metoduose, tokiuose kaip gilieji neuroniniai tinklai, tiesinio laiko skalės kraipymo metodai ir PMM, tiksliam šnekos modelių parametrų mokymuisi reikalingas statinis mokymo duomenų rinkinys. Šių mokymosi metodų sudėtingumas yra bent antros eilės, nes kiekvienoje mokymosi iteracijoje reikalingas skaičiavimų skaičius priklauso nuo duomenų rinkinio dydžio.

Šnekos mokymo ir testavimo medžiagos kokybė ir kiekybė ypač svarbios siekiant teisingai reprezentuoti modeliuojamą šneką ir jos atpažinimo lygį. Plačiai vartojamoms kalboms (pvz., anglų, ispanų, prancūzų, vokiečių, japonų) jau sukurtos labai pažangios atpažinimo sistemos ir didelės šnekų įrašų duomenų bazės. Tačiau mažiau vartojamų kalbų šnekos atpažinimo efektyvumas vis dar negali būti lyginamas su plačiai vartojamų kalbų, kurių šnekos duomenis galima lengvai rinkti ir

naudoti atpažinimo sistemose, efektyvumu. Atvejais, kai šnekos įrašų imtis apmokymui yra per maža, kad būtų praktiškai panaudojama atpažinimo sistemose, gali būti taikomi ne tradiciniai mokymosi metodai, o palaipsnis mokymasis. Palaipsniai mokymosi metodai galėtų padėti rinkti šnekos duomenis realiu laiku.

Neseniai daug dėmesio pradėta skirti palaipsniams modelio parametrų mokymosi metodams [48,52,97]. Pastebėtina, kad palaipsnių mokymosi algoritmų taikymas realaus laiko šnekos atpažinimo sistemose, kurios yra paremtos PMM, nėra labai išsamiai ištirtas. Dauguma realaus laiko šnekos atpažinimo sistemų tariamiems žodžiams iš gautų šnekos signalų atpažinti naudoja statinį apmokytą modelį. Kai mokymui pateikiami nauji šnekos duomenys, šios sistemos negali pritaikyti naujo duomenų rinkinio modeliui, jei šis pirmiau iš naujo neapmokomas su agreguotais duomenimis. Šį trūkumą galima pašalinti, jei mokymas ir modelio parametrų adaptavimas atliekami laipsniškai, apdorojant ir atpažįstant šnekos signalus. Tokie algoritmai leistų sukurti šnekos atpažinimo sistemą, kuri nuolat prisitaikytų prie naujų šnekos signalų ir nesumažintų sistemos atpažinimo tikslumo.

Šiame darbe pavienių žodžių atpažinimas (angl. *isolated word recognition*, santr. PŽA), AŠA poklasis, atliktas naudojant P_GPMM_PVA algoritmą PMM parametrams įvertinti. PŽA sistemoje įvesties duomenys laikomi žodžiais, kiekvienas iš jų apdorojamas atskirai, o pirmiau ištarti žodžiai įtakos atpažinimui neturi. Įvesties duomenys yra neapdorotas šnekos failas, konvertuojamas į akustinių požymių vektorių ir per tam tikrą laiką apdorojamas. Kiekvienam žodžiui modeliuoti naudojamas atskiras PMM su fiksuotu būsenų skaičiumi. Šiame darbe pateikiamas P_GPMM_PVA algoritmo taikymas pavieniams žodžiams atpažinti. Algoritmą sudaro dvi pagrindinės dalys: modelių mokymas ir atpažinimas bei pakartotinis modelio parametrų vertinimas. Mokymo dalyje iš įvesties failų gaunami šnekos akustiniai požymiai ir atliekamas kiekvieno modelio žodžio PMM parametrų įvertinimas. Atpažinimo bei pakartotinio modelio parametrų vertinimo dalyje kiekvieną įvestį bandoma atpažinti, o atpažinto žodžio modelio parametrai atnaujinami. Tai leidžia algoritmui nuolat vertinti modelio parametrus ir sykiu atlikti atpažinimą. Šiame darbe taip pat aptariami sukurto algoritmo eksperimentiniai rezultatai.

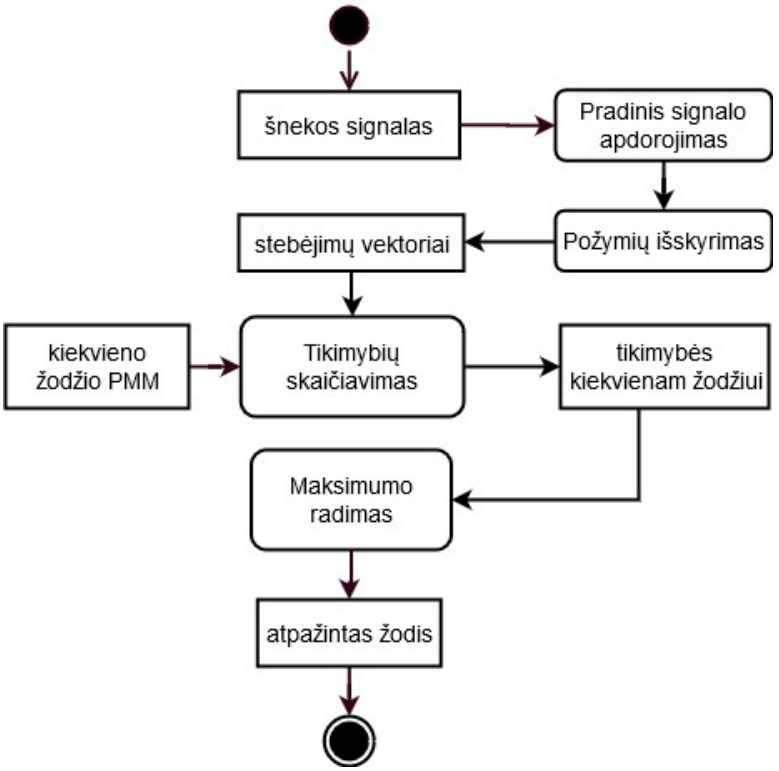
5.1.1 Automatinis šnekos atpažinimas ir PMM

Automatinio šnekos atpažinimo sistemos gauta įvestis – garso signalas – apdorojamas ir išvedamas tekstinis įvesties atitikmuo (žr. 17 pav.). Matematiškai automatinės atpažinimo sistemos uždavinį galima suformuluoti taip: turint garso

signalo požymių seką $\mathbf{O} = o_1,o_2,\ldots,o_n$, rasti žodžių seką $W = w_1,w_2,\ldots,w_m$, kuri turi maksimalią *posterior* tikimybę $P(W|\mathbf{O})$, išreikštą Bajeso formule:

$$W = argmax_w P(W|\mathbf{O}) = argmax_w \frac{P(W)P(\mathbf{O}|W)}{P(\mathbf{O})}, \tag{32}$$

čia $P(\mathbf{O}|W)$ yra tikimybė, kad tariant žodžių seką W, bus stebima požymių seka **O**, $P(W)$ – *a priori* tikimybė, kad bus ištarta žodžių seka W, $P(\mathbf{O})$ – tikimybė, kad bus stebima požymių seka **O**.



17 pav.: AŠA sistemos veikimo principas.

Automatinis šnekos atpažinimas paprastai susideda iš kelių etapų – pirminio apdorojimo, signalo analizės ir galutinio apdorojimo. Pirminis apdorojimas apima signalo kokybės didinimo ir paruošimo požymių išskyrimui operacijas, signalo analizės etapu išskiriami požymiai, o galutinis apdorojimas sudaro šnekos atpažinimo variklį, kuris turi akustinį modelį, žodyną ir gramatiką. Jei sistema taikoma tik pavieniam žodžių atpažinimui, kalbos modelis ir gramatika nėra būtini. PŽA sistemos atpažįsta tariamus pavienius žodžius, atskirtus pauzėmis [95,98]. Šios sistemos

turi „Klausymo / Neklausymo“ būsenas, kai vartotojas turi laukti (paprastai per šias pauzes vykdomas apdorojimas). Šios sistemos parankios, kai vartotojas turi ištarti pavienius žodžius, komandas.

Jei visos šios dalys yra korektiškos, šnekos atpažinimo variklis identifikuoja labiausiai tikėtiną atitikmenį gautai įvesčiai ir grąžina atpažintus žodžius kaip tekstą. Tinkamų požymių išskyrimo ir šnekos atpažinimo metodų parinkimas turi didelės įtakos atpažinimo sistemos tikslumui.

Požymių išskyrimas

Skaitmeninis signalas **Y** į automatinio šnekos atpažinimo sistemą dažniausiai įvedamas iš failo arba mikrofono. Šiame garso signale be ištartų žodžių gali būti daug kitos informacijos, pvz., aplinkos triukšmas, akcentas, intonacija ir t. t. Požymių išskyrimo užduotis yra transformuoti garso signalą **Y** į požymių seką **O**.

Požymių išskyrimas dažniausiai remiasi dažnine signalų analize. Daroma prielaida, kad šnekos signalas yra stacionarus trumpame intervale. Signalas **Y** skaidomas į mažus persidengiančius langus (kadrus). Dažniausiai naudojamas nuo 15 ms iki 30 ms trukmės langas, požymių išskyrimo metu slenkamas po 10–20 ms. Naudojant Furje transformaciją arba tiesinę prognozę, kiekvienam langui randamas dažnių spektras. Langą atitinkančių požymių seka o_i gaunama atliekant netiesines transformacijas (pvz., dažnių skalės iškraipymo transformacija, logaritmavimu). Norint įvertinti signalo pokyčius, naudojamos požymių išvestinės, skaičiuojamos tarp gretimų langų. Dažniausiai į požymių vektorių įtraukiamos pirmosios ir antrosios eilės išvestinės.

Yra įvairių požymių išskyrimo metodų: tiesinio prognozavimo metodas, tiesinės suvokimo prognozės modelio analizė, Mel dažnių kepstro koeficientai ir t. t.

Mel dažnių kepstro koeficientų (angl. *Mel Frequency Cepstral Coefficients*, santr. MDKK) metodo išskirti požymiai dažnai naudojami šnekai atpažinti. Šis metodas remiasi trumpalaike analize (angl. *short-term analysis*), taip iš kiekvieno kadro apskaičiuojamas MDKK vektorius. Pirmiausia, išskiriami garso signalo langai (kadrai), kad sumažėtų garso signalo pertrūkių. Tada atliekama Furje transformacija ir sugeneruojamas Mel filtras (angl. *Mel filter bank*). Galiausiai atliekama atvirkštinė Furjė transformacija ir apskaičiuojami Kepstro koeficientai.

Akustinis modelis

Šiame darbe aptariamas tik akustinis modelis (į sistemą neįtraukiami kalbos modelis ir gramatika), nes pavieniams žodžiams atpažinti taikomas P_GPMM_PVA

algoritmas. Akustinio modelio užduotis – įvertinti žodžių sekos tikimybę $P(\mathbf{O}|W)$ formulei. Teoriškai būtų galima surinkti daug žodžio w garso pavyzdžių ir sudaryti tikimybinio požymių vektorių pasiskirstymo priklausomybę nuo žodžio, bet praktiškai tai sunkiai įgyvendinama, nes didelio žodyno atveju daug žodžių mokymo duomenyse pasitaiko retai arba iš viso net nėra žodį atitinkančio garso įrašo. Todėl dažniausiai požymių vektoriaus **O** skirstinys modeliuojamas pagal mažesnius už žodį fonetinius vienetus – fonemas, kontekstines fonemas arba skiemenis. Šiam skirstiniui modeliuoti naudojami paslėptieji Markovo modeliai.

PMM yra populiarus metodas, nes modeliai gali būti apmokomi automatiškai ir juos galima paprastai panaudoti skaičiavimams. Viso žodžio PMM galima sukurti sujungiant atskirų fonemų (angl. *phoneme*) PMM, apskaičiuojant žodžio sekos tikimybes ir randant viso tinklo paieškos geriausią kelią, atitinkantį optimalią žodžio seką. Šio modelio parametrai yra būsenų perėjimo tikimybės **A** ir vidurkių μ, dispersijų σ svoriai, apibūdinantys būsenos išvesties skirstinius **B**. Kiekvienas žodis ar fonema turės skirtingą išvesties skirstinį. Keleto žodžių ar fonemų sekai skirtas PMM sukuriamas sujungiant individualiai atskiriems žodžiams ir fonemoms apmokytus PMM.

PMM naudojimas šnekai atpažinti remiasi prielaida, kad kalbos signalas yra atsitiktinis procesas, kurio parametrus galima nustatyti.

PMM galima įsivaizduoti kaip atsitiktinį procesą, kuris keliauja per būsenų aibę S ir generuoja požymių vektorius **O**. Tai stochastinis Markovo procesas su nežinomais parametrais, kurie atskleidžiami remiantis stebėjimais [99]. Kitaip tariant, yra du stochastiniai procesai. Pirmasis yra Markovo grandinė, charakterizuojama paslėptomis būsenomis S ir būsenų perėjimo tikimybėmis **A**, o antrasis procesas generuoja stebėjimus priklausomai nuo būsenos priklausomo tikimybinio išvesties skirstinio **B**.

PMM naudojami kiekvienai požymių vektorių sekai klasifikuoti į tam tikrą klasę, kuri pateikiama kaip objektų seka (pvz., raidės, žodžiai ir kt.). Visų galimų klasių sekų tikimybiniai skirstiniai apskaičiuojami ir parenkama geriausia klasių seka. PMM apibrėžia stebimus įvykius (pvz., šnekos signalus kaip įvestį) ir paslėptuosius įvykius (pvz., šnekos atpažinimą ir transkripcijas). Modeliuojamas kiekvieno akustinio vieneto vienas iš kelių būsenų sudarytas PMM. Dažniausiai naudojami trijų būsenų (garso pradžia, vidurys ir pabaiga) PMM. Konkretaus žodžio PMM tinklas gaunamas sujungiant žodžio tarimą atitinkančių akustinių vienetų PMM. Bendruoju atveju nežinoma, kokia būsenų seka sugeneravo požymių vektorių. Egzistuoja iteracinis tiesioginio-atbulinio sklidimo (angl. *forward-backward*) algoritmas,

leidžiantis efektyviai apskaičiuoti šią tikimybę. Tokiame modelyje ieškoma būsenos seka S, generuojanti požymių vektorių $\mathbf{O}$ su didžiausia tikimybe. Geriausiai sekai surasti dažnai naudojamas Viterbi algoritmas.

Tarkime, turime V šnekos pavyzdžių, kuriems atpažinti norime pritaikyti PMM metodą. Pirmasis žingsnis – žodyno sukūrimas. Kiekvienam iš V pavyzdžių sukuriame modelį λ. Modelio tikimybiniai parametrai V, U ir π nustatomi taikant įvertinimo procedūras iš apmokymui pateiktų pavyzdžių. Nagrinėjant nežinomąjį kalbos pavyzdį, atliekame signalo analizę ir gauname stebėjimų seką $\mathbf{O}$. Atpažintuoju pavyzdžiu paskelbiamas etaloninis pavyzdys, kurio modelis geriausiai atitinka nagrinėjamą stebėjimų seką. Modelio atitikimą stebėjimų sekai įvertinant tikėtinumu, kad nagrinėjamoji stebėjimų seka yra sugeneruota modelio, atpažintuoju pavyzdžiu skelbiamas etalonas:

$$Z = arg \max_{1<k<V} P(\mathbf{O}|\lambda_k). \tag{33}$$

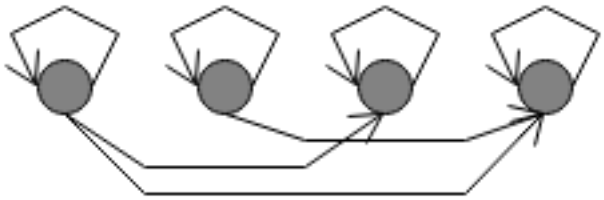
„Iš karės į dešinę“ PMM (žr. 18 pav.) yra dažniausiai naudojami šnekai atpažinti. Šiuo atveju perėjimas iš dabartinės būsenos į kitą galimas tik tada, jei tos būsenos indeksas yra ne mažesnis už dabartinės būsenos indeksą, t. y. $\mathbf{A}_{i,j} = 0, j < i$. Taip pat dažnai šie PMM turi papildomą apribojimą būsenų perėjimo koeficientams, neleidžiantį didelių pokyčių būsenų indeksuose $\mathbf{A}_{i,j} = 0, j > i + \Delta$.

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & a_{3,2} & a_{3,3} & a_{3,4} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & a_{4,3} & a_{4,4} & a_{4,5} & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & a_{5,4} & a_{5,5} & a_{5,6} & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{6,5} & a_{6,6} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & a_{N-1,N-1} & a_{N-1,N} \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix},$$

čia N yra būsenų skaičius. Čia kiekvienos matricos eilutės elementų suma yra lygi vienetui. $\Delta = 2$, t. y. neleidžiama ”peršokti”daugiau kaip per 2 būsenas.

Taigi norint sėkmingai naudoti PMM kalbos signalams atpažinti reikia išspręsti tris uždavinius:

- Įvertinimo uždavinį: turint stebėjimų seką $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ir grandinę aprašančio modelio parametrus $\lambda = (V,U,\pi)$, apskaičiuoti tikimybę $P(\mathbf{O}|\lambda)$,



18 pav.: „Iš karės į dešinę“ PMM.

kad nagrinėjamoji stebėjimų seka sugeneruota nagrinėjamo modelio;

- Paslėptųjų būsenų nustatymo uždavinį: turint stebėjimų seką $\mathbf{O} = (o_1, o_2, \ldots, o_T)$, nustatyti būsenų seką, kuri būtų optimali tam tikro pasirinkto reikšmingo kriterijaus prasme;
- Apmokymo uždavinį: kaip parinkti modelio parametrus, kad būtų maksimizuota tikimybė $P(\mathbf{O}|\lambda)$.

Šiame darbe siūlome P_GPMM_PVA algoritmą pavieniams žodžiams atpažinti. Tokiu būdu gaunami duomenys gali būti apdorojami palaipsniui, o PMM parametrai gali būti atnaujinami nuosekliai, kai tik atsiranda naujų duomenų.

Šnekos atpažinimo sistemos tikslumas

Šnekos atpažinimo sistemos paprastai apibūdinamos pagal atpažinimo tikslumą ir vykdymo greitį. Tikslumą vertinti galima pagal tai, kokią žodžių dalį sistema atpažįsta teisingai (angl. *Word Recognition Rate*, santr. ŽAT). Dažniausiai matavimas atliekamas su testine imtimi. Sistemos atpažinta žodžių seka lyginama su tikrąja žodžių seka. Tuomet tikslumą galima apskaičiuoti pagal formulę:

$$AT = \frac{N - E - D - I}{N} = \frac{H - I}{N}, \tag{34}$$

čia

- E – klaidingai atpažintų žodžių kiekis,
- D – praleistų žodžių kiekis,
- I – įterptų žodžių kiekis,

- H – yra $N - (E + D)$ – teisingai atpažintų žodžių kiekis,
- N – žodžių kiekis testinėje imtyje (N=S+D+H).

5.1.2 Palaipsnio algoritmo (P_GPMM_PVA) taikymas pavieniams žodžiams atpažinti

Norėdami pritaikyti P_GPMM_PVA algoritmą PŽA, turime apibrėžti duomenų apdorojimo procedūrą. Pavienius žodžius galima apdoroti dviem būdais – pasimboliui / pafonemiui arba pažodžiui. Šiame darbe duomenys yra apdorojami pažodžiui.

Pirmoji P_GPMM_PVA algoritmo dalis atlieka pradinį PMM parametrų aproksimavimą, o antroji – atpažinimo procedūrą, skirtą identifikuoti stebėjimus. Identifikavimas atliekamas naudojant Viterbi dekodavimo metodą, populiarų signalų apdorojimo uždaviniuose metodą, nes pasiekiamas žemas klaidų lygis. Optimalus Viterbi dekodavimo metodas naudoja didžiausio tikėtinumo dekodavimo (angl. *maximum likelihood decoding*) algoritmą, kuris sudaro tinklą, skirtą apskaičiuoti stebėjimų seką geriausiai atitinkančią paslėptųjų būsenų seką [87] Turėdamas stebėjimų seką ir PMM, šis algoritmas grąžina būsenų seką, turinčią didžiausią tikėtinumą priklausomai nuo apdorotos stebėjimų sekos. Tuomet atitinkamo PMM parametrai atnaujinami pagal identifikuotą žodį. Pritaikyto P_GPMM_PVA algoritmo schema pateikta 19 paveiksle.

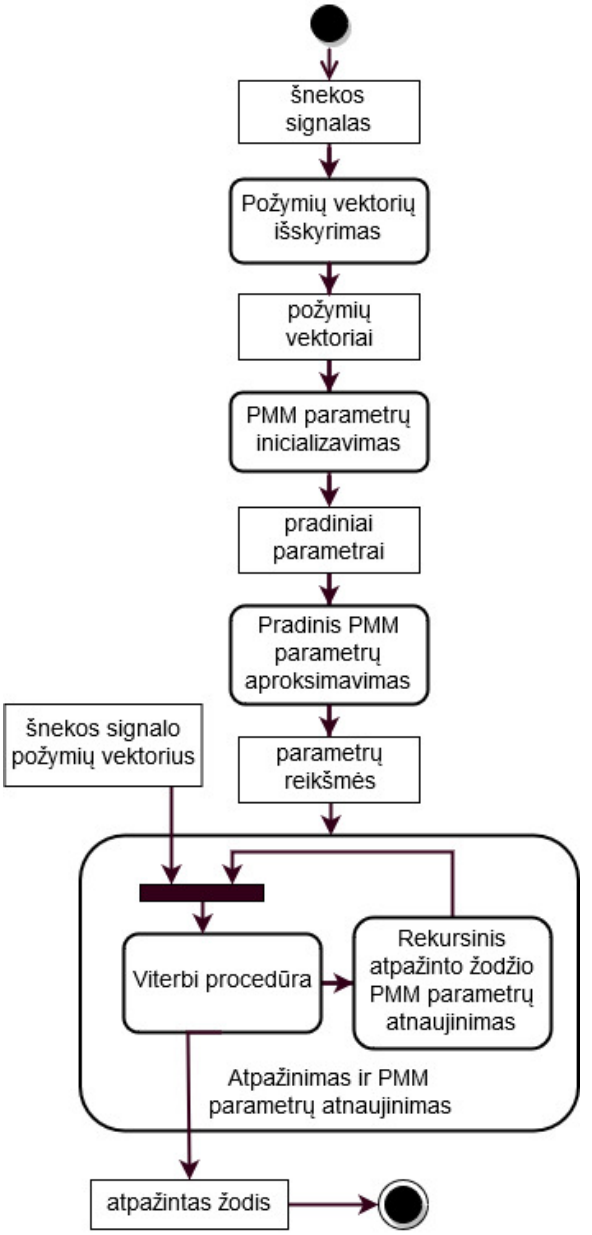
5.1.3 Pavienių žodžių atpažinimo rezultatai

Tyrimo priemonės

Pavienių žodžių atpažinimo eksperimentas modeliuotas MATLAB aplinkoje. Tam sukurtas pavienių žodžių atpažinimo sistemos prototipas, leidžiantis vykdyti eksperimentus ir vertinti jų rezultatus.

Tyrimo sąlygos

Visiems pavienių žodžių atpažinimo eksperimentams atlikti naudojamos toliau aprašytos būsenų perėjimo tikimybių matricos ir pradinio būsenų pasiskirstymo vektoriaus reikšmės.



19 pav.: Realizuoto algoritmo koncepcinis modelis pavieniams žodžiams atpažinti.

Būsenų perėjimo tikimybių matrica nustatyta kaip:

$$\mathbf{A} = \begin{bmatrix} 0 & 0{,}8 & 0{,}2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0{,}6 & 0{,}3 & 0{,}1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0{,}67 & 0{,}33 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Pradinis būsenų pasiskirstymo vektorius nustatytas kaip:

$$\pi = \begin{bmatrix} 0 & 0{,}8 & 0{,}2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T.$$

Eksperimentų su *TIDIGITS* garsynu rezultatai

P_GPMM_PVA algoritmas pritaikytas pavieniams šnekos duomenims atpažinti ir modelio parametrams įvertinti. Modelių mokymas ir testavimas atlikti su *TIDIGITS* garsyno poaibiu [100]. *TIDIGITS* garsynas naudojamas mokyti nuo šnekėtojo nepriklausomų šnekos signalų – sujungtų skaitmenų sekų – atpažinimo algoritmus. Eksperimentams naudotą šio garsyno poaibį sudaro 208 šnekėtojai (94 vyrai, 114 moterys), iš kurių kiekvienas įrašė 22 skaitmenų sekas – nuo nulio iki devynių (”zero”, ”one”, ”two”, ... , ”nine”ir ”oh”). Kiekviena šnekėtojų grupė suskirstyta į testavimo ir mokymo aibes. Požymių vektorių sudaro 39 MDKK formato požymiai. Kiekvienas žodis (ištartas skaitmuo) modeliuojamas kaip dešimties būsenų PMM. Kiekviena būsena aprašoma 39-mačiumi vidurkių vektoriumi ir kovariacijų matrica.

Eksperimentai atlikti toliau aprašytu būdu. Pirmiausia skaičiavimams atlikti pasirinkti įvairių dydžių ($100 \leq t \leq 2000$ žodžiai) fiksuoto pradinio mokymosi duomenų rinkiniai. Antra, tolesnis mokymas ir žodžių atpažinimas atliekami su 1500 žodžių duomenų rinkiniu. Antrojoje algoritmo vykdymo dalyje taip pat apskaičiuojamas žodžių atpažinimo tikslumas. Algoritmo stabdymo kriterijus nustatytas kaip $\epsilon = 0{,}01$. Pagrindinis eksperimento tikslas – ištirti pradinio mokymo (aproksimacijos) duomenų rinkinio dydžio įtaką žodžių atpažinimo tikslumui. Eksperimento rezultatai pateikti 10 lentelėje. Pirmajame 10 lentelės stulpelyje pateikiamas mokymo imties dydis žodžiais, antrajame – atpažinimo tikslumas procentais, o trečiajame – apskaičiuotas atpažinimo tikslumo pasikliautinasis intervalas su 95 % reikšmingumo lygmeniu.

10 lentelė: P_GPMM_PVA algoritmo žodžių atpažinimo tikslumas.

Mokymo imties dydis (žodžiais)	**Atpažinimo tikslumas (%)**	**Pasikliautinasis intervalas**
100	92,53	$\pm$0,051
500	94,33	$\pm$0,020
1000	95,87	$\pm$0,012
1500	97,60	$\pm$0,007
2000	97,27	$\pm$0,007

Žodžių atpažinimo tikslumas siekė 92,53 %, kai pradinio duomenų rinkinio dydis – 100 žodžių. Padidinus pradinio duomenų rinkinio dydį iki 2000 žodžių, atpažinimo tikslumas pakilo iki 97 %. Pradiniui mokymo duomenų rinkiniui didėjant nuo 100 iki 1500, žodžių atpažinimo tikslumas taip pat nuosekliai didėja. Taigi svarbu parinkti tinkamą pradinio modelio apmokymo duomenų rinkinio dydį, užtikrinantį algoritmo stabilumą.

Eksperimentų su „Spoken Arabic Digits“ garsynu rezultatai

Papildomi eksperimentai atlikti naudojant *„Spoken Arabic Digits“* garsyną [101], kurį sudaro du poaibiai: mokymo ir testavimo. Mokymo duomenų poaibis susideda iš 8143 stebėjimų, kurie naudojami pradiniam PMM parametrų mokymuisi, o testavimo duomenų poaibis – iš 2665 stebėjimų, kurie naudojami pakartotiniam modelio parametrų vertinimui ir nuosekliam atpažinimui.

Pavieniams žodžiams modeliuoti naudotas daugiamatis Gauso PMM su daugiamačiais parametrais. Duomenų rinkinį sudaro 12-kos požymių vektorius. Taigi PMM būsenos aprašomos 12-mačiais vidurkių vektoriais ir kovariacijų matricomis. Kiekvienas žodis (skaitmenys) modeliuojamas kaip 10-ties būsenų PMM.

Žodžių atpažinimo tikslumas apskaičiuojamas pakartotinio modelio parametrų vertinimo ir žodžių atpažinimo metu. Algoritmo stabdymo kriterijus nustatytas kaip $\epsilon = 0{,}01$.

Rezultatai parodė 91,86 % pavienių žodžių atpažinimo tikslumą, kurio pasikliautinasis intervalas su 95 % reikšmingumo lygmeniu yra $\pm 0{,}011$. Iš 2200 žodžių P_GPMM_PVA algoritmas teisingai suklasifikavo 2021 žodžius.

5.2 Užimtumo nustatymas

Pastaraisiais dešimtmečiais plačiai paplito komercinių ir gyvenamųjų patalpų užimtumo nustatymo (angl. *Occupancy detection*) uždavinys. Jis yra svarbus dėl to, kad nustačius patalpos komerciniame ar gyvenamajame pastate užimtumą, galima taikyti automatizavimo programas, kurios padėtų kontroliuoti patalpos energijos suvartojimą. Turima informacija apie užimtumą komercinio ar gyvenamojo pastato patalpose taip pat gali būti naudojama siekiant kontroliuoti pastatų energijos, temperatūros (termostatų), ŠVOK sistemų (šildymas – vėdinimas – oro kondicionavimas), apšvietimo ir kitų prietaisų valdymą. Taip padidėtų energijos taupymas arba gyventojų komfortas. Šiuo metu egzistuoja automatinių sistemų, integruojamų su užimtumo nustatymu, kuriomis siekiama efektyviau valdyti patalpos temperatūrą, oro kondicionavimą ir apšvietimą. Užimtumo nustatymas taip pat gali būti naudojamas populiarėjančiose automatinėse namų valdymo sistemose. Be to, užimtumo nustatymas ir vertinimas verslo aplinkoje gali praversti siekiant gauti naudingus statistinius duomenis ir patalpų naudojimo analizei.

Šiame darbe nagrinėjamas patalpų užimtumo nustatymo uždavinys. Uždavinys apima dviejų klasių klasifikavimo – patalpa užimta, patalpa neužimta – uždavinį. Mašininio mokymo algoritmai apmokomi duomenų rinkiniu, kurį sudaro duomenys, gauti iš išmaniųjų skaitiklių (temperatūra, santykinis drėgnumas, apšvietimas, anglies dioksido koncentracija, vandens garų kiekis ore).

Vienas iš eksperimento etapų – išanalizuoti, kokie požymiai duomenų rinkinyje yra naudingiausi ir kuriais galima apmokyti algoritmą, kad būtų gaunama reikšmingiausia informacija apie patalpų užimtumą.

Užimtumo nustatymo uždaviniui spręsti taikomi du algoritmai – P_GPMM_PVA ir P_DPMM_PVA. Gauti rezultatai palyginti su kitais mašininio mokymosi metodais (atraminių vektorių klasifikatoriumi, tiesinės regresijos klasifikatoriumi ir dirbtiniais neuroniniais tinklais), aprašytais [4] straipsnyje.

5.2.1 Susiję darbai

Užimtumo nustatymo uždavinį galima suskirstyti į dvi pagrindines grupes pagal užimtumo stebėjimo principus: bendra ir individuali stebėsena. Pirmajame nurodomas bendro užimtumo erdvėje įvertinimas, o antrajame – individualaus asmens identifikavimas ir padėties sekimas erdvėje. Individualios stebėsenos užimtumo vertinimo sistemos apdoroja duomenis, gautus iš galutinių sistemos naudotojų turimų išmaniųjų įrenginių (pvz., mobiliųjų telefonų arba radijo dažnių atpažinimo

žymių (RDAŽ). Šiuo atveju daroma prielaida, kad galutiniai sistemos naudotojai visuomet su savimi turi išmaniuosius įrenginius. Kita vertus, bendros stebėsenos užimtumo nustatymo sistemos, naudojančios įvairius jutiklius (tokius kaip akustiniai jutikliai, pasyvi infraraudonoji spalva (PIS) ir t. t.), gali pateikti informaciją apie užimtumą, nereikalaujant naudotojų su savimi turėti įrenginių.

Infrastruktūros požiūriu atviros sistemos naudoja sensorius, tokius kaip: PIS judesio jutikliai, durų skaitikliai, akustiniai jutikliai ir gylio kameros, kad būtų galima įvertinti patalpos užimtumą, o uždaros sistemos gauna užimtumo informaciją, naudodamos netiesioginius sekimo metodus, pvz., naudojamo įrenginio būseną, elektros energijos suvartojimą ir pan.

Literatūroje nagrinėjami taisyklėmis pagrįsti (angl. *rule-based methods*) ir mašininio mokymosi metodai įvertinti patalpos užimtumui, sujungiant iš įvairių šaltinių gaunamą informaciją. Taikant mašininio mokymosi metodus, patalpos užimtumo įvertinimas yra laikomas klasifikavimo uždaviniu ir sprendžiamas naudojant pasirinktą klasifikatorių.

Literatūroje naudojami įvairūs mašininio mokymosi algoritmai, pvz., atraminių vektorių klasifikatorius (angl. *Support Vector Machines*, santr. AVK), dirbtiniai neuroniniai tinklai (angl. *Artificial Neural Networks*, santr. DNT), sprendimų medžiai (angl. *Decision Trees*, SM), agentais pagrįsti modeliai ir kt. [102] straipsnyje naudojami mokymosi metodai – sąlyginis atsitiktinio lauko modelis (angl. *Conditional Random Field Model*) ir paslėptosios Markovo atraminių vektorių mašinos (angl. *Hidden Markov Support Vector Machine*), siekiant įvertinti naudotojų skaičių trijų asmenų gyvenamojoje patalpoje iš signalizacijos sistemos PIR judesio jutiklių rodmenų.

Užimtumo įvertinimas atviro (angl. *open-plan*) biuro erdvėje tiriamas [103] straipsnyje. Aprašoma, kad DNT modelį apmokius garso, temperatūros, CO2, PIR judesio jutiklių duomenimis ir užimtumą vertinant iki 6 naudotojų, atpažinimo tikslumas siekia 75 %. [104] straipsnyje vertinant užimtumą gyvenamojoje patalpoje, naudojama informacija iš kelių jutiklių, tokių kaip: galios matuokliai, CO2, temperatūros jutikliai. Šiame straipsnyje aprašytas Dempster-Shafer teoriją su paslėptaisiais Markovo modeliais derinantis metodas, atsižvelgdamas į energijos suvartojimą, apskaičiuoja patalpos užimtumo įverčius.

Alternatyviai užimtumo nustatymą galima vertinti iš netiesioginių stebėjimo šaltinių, tokių kaip energijos skaitiklių duomenys, mobiliųjų įrenginių signalo stiprumo duomenys ir kompiuterio veikimas. Literatūroje vis dažniau nagrinėjami netiesioginiai užimtumo stebėjimo būdai, daugiausia dėmesio skiriama duomenų

rinkimui iš išmaniųjų energijos skaitiklių. [105] straipsnyje iš elektros skaitiklių duomenų, kartu ir PIR judesio daviklių, siekiama nustatyti biuro darbuotojų veiklą pagal darbo stalus ir suskaičiuoti naudotojų skaičių kambaryje. Darbo kompiuterių atveju bendras atpažinimo tikslumas siekė 95 %, o geriausiu atveju naudotojų skaičiaus atpažinimo tikslumas siekė 87 %. [106] straipsnio autorius tiria netiesioginio užimtumo stebėjimo efektyvumą iš pažangiųjų energijos skaitiklių elektros sunaudojimo duomenų, šiuos apdorodamas statistinės analizės metodais.

Užimtumo nustatymas gali būti atliekamas ir iš kitų jutiklių tipų, pavyzdžiui, vandens skaitiklių. Gyventojų sunaudotas vandens kiekis arba elektros prietaisai, tokie kaip indaplovės, gali parodyti, ar pastatas yra užimtas, ar ne. [107] straipsnyje pateikiama vandens sunaudojimo sistemų ir susijusių klasifikavimo metodų apžvalga.

[108] straipsnio autoriai pasiūlė užimtumo nustatymo sistemą, pagrįstą energijos suvartojimo duomenimis, siekiant išspręsti problemas, kylančias mokantis iš riboto skaičiaus duomenų arba jų visiškai neturint.

Pritaikius palaipsnius PMM parametrų vertinimo algoritmus galima būtų išspręsti dėl apmokymo duomenų aibės dydžio kylančią problemą. Palaipsnio algoritmo atveju nereikėtų saugoti jau surinktų duomenų iš sensorių. Palaipsnis algoritmas galėtų realiu laiku gaunamus iš įvairių jutiklių duomenis apdoroti ir nustatyti patalpos užimtumą.

5.2.2 Eksperimentinis tyrimas

Tyrimo tikslas – pritaikyti pasiūlytus palaipsnius algoritmus (P_GPMM_PVA ir P_DPMM_PVA) užimtumo nustatymo uždavinyje ir ištirti jų efektyvumą.

Eksperimentinio tyrimo metu nagrinėjamas požymių aibės tinkamumas užimtumui nustatyti. Algoritmų efektyvumui įvertinti pateiksime keletą klasifikavimo rodiklių. Įvertintų palaipsnių algoritmų efektyvumą lyginsime su kitais literatūroje publikuotais rezultatais ir bandysime įvertinti tarpusavio pranašumus.

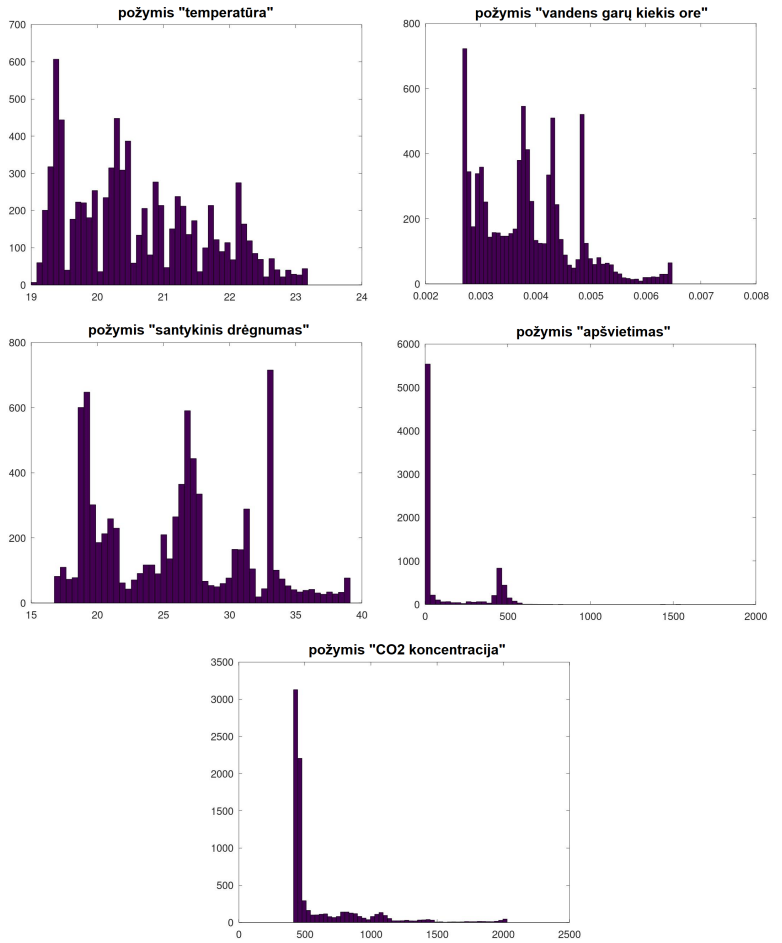
Tyrimo duomenys

Eksperimentai atlikti su užimtumo nustatymo duomenų rinkiniu [109]. Užimtumo nustatymo duomenų rinkinys naudojamas dviejų klasių klasifikavimui atlikti. Duomenys sudaro kambario užimtumą apibūdinančias dvi klases: užimtas ir neužimtas.

Duomenų rinkinio stebėjimų vektoriai aprašomi tokiais požymiais:

1. data (metai-mėnuo-diena);
2. laikas (valanda:minutės:sekundės);
3. temperatūra (Celsijus);
4. santykinis drėgnumas (%);
5. apšvietimas (liumenai);
6. anglies dioksido (CO2) koncentracija (ppm);
7. vandens garų kiekis ore; temperatūros ir santykinio drėgnumo santykis (vandens gramų / oro kilograme).

Duomenų rinkinio požymių histogramos pavaizduotos 20 paveiksle.



20 pav.: Užimtumo nustatymo duomenų rinkinio požymių histograma.

Histogramose pavaizduoti atskiri požymiai: temperatūra, santykinis drėgnumas, apšvietimas, anglies dioksido koncentracija ir vandens garų kiekis ore. Matyti, kad daugelis požymių yra daugiamodaliniai. O požymių „temperatūra“, „apšvietimas“ ir „anglies dioksido koncentracija“ histogramos krypsta į dešinę.

Tyrimo sąlygos

Eksperimentams visas duomenų rinkinys padalytas į dvi dalis: mokymo ir testavimo. Mokymo duomenų rinkinį sudaro 8143 stebėjimai, o testavimo duomenų rinkinį sudaro 2665 stebėjimai, kurie naudojami pakartotiniam modelio parametrų vertinimui ir nuosekliam atpažinimui.

Duomenų rinkinio apdorojimas: P_DPMM_PVA algoritmo atveju stebėjimų normalizavimas atliktas pagal *Softmax* (20) formulę, kad visi stebėjimo vektorių elementai būtų teigiami, o jų suma lygi vienetui. Tačiau P_GPMM_PVA algoritmo atveju *Softmax* normalizavimas neatliktas. Pradinės modelio parametrų vertės priskirtos apdorojant mokymo duomenis momentų metodu (angl. *method of moments*). Nustatytas lygus pasirinktai reikšmei $\epsilon = 0{,}01$ algoritmo stabdymo kriterijus (ϵ). Stebėjimų klasifikavimas atliktas pagal Bajeso klasifikavimo taisyklę. Atpažinimo tikslumas apskaičiuotas palaipsnių algoritmų pakartotinio parametrų vertinimo dalyje. Sykiu taip pat apskaičiuotas tiek bendras, tiek kiekvienos PMM būsenos atpažinimo tikslumas. Paskui pateikiamos algoritmų efektyvumo nustatymo metrikos ir atliktos požymių atrankos analizė.

Nustatyti užimtumo nustatymo duomenų rinkinio PMM parametrai pateikti 11 lentelėje.

11 lentelė: Nustatytos Gauso ir Dirichlė PMM parametrų vertės užimtumo nustatymo uždaviniui spręsti.

	Parametrai		
Duomenų rinkinys	N	**A**	π
Užimtumo nustatymas	2	$\begin{bmatrix} 0{,}99 & 0{,}01 \\ 0{,}01 & 0{,}99 \end{bmatrix}$	$\begin{bmatrix} 0{,}5 \\ 0{,}5 \end{bmatrix}$

Tyrimo priemonės

Užimtumo nustatymo eksperimentas modeliuotas MATLAB aplinkoje. Tam sukurtas užimtumo nustatymo sistemos prototipas, leidžiantis vykdyti eksperimentus ir vertinti jų rezultatus.

Tyrimo eiga ir algoritmo įvertinimo metodika

Algoritmų efektyvumui įvertinti naudojamas standartinis įvertinimo metodas – maišaties matrica (angl. *confusion matrix*) ir pagrindiniai rodikliai, apskaičiuojami iš šios matricos verčių.

Dviejų klasių atveju klasifikuojami objektai gali priklausyti klasei (tokius objektus vadinsime teigiamais) ir nepriklausyti klasei (neigiami objektai). Taip pat kiekvienas objektas klasifikatoriaus yra priskiriamas prie vienos iš klasių. Rezultatas taip pat gali būti arba teigiamas, arba neigiamas (objektas klasifikatoriaus priskirtas prie nurodytos klasės arba ne). Jei objektas teigiamas ir klasifikatoriaus priskirtas prie nurodytos klasės, jis vadinamas tikrai teigiamu (angl. *true positive*, TP). O jeigu neigiamas objektas klasifikatoriaus nepriskirtas prie nurodytos klasės, jis vadinamas tikrai neigiamu (angl. *true negative*, TN). Teigiamas objektas, klasifikatoriaus klaidingai priskirtas prie klasės, vadinamas klaidingai neigiamu (angl. *false negative*, FN), o neigiamas objektas, klasifikatoriaus priskirtas prie tiriamos klasės, vadinamas klaidingai teigiamu (angl. *false positive*, FP).

Apibrėžiamos pagrindinės sąvokos:

- tikrai teigiamas – objektas O_i priskirtas prie klasės S_j ir iš tiesų jai priklauso;
- tikrai neigiamas – objektas O_i nepriskirtas prie klasės S_j ir iš tiesų jai nepriklauso;
- klaidingai teigiamas – objektas O_i priskirtas prie klasės S_j, bet iš tiesų jai nepriklauso;
- klaidingai neigiamas – objektas O_i nepriskirtas prie klasės S_j, bet iš tiesų jai priklauso;

Norint įvertinti pasiūlytus algoritmus dviejų klasių klasifikavimo uždaviniui, naudojami tikslumo, atrinkimo, bendro klasifikavimo teisingumo ir F-rodiklio matai, apskaičiuojami pagal maišaties matricos reikšmes.

Tikslumas (angl. *Precision*) apskaičiuojamas pagal formulę:

$$Precision = \frac{TP}{TP + FP}. \tag{35}$$

Atrinkimas (angl. *Recall*) apskaičiuojamas pagal formulę:

$$Recall = \frac{TP}{TP + FN}. \tag{36}$$

Bendras klasifikavimo teisingumas (angl. *Accuracy*) yra bendras teisingų prognozių skaičius ir apskaičiuojamas pagal formulę:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{37}$$

Vien tikslumas ar atrinkimas negali apibūdinti klasifikatoriaus efektyvumo. Todėl įvedamas šių dviejų rodiklių derinys – F-rodiklis (angl. *F-score / F-measure*).

$$Fscore = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{38}$$

Artimesnė vienetui reikšmė reiškia geresnį klasifikatoriaus tikslumą ir atrinkimą, o mažesnė reikšmė reiškia blogesnį bendrą klasifikavimo teisingumą ar tikslumą.

Kad klasifikavimo vertinimas atitiktų užimtumo nustatymo uždavinį, paskirsime dvi – užimta ir neužimta – klases. Maišaties matrica užimtumui nustatyti pateikta 12 lentelėje.

12 lentelė: Maišaties matrica užimtumui nustatyti.

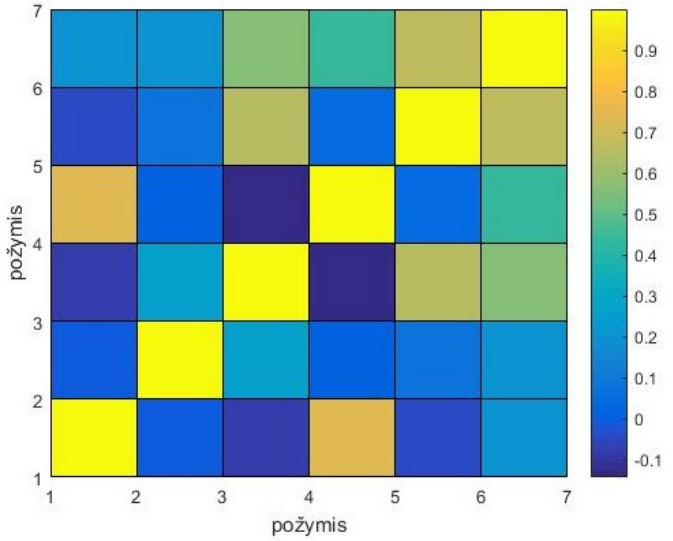
		Priskirta klasė	
		Neužimta	**Užimta**
Tikra klasė	**Neužimta**	TP	FN
	Užimta	FP	TN

Jei užimtumo nustatymą laikysime procesu, kai erdvė laikui bėgant tampa užimta arba neužimta, tokį procesą galėsime modeliuoti dviejų būsenų PMM.

Šiuo atveju užimtumo nustatymo uždaviniui spręsti galima taikyti palaipsnį PMM parametrų vertinimo algoritmą. Darant prielaidą, kad duomenų rinkinys apmokymui gali būti gana mažas, palaipsnis algoritmas išspręstų šią problemą, kadangi laikui bėgant iš sensorių gaunami duomenys būtų naudojami pastoviam modelio parametrų atnaujinimui, taip surenkant vis daugiau reikšmingos informacijos.

Kitas tyrimų tikslas yra patikrinti, ar duomenų rinkinio požymiai yra tarpusavyje susiję.

Eksperimento metu apskaičiuotos koreliacijos koeficientų matricos reikšmės ir P-reikšmės tarp užimtumo nustatymo duomenų aibės požymių (žr. 21 paveikslas). Iš 13 lentelėje pateiktų rezultatų matyti, kad koreliacijos koeficientų reikšmės daugeliu atvejų yra normalios. Labai stipri koreliacija (0,96) yra tarp požymių „santykinis drėgnumas“ ir „vandens garų kiekis ore“, o stipri koreliacija – tarp dviejų požymių porų: „data“ ir „santykinis drėgnumas“ bei „data“ ir „vandens garų kiekis ore“.



21 pav.: Užimtumo nustatymo duomenų požymių koreliacijos koeficientų matrica.

14 lentelėje pateiktos apskaičiuotos koreliacijos P-reikšmės.Matyti, kad požymių („data“ ir „santykinis drėgnumas“, „data“ ir „vandens garų kiekis ore“, „santykinis drėgnumas“ ir „vandens garų kiekis ore“) P-reikšmės yra mažesnės už standartinį reikšmingumo lygį (0,05), tad galime teigti, kad atitinkamos koreliacijos tarp minėtų požymių yra reikšmingos.

Remdamiesi tuo, kad tarp šių požymių yra stipri koreliacija, galime teigti, kad pašalinus atitinkamą požymį iš stipriai koreliuotų požymių poros, atpažinimo tikslumas apmokius algoritmą nepakistų arba geriausiu atveju padidėtų.

Toliau eksperimentai atliekami remiantis šiais rezultatais. Modelio apmokymas ir pakartotinis modelio parametrų vertinimas atliekamas iš požymių vektoriaus

13 lentelė: Užimtumo nustatymo duomenų požymių koreliacijos koeficientų matricos reikšmės.

Požymis	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**
#2	-0,01					
#3	-0,08	0,26				
#4	0,74	0,02	-0,14			
#5	-0,05	0,08	0,65	0,04		
#6	0,20	0,21	0,56	0,44	0,66	
#7	0,70	0,10	0,15	0,96	0,23	0,63

#1 – data; #2 – laikas; #3 – temperatūra; #4 – santykinis drėgnumas;
#5 – apšvietimas; #6 – anglies dioksido koncentracija;
#7 – vandens garų kiekis ore

14 lentelė: Užimtumo nustatymo duomenų aibės požymių koreliacijos P-reikšmės.

Požymis	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**
#2	0,33					
#3	0,00	0,00				
#4	0,00	0,12	0,00			
#5	0,00	0,00	0,00	0,001		
#6	0,00	0,00	0,00	0,00	0,00	
#7	0,00	0,00	0,00	0,00	0,00	0,00

#1 – data; #2 – laikas; #3 – temperatūra; #4 – santykinis drėgnumas;
#5 – apšvietimas; #6 – anglies dioksido koncentracija;
#7 – vandens garų kiekis ore

atitinkamai po vieną pašalinus požymius „data“, „santykinis drėgnumas“ ir „vandens garų kiekis ore“. Apskaičiuotas atpažinimo tikslumas pateiktas 15 lentelėje. 15 lentelės stulpeliuose pateikti atpažinimo rezultatai, gauti apmokius modelį su visa nemodifikuota duomenų aibe ir modifikuota duomenų aibe. Trys modifikuotos duomenų aibės gautos atitinkamai pašalinus požymius „data“ (1-asis požymis), „santykinis drėgnumas“ (4-asis požymis) ir „vandens garų kiekis ore“ (7-asis požymis).

Atpažinimo tikslumas nekinta, todėl galima teigti, kad pašalinus požymį sumažės skaičiavimams atlikti reikalingų resursų naudojimas, o atpažinimas nepasikeis.

Eksperimentais siekiama patikrinti P_DPMM_PVA algoritmo galimybes užimtumo nustatymo uždavinyje, apdorojant daugiamačius duomenis. P_DPMM_PVA algoritmo klasifikavimo tikslumas palygintas su P_GPMM_PVA.

15 lentelė: Užimtumo atpažinimo rezultatai atlikti su P_DPMM_PVA algoritmu.

	Nepašalinus požymių	**Pašalinus 7-ąjį požymį**	**Pašalinus 4-ąjį požymį**	**Pašalinus 1-ąjį požymį**
Teisingumas	0,98	0,97	0,97	0,98
Tikslumas	0,94	0,93	0,93	0,94
Atrinkimas	1	1	1	1
F-rodiklis	0,97	0,96	0,96	0,97

Kiekvienos PMM būsenos atpažinimo tikslumas (%) ir bendras atpažinimo tikslumas pateikti 16 lentelėje. Apdorojus užimtumo nustatymo duomenų rinkinį, P_GPMM_PVA algoritmas neteisingai suklasifikavo 346 stebėjimus iš 2665 stebėjimų. O P_DPMM_PVA neteisingai suklasifikavo tik 61 stebėjimą. Užimtumo nustatymo duomenų palaipsnio klasifikavimo atveju P_DPMM_PVA yra 10 % tikslesnis nei P_GPMM_PVA.

Detalesni algoritmų efektyvumo rodiklių rezultatai pateikti toliau.

16 lentelė: P_DPMM_PVA (Dirichlė PMM) ir P_GPMM_PVA (Gauso PMM) algoritmų atpažinimo tikslumas (%)) apdorojant užimtumo nustatymo duomenų rinkinį.

Duomenų rinkinys	**Gauso PMM**		**Dirichlė PMM**	
	Būsena #1	Būsena #2	Būsena #1	Būsena #2
Užimtumo nustatymas	90,49	80,96	96,39	100
	87,02		97,71	

17 ir 18 lentelėse pateikti užimtumo nustatymui gauti maišaties matricos vertės P_DPMM_PVA ir P_GPMM_PVA algoritmais. P_DPMM_PVA algoritmas 1632 stebėjimų vektorius, kurie priklauso klasei „Neužimta“, priskyrė prie teisingos klasės, o 61 stebėjimų vektorių, priklausantį klasei „Neužimta“, priskyrė prie klasės „Užimta“. Palyginimui – palaipsnis Gauso PMM parametrų vertinimo algoritmas 1532 stebėjimų vektorius, priklausančius klasei „Neužimta“, priskyrė prie teisingos klasės, o tos pačios klasės 161 stebėjimų vektorių priskyrė prie klasės „Užimta“.

Šios maišaties matricų vertės yra toliau naudojamos apskaičiuojant reikalingus algoritmų efektyvumo vertinimo rodiklius. Šios metrikos pateiktos 19 lentelėje.

17 lentelė: Užimtumui nustatyti gautos maišaties matricos vertės, apdorojus duomenis P_DPMM_PVA algoritmu.

		Priskirta klasė	
		Neužimta	**Užimta**
Tikra klasė	**Neužimta**	1632	61
	Užimta	0	972

18 lentelė: Užimtumui nustatyti gautos maišaties matricos vertės, apdorojus duomenis P_GPMM_PVA algoritmu.

		Priskirta klasė	
		Neužimta	**Užimta**
Tikra klasė	**Neužimta**	1532	161
	Užimta	185	787

19 lentelė: Užimtumui nustatyti gauti rodikliai, apdorojus duomenis P_DPMM_PVA ir P_GPMM_PVA algoritmais. P_DPMM_PVA visais atvejais duoda geresnius rodiklius už P_GPMM_PVA.

	Dirichlė	**Gauso**
Teisingumas	0,98	0,87
Tikslumas	0,94	0,83
Atrinkimas	1	0,81
F-rodiklis	0,97	0,82

Iš 19 lentelėje pavaizduotų rezultatų matyti, kad gautos metrikos P_DPMM_PVA algoritmu apdorojus duomenis yra geresnės už P_GPMM_PVA. P_DPMM_PVA algoritmo atveju atrinkimo rodiklis siekia 1, o P_GPMM_PVA algoritmo – 0,8. Taip pat ir tikslumo rodiklis P_DPMM_PVA atveju yra 0,94, o P_GPMM_PVA – 0,83. Matyti, kad P_DPMM_PVA algoritmo atveju F-rodiklis yra 0,97, o P_GPMM_PVA algoritmo F-rodiklis yra mažesnis – 0,82. Tai rodo, kad P_DPMM_PVA algoritmas yra tikslesnis už P_GPMM_PVA algoritmą.

Gautus eksperimentų rezultatus galime palyginti su [4] straipsnyje pateiktais rezultatais. Nors šiame straipsnyje autoriai nagrinėja įvairius mašininio mokymosi algoritmus užimtumo nustatymo uždavinyje, tačiau visi nagrinėjami algoritmai yra ne nuoseklūs. Literatūroje nerasta palaipsnio tipo algoritmų šiam uždaviniui spręsti. Palyginimui pasirinkti rezultatai tinkami dėl savo nagrinėjamo uždavinio pritaikymo, o ne dėl nagrinėjamų algoritmų tipo. Minėtame [4] straipsnyje pateikti

rezultatai gali būti laikomi papildomi siūlomo palaipsnio PMM parametrų vertinimo algoritmo (P_DPMM_PVA) efektyvumui nustatyti.

20 lentelėje pateikiami [4] straipsnyje gauti atpažinimo rezultatai. Stulpeliuose pateikti atpažinimo rezultatai gauti su tiesinės regresijos klasifikatoriumi (TRK), atraminių vektorių klasifikatoriumi (AVK), dirbtiniais neuroniniai tinklais (DNT). Matyti, kad didžiausia pasiekta F-rodiklio reikšmė lygi 0,956, o P_DPMM_PVA algoritmo gauta F-rodiklio reikšmė yra 0,97. Galime teigti, kad P_DPMM_PVA algoritmas nenusileidžia jau egzistuojantiems literatūroje aprašytiems algoritmams ir yra efektyvus užimtumo nustatymo uždaviniui spręsti.

20 lentelė: Aukščiausius F-rodiklio rezultatus gavę algoritmai (TRK, AVK ir DNT) užimtumo nustatymo uždavinyje [4].

	TRK	**AVK**	**DNT**
Apšvietimas	95,6	95,55	95,32
Temperatūra	89,8	89,71	89,72

5.3 Pulsarų nustatymas

Pulsaras – besisukanti neutroninė žvaigždė, skleidžianti elektromagnetinę spinduliuotę radijo bangų, šviesos, rentgeno ir gama spindulių pavidalu. Pulsarai spinduliuoja pluoštu, panašiu į siaurą kūgį iš magnetinių polių sričių, statmenai savo paviršiui. Dėl spinduliuotės ir pulsaro sukimosi apie ašį, kuri dažniausiai nesutampa su magnetine ašimi, spinduliavimas (kaip trumpi impulsai) iš Žemės registruojamas tik tada, kai spinduliavimo pluoštas atsisuka į Žemę. Šis modelis periodiškai kartojasi pulsarui greitai besisukant [110, 111].

Pulsarai yra labai svarbūs, nes gali būti naudojami kaip erdvės-laiko, tarpžvaigždinės terpės, super-skysčio, materijos būsenų zondai [110–112]. Šiuo metu Paukščių take, Magelano debesyje, žvaigždžių spiečiuje yra apie 2200 pulsarų. Tačiau pulsarų paieška nėra paprasta užduotis. Pulsarų atradimas apima periodinių signalų identifikavimą stebėjimo duomenyse. Tuomet šie duomenys sumažinami iki diagnostinių verčių ir grafinių atvaizdų, vadinamų kandidatu, rinkinio [113]. Deja, daugumą kandidatų sukelia radijo dažnio trukdžiai (RFI) ir triukšmas, kurie iš tikrųjų nėra pulsarai [110, 111].

Pulsarų tyrimai atliekami nukreipiant teleskopą į dangų kelioms minutėms ar valandoms. Išsaugomi stebėjimo duomenys, o teleskopas nukreipiamas į kitą dangaus sritį ieškoti naujų pulsarų [114]. Pulsarų kandidatai – tai užregistruoti radijo

signalų grafikai ir statistiniai duomenys, naudojami tolimesnei analizei. Kandidatų autentiškumas toliau tikrinamas rankiniu arba automatiniu būdu [110]. Perspektyvių kandidatų, kuriuos dar kartą būtų galima stebėti, atranka iki šiol priklauso nuo žmonių – ekspertų, tikrinimo patikimumo. Tačiau žmogaus atliekamas tikrinimas yra subjektyvus, daug laiko reikalaujantis ir klaidoms neatsparus procesas [110, 113]. Be to, tolimesnei analizei ir atrinktų tikėtinų pulsarų patvirtinimui skiriamas papildomas teleskopo darbo laikas. Pastaruoju metu orientuojamasi į mašininio mokymosi metodus, siekiant išspręsti kandidatų „atrankos“ uždavinį. Čia kandidato „atranka“ yra procesas, kurio metu nusprendžiama, kuris kandidatas iš tikrųjų yra arba nėra pulsaras [110].

Dirbtiniai neuroniniai tinklai yra dominuojanti mašininio mokymosi metodika, naudojama šioje srityje. Kiek yra žinoma, nebuvo atliekami jokie tyrimai, skirti įvertinti palaipsnių PMM parametrų vertinimo algoritmo naudojamumą pulsarų identifikavimo uždavinyje.

Šiame darbe pulsarų identifikavimo uždaviniui spręsti siūlomi P_DPMM_PVA ir P_GPMM_PVA algoritmai. Siūlomi algoritmai yra išbandyti su viešai prieinamu HTRU2 duomenų rinkiniu [110] ir įrodo pranašumą prieš kitus literatūroje naudojamus klasifikatorius.

5.3.1 Susiję darbai

Dirbtiniai neuroniniai tinklai ir perceptrono metodai yra efektyvūs klasifikatoriai esant atskiriamoms klasėms. Nepaisant to, dažnai pasitaiko atvejų, kai stebėjimai yra dviprasmiški, priklausantys daugiau kaip vienai klasei. Dirbtiniai neuroniniai tinklai konverguoja, jei klasės yra atskiriamos plokštuma, o kitu atveju gali ir nekonverguoti.

[115] straipsnyje pasiūlytas mašininio mokymosi metodas pulsarų kandidatų atrankos uždavinyje. Minėtame darbe kiekvienas kandidatas aprašomas dvylikos požymių vektoriumi, o dirbtiniai neuroniniai tinklai naudojami puslarų kandidatams identifikuoti. [116] straipsnio autoriai kandidatams aprašyti naudojo dešimties požymių stebėjimų vektorius, kuriais apmokė dirbtinių neuroninių tinklų klasifikatorių. Straipsnio autoriai [113] straipsnio autoriai pasiūlė SPINN sistemą, naudojančią dirbtinį neuroninį tinklą ir stebėjimų vektorius, sudarytus iš šešių požymių. [114] straipsnio autoriai pristato PEACE (*Pulsar Evaluation Algorithm for Candidate Extraction*). Taip pat [110] straipsnio autoriai nagrinėja kandidatų filtravimo uždavinį, taikydami modifikuoto sprendimo medžio (angl. *Modified Decision*

Tree) metodą. Autoriai siūlo naują kandidatų atrankos metodą, naudodami Gauso Hellingerio greitąjį sprendimų medį (angl. *Gaussian Hellinger Very Fast Decision Tree*).

Literatūros apžvalga rodo, kad mašininio mokymosi metodai yra nauji pulsarų identifikavimo srityje, kadangi dar gana neseniai visas procesas buvo vykdomas rankiniu būdu. Dėl šio proceso vykdymo apribojimų ir milžiniško duomenų kiekio rankinis identifikavimas tampa neįmanomas ir nepraktiškas. Taip pat pažymima, kad šioje srityje populiariausi naudojami klasifikatoriai yra neuroniniai tinklai ir nėra nagrinėta kitokių mašininio mokymosi metodų, pagrįstų PMM, kurie būtų naudojami šio uždavinio sprendime.

5.3.2 Eksperimentinis tyrimas

Tyrimo tikslas – pritaikyti pasiūlytus palaipsnius PMM parametrų vertinimo algoritmus pulsarų identifikavimo uždavinyje ir ištirti jų efektyvumą.

Eksperimentinio tyrimo metu nagrinėjamas požymių aibės tinkamumas pulsarams identifikuoti. Algoritmų efektyvumui įvertinti pateiksime keletą klasifikavimo rodiklių. Įvertintų palaipsnių algoritmų efektyvumus lyginsime su kitais literatūroje publikuotais rezultatais ir bandysime įvertinti tarpusavio pranašumus.

Tyrimo duomenys

Eksperimentams atlikti pasirinktas HTRU2 duomenų rinkinys [110, 117]. HTRU2 duomenų rinkinys yra naujausias viešai prieinamas duomenų rinkinys [110], kuriame yra pulsarų kandidatų imtis, surinkta HTRU (angl. *High Time Resolution Universe*) tyrimo metu [110]. HTRU tyrimas skirtas dangaus tyrimui, kuris pradėtas 2008 metais.

Siekiant įgyvendinti greitą analizę, mašininio mokymosi metodai yra naudojami automatiškai identifikuoti pulsarų kandidatus. Šiam tikslui plačiai naudojamos klasifikavimo sistemos [110, 115, 116, 118–120], kuriose kandidatų duomenų rinkiniai traktuojami kaip dviejų klasių klasifikavimo uždavinys. Čia tikrų pulsarų kandidatų aibė yra stebėjimų mažuma, o ne pulsarų kandidatų aibė sudaro duomenų daugumą. Šiuo metu kandidatai nėra ženklinti kelių klasių etiketėmis, kadangi duomenų anotacijos darbas yra per brangus.

HTRU2 duomenų rinkinyje yra 16259 netikri RFI / triukšmo sukelti kandidatai (ne pulsarai) ir 1639 realūs pulsarai. Šiuos visus duomenis patikrino ekspertinė žmonių anotatorių grupė.

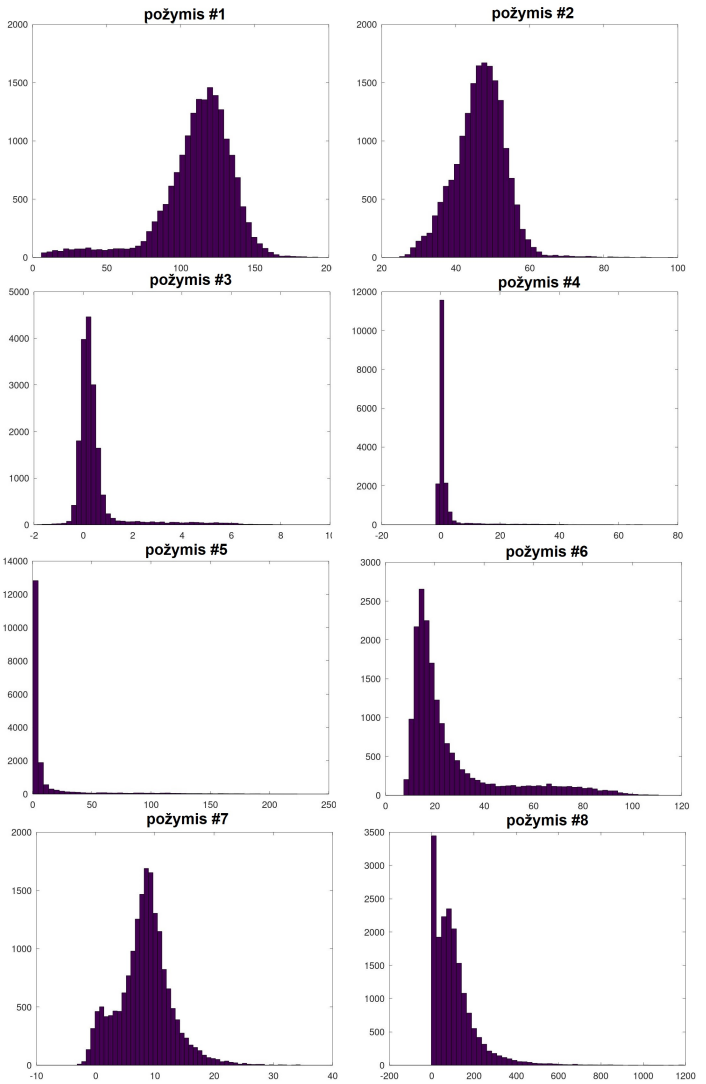
HTRU2 duomenų rinkinyje kiekvienas pulsaro kandidatas aprašomas aštuonių požymių $\mathbf{X}_i \in \{X_1, \ldots, X_8\}$ vektoriumi, pateikiančiu tam tikrą statistinę informaciją apie HTRU signalus. Pirmieji keturi požymiai yra statistiniai duomenys, gauti iš integruoto impulso profilio (angl. *Integrated Pulse Profile*). Tai tolydžių kintamųjų masyvas, apibūdinantis signalo versiją, kurios vidurkis apskaičiuojamas tiek pagal laiką, tiek pagal dažnį. Likusieji keturi požymiai panašiai gaunami iš DM-SNR kreivės [110]. Požymiai detalizuojami toliau:

1. Integruoto profilio vidurkis (angl. *Mean of the integrated profile*);
2. Integruoto profilio standartinis nuokrypis (angl. *Standard deviation of the integrated profile*);
3. Integruoto profilio eksceso koeficientas (angl. *Excess kurtosis of the integrated profile*);
4. Integruoto profilio asimetriškumo koeficientas (angl. *Skewness of the integrated profile*);
5. DM-SNR kreivės vidurkis (angl. *Mean of the DM-SNR curve*);
6. DM-SNR kreivės standartinis nuokrypis (angl. *Standard deviation of the DM-SNR curve*);
7. DM-SNR kreivės eksceso koeficientas (angl. *Excess kurtosis of the DM-SNR curve*);
8. DM-SNR kreivės asimetriškumo koeficientas (angl. *Skewness of the DM-SNR curve*).

Klasifikavimo proceso tikslas – identifikuoti duotus kandidatus kaip pulsarus ir ne pulsarus. Klasės dvejetainės etiketės – $Y = \{0,1\}$, kur $Y = 0$ reiškia, kad kandidatas yra ne pulsaras, o kai $Y = 1$, tai reiškia, kad kandidatas yra pulsaras.

HTRU2 duomenų rinkinio požymių histograma pavaizduota 22 paveiksle. Histogramoje pavaizduoti atskiri požymiai: integruoto profilio vidurkis, integruoto profilio standartinis nuokrypis, integruoto profilio eksceso koeficientas, integruoto profilio asimetriškumo koeficientas, DM-SNR kreivės vidurkis, DM-SNR kreivės standartinis nuokrypis, DM-SNR kreivės eksceso koeficientas, DM-SNR kreivės asimetriškumo koeficientas.

Matyti, kad požymiai yra vienamodaliniai. Požymių „integruoto profilio standartinis nuokrypis“, „integruoto profilio eksceso koeficientas“, „integruoto profilio asimetriškumo koeficientas“, „DM-SNR kreivės vidurkis“, „DM-SNR kreivės standartinis nuokrypis“, „DM-SNR kreivės eksceso koeficientas“, „DM-SNR kreivės asimetriškumo koeficientas“ histogramos krypsta į dešinę, o požymio „integruoto profilio vidurkis“ histograma krypsta į kairę. Ilgos „uodegos“ rodo, kad požymių skirstiniai galimai yra ne Gauso. Tad PMM būsenoms modeliuoti tinkamesnis galėtų būti Dirichlė skirstinys.



22 pav.: Aštuonių HTRU2 duomenų rinkinio požymių histograma.

Tyrimo sąlygos

Duomenų rinkinys buvo padalytas į du – mokymo ir pakartotinio modelio parametrų vertinimo – poaibius. Pirmąjį pradinio modelio mokymo poaibį sudaro 7898 stebėjimų vektoriai. Antrąjį poaibį modelio parametrų pakartotiniam vertinimui realiu laiku sudaro 10000 stebėjimo vektorių. Nustatyti PMM parametrai šio duomenų rinkinio apdorojimui pateikti 21 lentelėje.

21 lentelė: Nustatytos Gauso ir Dirichlė PMM parametrų vertės pulsarų identifikavimo uždaviniui spręsti.

	Parametrai		
Duomenų rinkinys	N	**A**	π
HTRU2	2	$\begin{bmatrix} 0{,}92 & 0{,}08 \\ 0{,}76 & 0{,}24 \end{bmatrix}$	$\begin{bmatrix} 0{,}5 \\ 0{,}5 \end{bmatrix}$

HTRU2 duomenų rinkinys apdorotas toliau aprašytu būdu. P_DPMM_PVA algoritmo atveju stebėjimų normalizavimas atliktas pagal (20) *Softmax* formulę, kad visi stebėjimo vektorių elementai būtų teigiami, o jų suma lygi vienetui. Tačiau P_GPMM_PVA algoritmo atveju *Softmax* normalizavimas neatliktas. Pradinės modelio parametrų vertės priskirtos apdorojant mokymo duomenis momentų metodu. Nustatytas algoritmo stabdymo kriterijus $\epsilon = 0{,}01$. Stebėjimų klasifikavimas atliktas pagal Bajeso klasifikavimo taisyklę. Atpažinimo tikslumas apskaičiuotas palaipsnių algoritmų pakartotinio parametrų vertinimo dalyje. Sykiu taip pat apskaičiuotas tiek bendras, tiek kiekvienos PMM būsenos atpažinimo tikslumas. Paskui pateikiamos algoritmų efektyvumo nustatymo metrikos ir atliktos požymių atrankos analizė.

Tyrimo priemonės

Pulsarų identifikavimo eksperimentas modeliuotas MATLAB aplinkoje. Tam sukurtas pulsarų identifikavimo sistemos prototipas, leidžiantis vykdyti eksperimentus bei vertinti jų rezultatus.

Tyrimo eiga

Pirmiausia apibrėžiami algoritmo vertinimo rodikliai. Tikrai teigiamas (TP) rodiklis yra kandidatų, kurie yra pulsarai ir yra klasifikuojami kaip pulsarai,

skaičius. Tikrai neigiamas (TN) rodiklis yra kandidatų, kurie yra ne pulsarai ir yra klasifikuojami kaip ne pulsarai, skaičius. Klaidingai neigiamas (FN) rodiklis yra kandidatų, kurie yra pulsarai, tačiau klasifikuojami kaip ne pulsarai, skaičius. Klaidingai teigiamas (FP) rodiklis yra kandidatų, kurie yra ne pulsarai, tačiau klasifikuojami kaip pulsarai, skaičius.

Papildomi rodikliai apskaičiuojami įvairių algoritmų efektyvumui palyginti. Šiuo atveju geras klasifikatorius turėtų maksimizuoti teisingumą (angl. *accuracy*) (37 formulė), tikslumą (angl. *precision*) (35 formulė), atrinkimą (angl. *recall*) (36 formulė), F-rodiklį (angl. *F-score*) (38 formulė), specifiškumą (angl. *specificity*) ir geometrinį vidurkį (angl. *Gmean*):

$$KTR = \frac{FP}{TN + FP}, \tag{39}$$

$$Specificity = \frac{TN}{FP + TN}, \tag{40}$$

$$GMean = \sqrt{\left(\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}\right)}. \tag{41}$$

Klaidingai teigiamas rodiklis (angl. *false positive rate*, santr. KTR) turėtų būti minimizuojamas. Šiuose eksperimentuose visos aukščiau minėtos metrikos yra apskaičiuojamos norint visapusiškai įvertinti siūlomų algoritmų efektyvumą.

Eksperimentų rezultatai

Eksperimentais siekiama patikrinti P_DPMM_PVA algoritmo galimybes identifikuoti pulsarus, apdorojus daugiamačius stebėjimo duomenis. P_DPMM_PVA algoritmo klasifikavimo tikslumas palygintas su P_GPMM_PVA algoritmu.

22 ir 23 lentelėse pateikti pulsarams nustatyti gautos maišaties matricos vertės P_DPMM_PVA ir P_GPMM_PVA algoritmais. Šios maišaties matricų vertės toliau naudojamos apskaičiuojant reikalingas algoritmų efektyvumo vertinimo metrikas. Šios metrikos pateiktos 24 lentelėje. Matome, kad P_DPMM_PVA algoritmo atveju F-rodiklis siekia 0,57, o P_GPMM_PVA algoritmo atveju – 0,36. F-rodikliai abiejų algoritmų atveju nėra pakankamai dideli, tačiau pažymėtina, kad P_DPMM_PVA algoritmo atveju KTR rodiklis yra reikšmingai mažas – 0,06.

22 lentelė: Pulsarams identifikuoti gautos maišaties matricos vertės, apdorojus duomenis P_DPMM_PVA algoritmu.

		Priskirta klasė	
		Pulsaras	**Ne pulsaras**
Tikra klasė	**Pulsaras**	475	89
	Ne pulsaras	662	8814

23 lentelė: Pulsaras identifikuoti gautos maišaties matricos vertės, apdorojus duomenis P_GPMM_PVA algoritmu.

		Priskirta klasė	
		Pulsaras	**Ne pulsaras**
Tikra klasė	**Pulsaras**	514	50
	Ne pulsaras	1735	7701

24 lentelė: Pulsarų nustatymo uždavinio sprendimo metu gauti rodikliai, apdorojus duomenis P_DPMM_PVA ir P_GPMM_PVA algoritmais. P_DPMM_PVA visais atvejais duoda geresnius rodiklius už P_GPMM_PVA.

	P_DPMM_PVA	**P_GPMM_PVA**
Teisingumas	0,92	0,82
KTR	0,07	0,18
Tikslumas	0,43	0,23
Atrinkimas	0,84	0,91
F-rodiklis	0,57	0,37
Specifiškumas	0,93	0,82
Geometrinis vidurkis	0,89	0,86

Kiekvienos PMM būsenos atpažinimo tikslumas (%) ir bendras atpažinimo tikslumas pateikti 25 lentelėje. Apdorojus HTRU2 duomenų rinkinį, P_GPMM_PVA algoritmas neteisingai suklasifikavo 1785 stebėjimus iš 10000 stebėjimų (atpažinimo tikslumas siekė 82,15 %). O P_DPMM_PVA neteisingai suklasifikavo tik 711 stebėjimus (atpažinimo tikslumas siekė 92,89 %). Užimtumo nustatymo duomenų palaipsnio klasifikavimo atveju P_DPMM_PVA yra 10 % tikslesnis už P_GPMM_PVA.

25 lentelė: P_DPMM_PVA ir P_GPMM_PVA algoritmų atpažinimo tikslumas (%) apdorojant HTRU2 duomenų rinkinį.

	P_GPMM_PVA		**P_DPMM_PVA**	
Duomenų rinkinys	Būsena #1	Būsena #2	Būsena #1	Būsena #2
HTRU2	81,61	91,14	93,41	84,22
	82,15		92,89	

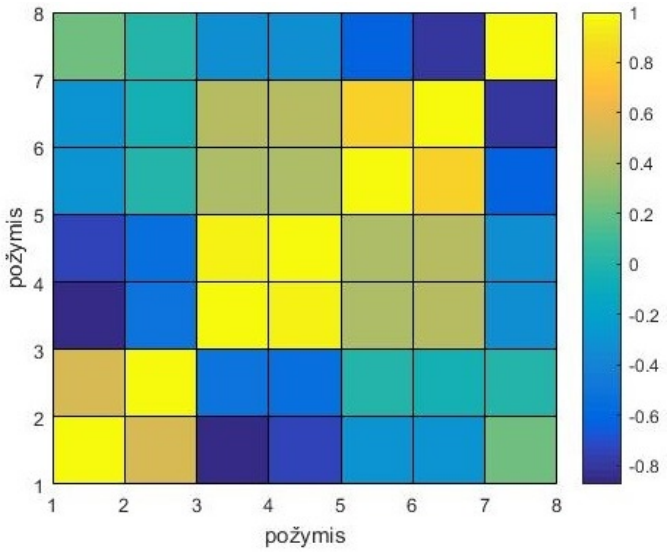
P_DPMM_PVA algoritmo efektyvumas palygintas su [5] straipsnyje aprašytu „Fuzzy KNN“ algoritmu (žr. 26 lentelė). 26 lentelės pirmame stulpelyje pateikiami algoritmų efektyvumo rodikliai (teisingumas, F-rodiklis, geometrinis vidurkis, KTR). Antrame stulpelyje pateikti apskaičiuotos minėtų rodiklių vertės P_DPMM_PVA algoritmo atveju, o trečiame – „Fuzzy KNN“ algoritmo rezultatai, pateikti [5] straipsnyje. Literatūroje nebuvo rasta pulsarų nustatymo uždaviniui spręsti aprašytų palaipsnių algoritmų. Todėl šis algoritmas palyginimui pasirinktas, nes minėtame straipsnyje eksperimentai atlikti su HTRU2 duomenų rinkiniu.

26 lentelė: P_DPMM_PVA algoritmo ir „Fuzzy KNN“ [5] efektyvumo lyginimas, apdorojant HTRU2 duomenis pulsarų nustatymo uždavinyje.

	Dirichlė PMM	**Fuzzy KNN**
Teisingumas	0,929	0,978
F-rodiklis	0,57	0,873
Geometrinis vidurkis	0,88	0,961
KTR	0,06	0,17

26 lentelėje pateikti abiejų algoritmų efektyvumo rodikliai (teisingumas, F-rodiklis, geometrinis vidurkis, KTR). Matome, kad P_DPMM_PVA algoritmo atveju teisingumo rodiklis yra 1,05 karto mažesnis, F-rodiklis – 1,5 karto mažesnis, geometrinis vidurkis – 1,09 karto mažesnis, o KTR – 2,8 karto mažesnis už „Fuzzy KNN“ klasifikatoriaus gautas rodiklių reikšmes. P_DPMM_PVA algoritmo gauta F-rodiklio reikšmė yra mažesnė už „Fuzzy KNN“, tačiau reikšmingas rezultatas lieka tas, kad ji vis dar viršija 0,5 ribą. P_DPMM_PVA algoritmo gauta KTR rodiklio reikšmė yra reikšmingai mažesnė už „Fuzzy KNN“. Tai rodo, kad klaidingai teigiamas rodiklis yra minimizuojamas (pulsaro etiketė priskiriama ne pulsarams daug rečiau negu „Fuzzy KNN“ klasifikatoriaus atveju).

Antrasis tyrimų tikslas – patikrinti, ar duomenyse yra ryšys tarp požymių ir ar jis turi įtakos atpažinimo tikslumui. Eksperimento metu apskaičiuotos koreliacijos koeficientų matricos reikšmės ir P-reikšmės tarp HTRU2 stebėjimų vektorių požymių (žr. 23 paveikslas).



23 pav.: HTRU2 požymių koreliacijos koeficientų matrica.

Iš 27 lentelėje pateiktų rezultatų matyti, kad koreliacijos koeficientų reikšmės daugeliu atvejų yra normalios. Tik tarp #3 ir #4, #8 ir #7 požymių egzistuoja labai stipri koreliacija.

28 lentelėje pateiktos apskaičiuotos koreliacijos P-reikšmės. Matyti, kad požymių (#3 ir #4, #8 ir #7) P-reikšmės yra mažesnės už reikšmingumo lygį (0,05), tad galime teigti, kad atitinkamos koreliacijos tarp minėtų požymių yra reikšmingos. Stipri koreliacija tarp šių požymių rodo, kad pašalinus atitinkamą požymį iš stipriai koreliuotų požymių poros ir apmokius algoritmą atpažinimo tikslumas galėtų pagerėti.

27 lentelė: HTRU2 požymių tarpusavio koreliacijos koeficientų reikšmės.

Požymis	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**
#2	0,55						
#3	-0,87	-0,52					
#4	-0,74	-0,54	0,95				
#5	-0,30	0,01	0,41	0,41			
#6	-0,31	-0,05	0,43	0,42	0,80		
#7	0,23	0,03	-0,34	-0,33	-0,62	-0,81	
#8	0,14	0,03	-0,21	-0,21	-0,35	-0,58	0,92

#1 – integruoto profilio vidurkis; #2 – integruoto profilio standartinis nuokrypis; #3 – integruoto profilio eksceso koeficientas; #4 – integruoto profilio asimetriškumo koeficientas; #5 – DM-SNR kreivės vidurkis; #6 – DM-SNR kreivės standartinis nuokrypis; #7 – DM-SNR kreivės eksceso koeficientas; #8 – DM-SNR kreivės asimetriškumo koeficientas

28 lentelė: HTRU2 požymių koreliacijos P-reikšmės.

Požymis	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**
#2	0,00						
#3	0,00	0,00					
#4	0,00	0,00	0,00				
#5	0,00	0,36	0,00	0,00			
#6	0,00	0,00	0,00	0,00	0,00		
#7	0,00	0,00	0,00	0,00	0,00	0,00	
#8	0,00	0,00	0,00	0,00	0,00	0,00	0,00

#1 – integruoto profilio vidurkis; #2 – integruoto profilio standartinis nuokrypis; #3 – integruoto profilio eksceso koeficientas; #4 – integruoto profilio asimetriškumo koeficientas; #5 – DM-SNR kreivės vidurkis; #6 – DM-SNR kreivės standartinis nuokrypis; #7 – DM-SNR kreivės eksceso koeficientas; #8 – DM-SNR kreivės asimetriškumo koeficientas

Toliau eksperimentai atliekami remiantis šiais rezultatais. Modelio apmokymas ir pakartotinis modelio parametrų vertinimas atliekamas iš požymių vektoriaus pašalinus 4-tąjį požymį ir po to pašalinus 8-tąjį požymį.

29 lentelėje pateikti P_DPMM_PVA algoritmo atpažinimo tikslumo ir F-rodiklio rezultatai, kai naudojamos skirtingos požymių aibės. Stulpeliuose pateikti

atpažinimo rezultatai, gauti apmokius modelį su visa nemodifikuota duomenų aibe ir modifikuota duomenų aibe. Trys modifikuotos duomenų aibės gautos atitinkamai pašalinus požymius „integruoto profilio asimetriškumo koeficientas“ (4-tasis požymis), „DM-SNR kreivės asimetriškumo koeficientas“ (8-tasis požymis) ir abu požymius (4-tąjį ir 8-tąjį) kartu). Pirmasis stulpelis rodo algoritmo atpažinimo rezultatus, kai naudojama visa pradinė požymių aibė. Antrasis stulpelis rodo algoritmo atpažinimo rezultatus, kai iš požymių aibės pašalintas 4-tas požymis, o trečiajame stulpelyje – kai pašalintas 8-tas požymis. Požymių šalinimas remiasi apskaičiuota koreliacijos koeficientų matrica.

29 lentelė: Pulsarų nustatymo uždavinio sprendimo su P_DPMM_PVA algoritmu rezultatai.

	Visi požymiai	**Pašalinus 4-tąjį požymį**	**Pašalinus 8-tąjį požymį**	**Pašalinus 4-tąjį ir 8-tąjį požymius**
Teisingumas	0,93	0,91	0,93	0,92
KTR	0,06	0,09	0,06	0,07
Tikslumas	0,43	0,36	0,44	0,39
Atrinkimas	0,83	0,86	0,82	0,74
F-rodiklis	0,57	0,51	0,58	0,51
Specifiškumas	0,94	0,91	0,94	0,93
Geometrinis vidurkis	0,88	0,88	0,88	0,83

29 lentelėje pateikti rezultatai rodo, kad pašalinus 8-ąjį požymį iš duomenų aibės ir atlikus apmokymo bei atpažinimo etapus, teisingumas padidėja. Pašalinus 8-tąjį ir 4-tąjį požymius kartu – atpažinimo tikslumas nekinta. Pašalinus 4-ąjį požymį – atpažinimo teisingumas nukrenta. Taip pat matyti, kad pašalinus 8-ąjį požymį, nežymiai pagerėja tikslumo rodiklis, tačiau sumažėja geometrinio vidurkio rodiklis, lyginant su nemodifikuota aibe.

Atlikus eksperimentus su modifikuotu duomenų rinkiniu (kai pašalinti tam tikri požymiai), galime daryti išvadą, kad pašalinę 8-tąjį požymį, galime šiek tiek pagerinti atpažinimo tikslumą. Taip pat šio požymio pašalinimas sumažina naudojamų skaičiavimams resursų kiekį, kadangi skaičiavimai atliekami su mažesnėmis matricomis ir vektoriais.

5.4 Skyriaus apibendrinimas

- Šiame darbe pasiūlytas palaipsnis daugiamačių PMM parametrų vertinimo algoritmas, kai modelio išvesties tikimybinis skirstinys yra daugiamatis Gauso (P_GPMM_PVA). Šis algoritmas pritaikytas pavieniams žodžiams atpažinti realiu laiku. Eksperimento metu išnagrinėta pradinio mokymo duomenų rinkinio dydžio įtaka atpažinimo tikslumui. Eksperimentiniai rezultatai parodė, kad turint pakankamą pradinį PMM parametrų vertinimo duomenų rinkinį gaunamas pavienių žodžių atpažinimo tikslumas yra didesnis kaip 90 %. Remiantis atliktais eksperimentais, galime daryti išvadą, kad P_DPMM_PVA gali būti efektyviai pritaikytas realaus laiko pavienių žodžių atpažinimo užduotims, pagrįstoms daugiamačiu PMM modeliu.

- Šiame darbe pasiūlytas palaipsnis daugiamačių PMM parametrų vertinimo algoritmas, kai modelio išvesties tikimybinis skirstinys yra daugiamatis Dirichlė (P_DPMM_PVA). Šis algoritmas pritaikytas patalpų užimtumo nustatymo uždaviniui spręsti. Eksperimentai atlikti su užimtumo nustatymo duomenų rinkiniu, kuriame daugiamačiai stebėjimai surinkti iš įvairių skaitiklių / sensorių. Atlikti eksperimentai parodė, kad pasiūlytas algoritmas efektyviai suklasifikuoja turimus stebėjimus – pasiektas 97 % atpažinimo tikslumas. Palyginus algoritmą su literatūroje aprašytais šio uždavinio sprendimui taikytais nepalaipsniais algoritmais (palaipsnių algoritmų, paremtų PMM, taikymo literatūroje nerasta) matyti, kad jis efektyvumu šiems nenusileidžia.

- P_DPMM_PVA algoritmas pritaikytas pulsarų nustatymo uždaviniui spręsti. Eksperimentais, atliktais su HTRU2 duomenų rinkiniu, nustatyta, kad šiuo algoritmu pasiektas 92 % atpažinimo tikslumas. Algoritmas palygintas su literatūroje aprašytu šio uždavinio sprendimui taikomu „Fuzzy KNN“ algoritmu. Eksperimentų rezultatai parodė, kad P_DPMM_PVA algoritmo atveju atpažinimo tikslumas yra 1,05 karto mažesnis už „Fuzzy KNN“ klasifikatoriaus. Tačiau P_DPMM_PVA algoritmo gauta KTR rodiklio reikšmė yra reikšmingai mažesnė už „Fuzzy KNN“. Tai rodo, kad klaidingai teigiamas rodiklis yra minimizuojamas (pulsaro etiketė priskiriama ne pulsarams daug rečiau negu „Fuzzy KNN“ klasifikatoriaus atveju).

- Užimtumo nustatymo ir pulsarų nustatymo uždavinių sprendimo atvejais buvo tiriama duomenų rinkinio požymio aibės įtaka atpažinimui. Ištirta, kurie požymiai koreliuoja tarpusavyje, ir pagal gautus koreliacijos rezultatus atlikti

eksperimentai pašalinus labai stiprios koreliacijos požymius. Atlikti eksperimentai parodė, kad tam tikrais atvejais stebėjimų atpažinimo tikslumas išlieka toks pat kaip ir nemodifikuotos požymių aibės atveju, o kartais atpažinimo tikslumas pagerėja. Tokiu atveju pašalinus atitinkamus požymius iš duomenų aibės, atpažinimo tikslumas nenukenčia, o skaičiavimams reikalingų resursų kiekis sumažėja.

6 BENDROS IŠVADOS

Disertacijoje išnagrinėtas palaipsnis paslėptųjų Markovo modelių daugiamačių parametrų vertinimo uždavinys. Sudaryti algoritmai palaipsniam paslėptųjų Markovo modelių parametrų vertinimui su daugiamačiais Gauso ir ne Gauso (Dirichlė) PMM išvesties tikimybiniais skirstiniais.

Pagrindiniai rezultatai:

- Remiantis didžiausio tikėtinumo metodu ir klasikiniu MVM algoritmu, suformuoti palaipsniai PMM parametrų vertinimo algoritmai, kai daugiamačiai Gauso ir Dirichlė dėsniai yra modelio būsenų išvesties tikimybiniai skirstiniai (atitinkamai, P_GPMM_PVA ir P_DPMM_PVA).
- PMM parametrų vertinimo dalyje būsenų perėjimo tikimybes siūloma skaičiuoti remiantis Čapmano ir Kolmogorovo lygtimi. Taip pakeista tradicinė tiesioginio-atbulinio sklidimo procedūra, įprastai taikoma „Baum-Welch“ algoritmuose, o palaipsniuose – tiesioginio sklidimo procedūra.
- Atliktas pasiūlytų palaipsnių algoritmų eksperimentinis tyrimas su sintetiniais duomenimis Monte-Karlo metodu, pavienių žodžių, užimtumo nustatymo ir pulsarų nustatymo duomenų rinkiniais.

Atlikus eksperimentinius tyrimus, suformuluotos šios išvados bei jas patvirtinantys eksperimentų rezultatai:

- Palaipsnis PMM parametrų vertinimo algoritmas, kai išvesties tikimybinis skirstinys yra Gauso, yra greitesnis ir klasifikavimo tikslumu prilygstantis tradiciniam rinkinio („Baum-Welch“) algoritmui.
  - Atlikus eksperimentus gautas palaipsnio algoritmo (P_GPMM_PVA) duomenų apdorojimo greitis yra didesnis už rinkinio algoritmo. Palaipsnio algoritmo skaičiavimo laikas yra 3–9 kartus greitesnis už rinkinio algoritmo, kai apdorojami skirtingo dimensijų skaičiaus stebėjimai.
  - Rinkinio ir Palaipsnio (P_GPMM_PVA) algoritmų būsenų perėjimo tikimybių matricos, gautos PMM parametrų vertinimo metu apdorojant iki 10000 stebėjimų, nesiskiria.
  - Palaipsniu (P_DPMM_PVA) ir rinkinio algoritmais gautų PMM parametrų įverčių standartinių paklaidų santykis rodo, kad palaipsnis algoritmas, apdorodamas skirtingų dimensijų stebėjimus, duoda minimaliai

nuo rinkinio algoritmo besiskiriančius parametrų įverčius. Palaipsnio ir rinkinio PMM parametrų vertinimo algoritmų standartinių paklaidų santykis svyruoja nuo 1,1 iki 1,5 karto, priklausomai nuo apdorojamų duomenų dimensijų ir sklaidos didėjimo.

– Atliktas eksperimentinis tyrimas, kuriame lyginti palaipsnio (P_GPMM_PVA) algoritmo rezultatai su žinomu palaipsniu algoritmu. Rezultatai parodė, kad būsenų perėjimo tikimybių skaičiavimas su Čapmano ir Kolmogorovo lygtimi pagerina parametrų artėjimą prie tikrųjų modelio parametrų reikšmių, lyginant su algoritmu, kuriame būsenų perėjimo tikimybių matrica skaičiuojama su tiesioginio sklidimo procedūra. P_GPMM_PVA algoritmu apdorojus trimačius stebėjimus, standartinė PMM parametrų įverčių paklaida siekė 0,008, o žinomo algoritmo – 0,19.

- Egzistuoja pakankamas pradinis duomenų rinkinys, skirtas pradinei algoritmo aproksimacijai, kuris užtikrina algoritmo stabilumą ir neleidžia jam konverguoti į išsigimusius lokalius tikėtinumo funkcijos ekstremumus.

  – Atliktų eksperimentų, skirtų ištirti pradinės aproksimacijos dydžio įtaką atpažinimo tikslumui, gauti rezultatai parodė, kad standartinė parametrų įverčių paklaida mažėja, kai didėja bendras duomenų rinkinio dydys ir kai minimalus pradinės aproksimacijos duomenų rinkinio dydis yra pakankamas. Penkių klasterių dvimačių ir aštuonmačių stebėjimų apdorojimo atveju modelio parametrų įverčių standartinė paklaida nesiskiria, pradinio duomenų rinkinio dydis yra 300 stebėjimų ir didinamas iki 400.

  – Esant nedideliam klasterių (PMM būsenų) skaičiui, pradiniam mokymui naudojamų duomenų rinkinio dydis taip pat gali būti mažas, nepriklausomai nuo stebėjimo vektorių dimensijų skaičiaus. Tačiau didėjant klasterių skaičiui, turi didėti ir pradinės aproksimacijos duomenų rinkinys.

  – o Pritaikius palaipsnį Gauso PMM parametrų vertinimo algoritmą pavieniams žodžiams atpažinti pasiektas 97 % atpažinimo tikslumas „TI-DIGITS“ garsyno atveju, o „Spoken Arabic Digits“ garsyno atveju atpažinimo tikslumas siekė 91 %.

- Duomenų sklaida ir dimensijų skaičius lemia palaipsnio PMM parametrų vertinimo efektyvumą – kai didėja dimensijų skaičius, didėja ir PMM parametrų įverčių standartinė paklaida – skirtumas tarp tikrųjų parametrų reikšmių ir parametrų įverčių.
  - Standartinė paklaida, kai duomenys yra dvimačiai ir sklaida – 20, yra apie 4–6 kartus didesnė už standartinę paklaidą, gautą apdorojus 16-mačius duomenis, kurių sklaida yra 50.
  - Kai didėja duomenų sklaida ir dimensijos, atitinkamai didėja algoritmo ir skaičiavimo laikas.
  - Kai didėja apdorojamų duomenų kiekis, modelio parametrų įverčių standartinė paklaida atitinkamai mažėja.

Atlikus eksperimentinius tyrimus su palaipsniu Dirichlė PMM parametrų vertinimo algoritmu, suformuluotos šios išvados bei jas patvirtinantys eksperimentiniai rezultatai:

- Atliktų eksperimentų, skirtų patikrinti palaipsnio Dirichlė PMM parametrų vertinimo algoritmo efektyvumą lyginant su palaipsniu Gauso PMM parametrų vertinimo algoritmu, rezultatai parodė, kad plačiai paplitę Gauso PMM negali būti taikomi uždaviniuose, kuriuose stebėjimai yra pasiskirstę pagal Dirichlė dėsnį – tokiu atveju Dirichlė PMM yra efektyvesnis ir duoda 26 % didesnį stebėjimų klasifikavimo tikslumą.

- Atlikus eksperimentus nustatyta, kad atpažinimo (klasifikavimo) tikslumas didėjant mokymo duomenų kiekiui išlieka stabilus ir nemažėja, o standartinė parametrų įverčių paklaida mažėja. Po pradinės aproksimacijos gauta PMM parametrų įverčių standartinė paklaida sumažėjo 2,5 karto pakartotinio parametrų vertinimo metu apdorojus 4000 stebėjimų.

- Pritaikius Dirichlė PMM praktiniuose uždaviniuose, gauti rezultatai parodė, kad užimtumo nustatymo duomenų rinkinio atveju pasiektas 97 % atpažinimo tikslumas, o pulsarų nustatymo duomenų rinkinio atveju atpažinimo tikslumas siekė 93 %.

- Požymių aibės modifikavimas turi įtakos atpažinimo tikslumui. Užimtumo nustatymo atveju atpažinimo tikslumas nekinta modifikavus požymių aibę –

iš jos atitinkamai pašalinus „vandens garų kiekis ore“, „santykinis drėgnumas“ ir „data“ požymius. Pulsarų nustatymo atveju – modifikavus požymių aibę (iš jos pašalinus tik „DM-SNR kreivės asimetriškumo koeficientas“ požymį) nežymiai (0,01 %) pagerėja atpažinimo tikslumas.

Palaipsnių algoritmų tyrimai yra naudingi kuriant kito tipo algoritmų palaipsnius atitikmenis ir juos taikant klasifikavimui realiu laiku.

LITERATŪROS SĄRAŠAS

[1] G. Teyou, “Deep Learning Acceleration Techniques for Real Time Mobile Vision Applications”. *arXiv*. *Computer Vision and Pattern Recognition*, e.print arXiv 1905.03418, 2019.

[2] R. Nishihara, P. Moritz, *et al.*, “Real-Time Machine Learning: The Missing Pieces”. *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*, isbn 9781450350686, p. 106–-110, 2017.

[3] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, and J. M. Buhmann, “Topology free hidden Markov models: Application to background modeling”. *Proceedings of the IEEE International Conference on Computer Vision*, vol. 1, no. C, pp. 294–301, 2001.

[4] B. Abade, D. Perez Abreu, and M. Curado, “A non-intrusive approach for indoor occupancy detection in smart environments”. *Sensors*, vol. 18, no. 11, p. 3953, 2018.

[5] T. M. Mohamed, “Pulsar selection using Fuzzy KNN classifier”. *Future Computing and Informatics Journal*, vol. 3, no. 1, pp. 1 – 6, 2018.

[6] L. Xie, V. A. Ugrinovskii, and I. R. Petersen, “Finite horizon robust state estimation for uncertain finite-alphabet hidden Markov models with conditional relative entropy constraints”. *SIAM Journal on Control and Optimization*, vol. 47, no. 1, pp. 476–508, 2008.

[7] J. J. Ford, V. Ugrinovskii, and J. Lai, “An infinite-horizon robust filter for uncertain hidden Markov models with conditional relative entropy constraints”. In *2011 Australian Control Conference*, pp. 499–506, IEEE, 2011.

[8] W. L. Kendall, G. C. White, J. E. Hines, C. A. Langtimm, and J. Yoshizaki, “Estimating parameters of hidden Markov models based on marked individuals: use of robust design data”. *Ecology*, vol. 93, no. 4, pp. 913–920, 2012.

[9] O. Cappé, “Online EM Algorithm for Hidden Markov Models”. *Journal of Computational and Graphical Statistics*, vol. 20, no. 3, pp. 728–749, 2011.

[10] W. Khreich, E. Granger, A. Miri, and R. Sabourin, “A survey of techniques for incremental learning of HMM parameters”. *Information Sciences*, vol. 197, p. 105–130, Aug 2012.

[11] A. Kontorovich, B. Nadler, and R. Weiss, “On learning parametric-output HMMs”. In *Proceedings of the 30th International Conference on Machine Learning*, vol. 28 of *Proceedings of Machine Learning Research*, (Atlanta, Georgia, USA), pp. 702–710, PMLR, 17–19 Jun 2013.

[12] J. Unnikrishnan, V. V. Veeravalli, and S. P. Meyn, “Minimax robust quickest change detection”. *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1604–1614, 2011.

[13] J. Dong, M. Verhaegen, and F. Gustafsson, “Robust fault detection with statistical uncertainty in identified parameters”. *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5064–5076, 2012.

[14] B. C. Levy, “Robust hypothesis testing with a relative entropy tolerance”. *IEEE Transactions on Information Theory*, vol. 55, no. 1, pp. 413–421, 2008.

[15] A. Nilim and L. El Ghaoui, “Robust control of Markov decision processes with uncertain transition matrices”. *Operations Research*, vol. 53, no. 5, pp. 780–798, 2005.

[16] W. Wiesemann, D. Kuhn, and B. Rustem, “Robust Markov decision processes”. *Mathematics of Operations Research*, vol. 38, no. 1, pp. 153–183, 2013.

[17] J. Lai, J. J. Ford, P. O’Shea, and L. Mejias, “Vision-based estimation of airborne target pseudobearing rate using hidden Markov model filters”. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2129–2145, 2013.

[18] D. W. Park, J. Kwon, and K. M. Lee, “Robust visual tracking using autoregressive hidden Markov model”. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 1964–1971, IEEE, 2012.

[19] T. Vojir, J. Matas, and J. Noskova, “Online adaptive hidden Markov model for multi-tracker fusion”. *Computer Vision and Image Understanding*, vol. 153, pp. 109–119, 2016.

[20] Y. Ephraim and N. Merhav, “Hidden Markov processes”. *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1518–1569, 2002.

[21] O. Cappé, E. Moulines, and T. Rydén, *Inference in hidden Markov models*. Springer, 2010.

[22] V. Krishnamurthy and J. B. Moore, "On-line estimation of hidden Markov model parameters based on the Kullback-Leibler information measure". *IEEE Trans. Signal Processing*, vol. 41, pp. 2557–2573, 1993.

[23] T. Ryden, "On recursive estimation for hidden Markov models". *Stochastic Processes and their Applications*, vol. 66, no. 1, pp. 79 – 96, 1997.

[24] G. Mongillo and S. Deneve, "Online learning with hidden Markov models". *Neural Computation*, vol. 20, no. 7, pp. 1706–1716, 2008.

[25] V. B. Tadic, "Analyticity, Convergence, and Convergence Rate of Recursive Maximum-Likelihood Estimation in Hidden Markov Models". *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6406–6432, 2010.

[26] F. LeGland and L. Mevel, "Recursive estimation in hidden Markov models". In *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 4, pp. 3468–3473 vol.4, Dec 1997.

[27] I. Collings, V. Krishnamurthy, and J. B. Moore, "On-line identification of hidden Markov models via recursive prediction error techniques". *Signal Processing, IEEE Transactions on*, vol. 42, pp. 3535–3539, 1995.

[28] J. J. Ford and J. B. Moore, "Adaptive estimation of HMM transition probabilities". *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1374–1385, 1998.

[29] J. Vaičiulytė and L. Sakalauskas, "Recursive estimation of multivariate hidden Markov model parameters". *Computational Statistics*, 2019.

[30] J. Vaičiulytė and L. Sakalauskas, "Recursive parameter estimation algorithm of the Dirichlet hidden Markov model". *Journal of Statistical Computation and Simulation*, vol. 90, no. 2, pp. 306–323, 2020.

[31] M. Stamp, "A Revealing Introduction to Hidden Markov Models". 2015.

[32] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition". *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[33] Y. Ephraim and L. R. Rabiner, “On the relations between modeling approaches for information sources”. In *Proc. ICASSP*, vol. 88, pp. 24–27, 1988.

[34] B. G. Leroux, “Maximum-likelihood estimation for hidden Markov models”. *Stochastic processes and their applications*, vol. 40, no. 1, pp. 127–143, 1992.

[35] P. J. Bickel, Y. Ritov, T. Ryden, *et al.*, “Asymptotic normality of the maximum-likelihood estimator for general hidden Markov models”. *The Annals of Statistics*, vol. 26, no. 4, pp. 1614–1635, 1998.

[36] N. Li and M. Stephens, “Modeling linkage disequilibrium and identifying recombination hotspots using single-nucleotide polymorphism data”. *Genetics*, vol. 165, no. 4, pp. 2213–2233, 2003.

[37] P. K. Dunn and G. K. Smyth, “Beyond linear regression: The method of maximum likelihood”. In *Generalized Linear Models With Examples in R*, pp. 165–209, Springer, 2018.

[38] A. B. Owen, *Empirical likelihood*. Chapman and Hall/CRC, ISBN 9781584880714, 2001.

[39] A. Vexler, J. Yu, and N. Lazar, “Bayesian empirical likelihood methods for quantile comparisons”. *Journal of the Korean Statistical Society*, vol. 46, no. 4, pp. 518–538, 2017.

[40] J. Duchi, P. Glynn, and H. Namkoong, “Statistics of robust optimization: A generalized empirical likelihood approach”. *arXiv preprint arXiv:1610.03425*, 2016.

[41] W. Zucchini, I. L. Macdonald, and R. Langrock, *Hidden Markov models for time series: an introduction using R*. Crc Press, 2016.

[42] N. M. Nasrabadi, “Pattern recognition and machine learning”. *Journal of Electronic Imaging*, vol. 16, no. 4, 2007.

[43] G. Manogaran, V. Vijayakumar, R. Varatharajan, P. Malarvizhi Kumar, R. Sundarasekar, and C.-H. Hsu, “Machine learning based big data processing framework for cancer diagnosis using hidden Markov model and GM clustering”. *Wireless Personal Communications*, vol. 102, pp. 2099–2116, Oct 2018.

[44] R. Mattila, C. R. Rojas, V. Krishnamurthy, and B. Wahlberg, “Identification of hidden Markov models using spectral learning with likelihood maximization”. *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec 2017.

[45] T. Cheng, S. Dixon, and M. Mauch, “Improving piano note tracking by HMM smoothing”. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 2009–2013, Aug 2015.

[46] S. Tugaç and M. Efe, “Hidden Markov model based target detection”. In *2010 13th International Conference on Information Fusion*, pp. 1–7, July 2010.

[47] L. R. L. Rodrigues and E. L. Pinto, “HMM models and estimation algorithms for real-time predictive spectrum sensing and cognitive usage”. In *XXXV SIMPOSIO BRASILEIRO DE TELECOMUNICACOES E PROCESSAMENTO DE SINAIS-SBrT2017*, 2017.

[48] W. Khreich, E. Granger, A. Miri, and R. Sabourin, “On the memory complexity of the forward-backward algorithm”. *Pattern Recognition Letters*, vol. 31, no. 2, pp. 91–99, 2010.

[49] Qiang Huo and Chin-Hui Lee, “On-line adaptive learning of the continuous density hidden Markov model based on approximate recursive Bayes estimate”. *IEEE Transactions on Speech and Audio Processing*, vol. 5, pp. 161–172, March 1997.

[50] A. Bietti, F. Bach, and A. Cont, “An online EM algorithm in hidden (semi-)Markov models for audio segmentation and clustering”. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1881–1885, April 2015.

[51] R. J. Elliott, L. Aggoun, and J. B. Moore, *Hidden Markov models: estimation and control*. Springer, 2011.

[52] E. Granger, “A survey of techniques for incremental learning of HMM parameters”. *Information Sciences*, vol. 197, no. February 2014, pp. 105–130, 2012.

[53] U. Holst and G. Lindgren, “Recursive estimation in mixture models with Markov regime”. *IEEE Transactions on Information Theory*, vol. 37, no. 6, pp. 1683–1690, 1991.

[54] J. Stiller and G. Radons, “Online estimation of hidden Markov models”. *IEEE Signal Processing Letters*, vol. 6, no. 8, pp. 213–215, 1999.

[55] I. B. Collings and T. Rydén, “A new maximum likelihood gradient algorithm for on-line hidden Markov model identification”. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP’98 (Cat. No. 98CH36181)*, vol. 4, pp. 2261–2264, IEEE, 1998.

[56] J. Kivinen and M. K. Warmuth, “Exponentiated gradient versus gradient descent for linear predictors”. *Information and computation*, vol. 132, no. 1, pp. 1–63, 1997.

[57] Y. Singer and M. K. Warmuth, “Training algorithms for hidden Markov models using entropy based distance functions”. In *Proceedings of the 9th International Conference on Neural Information Processing Systems*, NIPS’96, (Cambridge, MA, USA), pp. 641–647, MIT Press, 1996.

[58] Y. Singer and M. K. Warmuth, “Training algorithms for hidden Markov models using entropy based distance functions”. In *Advances in Neural Information Processing Systems*, pp. 641–647, 1997.

[59] A. Garg and M. K. Warmuth, “Inline updates for HMMs”. In *INTERSPEECH*, 2003.

[60] R. M. Neal and G. E. Hinton, “A View of the EM Algorithm that Justifies Incremental, Sparse, and other Variants”. In *Learning in Graphical Models*, pp. 355–368. Springer Netherlands, 2012.

[61] Y. Gotoh, M. M. Hochberg, and H. F. Silverman, “Efficient training algorithms for HMMs using incremental estimation”. *IEEE transactions on speech and audio processing*, vol. 6, no. 6, pp. 539–548, 1998.

[62] V. V. Digalakis, “Online adaptation of hidden Markov models using incremental estimation algorithms”. *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 253–261, 1999.

[63] J. Mizuno, T. Watanabe, K. Ueki, K. Amano, E. Takimoto, and A. Maruoka, "On-line estimation of hidden Markov model parameters". In *International Conference on Discovery Science*, pp. 155–169, Springer, 2000.

[64] D. Vasquez, C. Laugier, and T. Fraichard, "Incremental learning of statistical motion patterns with growing hidden Markov models". In *Robotics Research*, pp. 75–86, Springer, 2010.

[65] G. Florez-Larrahondo, S. Bridges, and E. A. Hansen, "Incremental estimation of discrete hidden Markov models based on a new backward procedure". In *Proceedings of the National Conference on Artificial Intelligence*, vol. 20, p. 758, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[66] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, and J. M. Buhmann, "Topology free hidden Markov models: Application to background modeling". In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 1, pp. 294–301, IEEE, 2001.

[67] P. Baldi and Y. Chauvin, "Smooth on-line learning algorithms for hidden Markov models". *Neural Computation*, vol. 6, no. 2, pp. 307–318, 1994.

[68] O. Cappé, V. Buchoux, and E. Moulines, "Quasi-Newton method for maximum likelihood estimation of hidden Markov models". In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 4, pp. 2265–2268, IEEE, 1998.

[69] D. M. Titterington, "Recursive parameter estimation using incomplete data". *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 46, no. 2, pp. 257–267, 1984.

[70] E. Weinstein, M. Feder, and A. V. Oppenheim, "Sequential algorithms for parameter estimation based on the Kullback-Leibler information measure". *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 9, pp. 1652–1654, 1990.

[71] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Mar-

kov chains". *The annals of mathematical statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[72] O. Cappé and E. Moulines, "Online EM Algorithm for Latent Data Models". *Journal of the RoyalStatistical Society: Series B*, 71(3), pp.593–613, 2009.

[73] Y.-A. Ma, N. J. Foti, and E. B. Fox, "Stochastic gradient MCMC methods for hidden Markov models". In *Proceedings of the 34th International Conference on Machine Learning* (D. Precup and Y. W. Teh, eds.), vol. 70 of *Proceedings of Machine Learning Research*, (International Convention Centre, Sydney, Australia), pp. 2265–2274, PMLR, 06–11 Aug 2017.

[74] R. Mattila, C. R. Rojas, and B. Wahlberg, "Evaluation of spectral learning for the identification of hidden Markov models". *IFAC-PapersOnLine*, vol. 48, no. 28, pp. 897 – 902, 2015. 17th IFAC Symposium on System Identification SYSID 2015.

[75] Y. C. Subakan, J. Traa, P. Smaragdis, and D. Hsu, "Method of moments learning for left-to-right hidden Markov models". In *2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 1–5, Oct 2015.

[76] G. Cybenko and V. Crespi, "Learning hidden Markov models using nonnegative matrix factorization". *IEEE Transactions on Information Theory*, vol. 57, pp. 3963–3970, June 2011.

[77] B. Lakshminarayanan and R. Raich, "Non-negative matrix factorization for parameter estimation in hidden Markov models". In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pp. 89–94, Aug 2010.

[78] L. Ljung, "Analysis of recursive stochastic algorithms". *Automatic Control, IEEE Transactions on*, vol. 22, pp. 551–575, 1977.

[79] A. Arapostathis and S. I. Marcus, "Analysis of an identification algorithm arising in the adaptive estimation of Markov chains". *Mathematics of Control, Signals and Systems*, vol. 3, no. 1, pp. 1–29, 1990.

[80] J. Vaičiulytė and L. Sakalauskas, "Rekurentinis paslėptųjų Markovo modelių parametrų vertinimo algoritmas". *Computational Science and Techniques*, vol. 5, no. 1, pp. 529–540, 2017.

[81] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2009.

[82] S. V. Vaseghi, *Advanced digital signal processing and noise reduction.* John Wiley and Sons, 2000.

[83] O. Leveque, "Lecture notes on Markov chains". 2011. Online: http://www.hamilton.ie/ollie/Downloads/Mar1.pdf.

[84] B. A. Frigyik, A. Kapila, and M. R. Gupta, "Introduction to the Dirichlet Distribution and Related Processes". *UWEE Technical Report*, 2010.

[85] N. Bouguila, D. Ziou, and J. Vaillancourt, "Maximum Likelihood Estimation of the Generalized Dirichlet Mixture". 2002. Online: https://www.academia.edu/24413829/Maximum_likelihood_estimation_of_the_generalized_dirichlet_mixture?source=swp_share. Accessed: 2019-05-30.

[86] J. Vaičiulytė and G. Felinskas, "Paslėptųjų Markovo modelių metodo tyrimas ir taikymas balso įrašams stenografuoti". *Jaunųjų mokslininkų darbai*, vol. 1, pp. 71–78, lapkr. 2016.

[87] R. E. Gruhn, W. Minker, and S. Nakamura, *Automatic Speech Recognition*. In *Statistical Pronunciation Modeling for Non-Native Speech Processing*, pp. 5–17. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[88] A. Saulwick, J. Littlefield, and M. Broughton, "A Spoken Dialogue System for Command and Control". *DSTO-TR-2754*. Online: https://www.dst.defence.gov.au/sites/default/files/publications/documents/DSTO-TR-2754.pdf. 2012.

[89] S. P. Kodgire and D. Bharambe, "Speech recognition systems for voice controlled devices". *International Journal of Latest Trends in Engineering and Technology*, vol. 6, no. 3, pp. 455–459, 2016.

[90] T. Ibiyemi and A. Akintola, "Automatic speech recognition for telephone voice dialling in yorùbá". *International Journal of Engineering*, vol. 1, no. 4, 2012.

[91] C. Chelba, J. Schalkwyk, B. Harb, C. Parada, C. Allauzen, L. Johnson, M. Riley, P. Xu, P. Jyothi, T. Brants, and Others, "Language

Modeling for Automatic Speech Recognition Meets the Web: Google Search by Voice". Online:https://storage.googleapis.com/pub-tools-public-publication-data/pdf/40380.pdf. 2012.

[92] A. B. Bajpei, M. S. Shaikh, and N. S. Ratate, "Voice operated WEB browser". *International Journal of Soft Computing and Artificial Intelligence*, vol. 3, no. 1, pp. 30–32, 2015.

[93] X. Sun, Y. Miyanaga, and B. Sai, "Dynamic Time Warping for Speech Recognition with Training Part to Reduce the Computation". *Journal of Signal Processing*, vol. 18, no. 2, pp. 89–96, 2014.

[94] R. S. Chavan and G. S. Sable, "An overview of speech recognition using HMM". *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 6, pp. 233–238, 2013.

[95] D. Verstraeten, B. Schrauwen, D. Stroobandt, and J. V. Campenhout, "Isolated word recognition with the Liquid State Machine: a case study". *Information Processing Letters*, vol. 95, no. 6, pp. 521–528, 2005.

[96] D. Fohr, O. Mella, and I. Illina, "New Paradigm in Speech Recognition: Deep Neural Networks". In *IEEE International Conference on Information Systems and Economic Intelligence*, (Marrakech, Morocco), 2017.

[97] O. Cappe, "Online EM Algorithm for Hidden Markov Models". In *Journal of Computational and Graphical Statistics* pp. 728-749, 2011.

[98] V. Tungikar and J. Mokashi, "Study of Hidden Markov Model for Isolated Word Recognition". *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering ISO*, vol. 3297, no. 8, pp. 100–103, 2007.

[99] S. V. Vaseghi, *Hidden Markov Models*, ch. 5, pp. 147–172. John Wiley & Sons, Ltd, 2009.

[100] R. G. Leonard and G. Doddington, "[dataset] TIDIGITS LDC93S10". Online: https://catalog.ldc.upenn.edu/LDC93S10 1993.

[101] D. Dua and C. Graff, "[dataset] Spoken Arabic Digits in UCI Machine Learning Repository". Online: http://archive.ics.uci.edu/ml 2017.

[102] L. Yang, K. Ting, and M. B. Srivastava, “Inferring occupancy from opportunistically available sensor data”. In *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 60–68, IEEE, 2014.

[103] T. Ekwevugbe, N. Brown, V. Pakka, and D. Fan, “Real-time building occupancy sensing using neural-network based sensor network”. In *2013 7th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, pp. 114–119, IEEE, 2013.

[104] J. Chaney, E. H. Owens, and A. D. Peacock, “An evidence based approach to determining residential occupancy and its role in demand response management”. *Energy and Buildings*, vol. 125, pp. 254–266, 2016.

[105] M. Milenkovic and O. Amft, “An opportunistic activity-sensing approach to save energy in office buildings”. In *Proceedings of the fourth international conference on Future energy systems*, pp. 247–258, ACM, 2013.

[106] D. Chen, S. Barker, A. Subbaswamy, D. Irwin, and P. Shenoy, “Non-intrusive occupancy monitoring using smart meters”. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, BuildSys’13, (New York, NY, USA), pp. 9:1–9:8, ACM, 2013.

[107] D. Carboni, A. Gluhak, J. McCann, and T. Beach, “Contextualising water use in residential settings: A survey of non-intrusive techniques and approaches”. *Sensors*, vol. 16, p. 738, May 2016.

[108] M. Jin, R. Jia, and C. J. Spanos, “Virtual occupancy sensing: Using smart meters to indicate your presence”. *IEEE Transactions on Mobile Computing*, vol. 16, pp. 3264–3277, Nov 2017.

[109] L. Candanedo Ibarra and V. Feldheim, “Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models”. *Energy and Buildings*, vol. 112, 2015.

[110] R. Lyon, B. Stappers, S. Cooper, J. Brooke, and J. Knowles, “Fifty years of pulsar candidate selection: From simple filters to a new principled real-time classification approach”. *Monthly Notices of the Royal Astronomical Society*, vol. 459, p. stw656, 04 2016.

[111] V. Beskin, S. Chernov, C. Gwinn, and A. Tchekhovskoy, “Radio pulsars”. *Space Science Reviews*, vol. 191, no. 1-4, pp. 207–237, 2015.

[112] A. Cameron, D. Champion, M. Kramer, M. Bailes, E. Barr, C. Bassa, S. Bhandari, N. Bhat, M. Burgay, S. Burke-Spolaor, *et al.*, “The high time resolution universe pulsar survey–xiii. psr j1757- 1854, the most accelerated binary pulsar”. *Monthly Notices of the Royal Astronomical Society: Letters*, vol. 475, no. 1, pp. L57–L61, 2018.

[113] V. Morello, E. D. Barr, M. Bailes, C. M. Flynn, E. F. Keane, and W. van Straten, “SPINN: a straightforward machine learning solution to the pulsar candidate selection problem”. *Mon. Not. Roy. Astron. Soc.*, vol. 443, no. 2, pp. 1651–1662, 2014.

[114] K. Lee, K. Stovall, F. Jenet, J. Martinez, L. Dartez, A. Mata, G. Lunsford, S. Cohen, C. Biwer, M. Rohr, *et al.*, “PEACE: pulsar evaluation algorithm for candidate extraction–a software package for post-analysis processing of pulsar survey candidates”. *Monthly Notices of the Royal Astronomical Society*, vol. 433, no. 1, pp. 688–694, 2013.

[115] R. P. Eatough, N. Molkenthin, M. Kramer, A. Noutsos, M. Keith, B. Stappers, and A. Lyne, “Selection of radio pulsar candidates using Artificial Neural Networks”. *Monthly Notices of the Royal Astronomical Society*, vol. 407, no. 4, pp. 2443–2450, 2010.

[116] S. Bates, M. Bailes, B. Barsdell, N. Bhat, M. Burgay, S. Burke-Spolaor, D. Champion, P. Coster, N. D’Amico, A. Jameson, *et al.*, “The high time resolution universe pulsar survey—vi. an artificial neural network and timing of 75 pulsars”. *Monthly Notices of the Royal Astronomical Society*, vol. 427, no. 2, pp. 1052–1065, 2012.

[117] J. M. Brooke, R. J. Lyon, J. D. Knowles, B. W. Stappers, and S. Cooper, “Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach”. *Monthly Notices of the Royal Astronomical Society*, vol. 459, no. 1, pp. 1104–1123, 2016.

[118] D. M. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation”. *Mach. Learn. Technol.*, vol. 2, 2011.

[119] D. R. Lorimer and M. Kramer, “Handbook of pulsar astronomy”. *Handbook of Pulsar Astronomy, by DR Lorimer, M. Kramer, Cambridge, UK: Cambridge University Press, 2012*, 2012.

[120] W. Zhu, A. Berndsen, E. Madsen, M. Tan, I. Stairs, A. Brazier, P. Lazarus, R. Lynch, P. Scholz, K. Stovall, *et al.*, “Searching for pulsars using image pattern recognition”. *The Astrophysical Journal*, vol. 781, no. 2, p. 117, 2014.

DARBO REZULTATŲ APROBAVIMAS

Pagrindiniai tyrimo rezultatai pristatyti tarptautinėse bei respublikinėse konferencijose.

Pranešimai skaityti šiose respublikinėse konferencijose:

- Konferencija „Kompiuterininkų dienų 2017“. Pranešimas „Rekurentinis paslėptųjų Markovo modelių parametro vertinimo algoritmas“. Lietuva, Kaunas, 2017 m. rugsėjo 21–22 d.
- Respublikinė mokslinė-praktinė konferencija „Informacinių technologijų iššūkiai kūrybos ekonomikoje“. Pranešimas „Paslėptųjų Markovo modelių parametrų rekurentinis vertinimas“. Lietuva, Šiauliai, 2017 m. kovo 17 d.
- Lietuvos Matematikų Draugijos 59-oji konferencija. Pranešimas „Rekurentinis paslėptųjų Markovo modelių parametrų vertinimo algoritmas ir jo taikymai“. Lietuva, Kaunas, 2018 m. birželio 18–19 d.
- Konferencija „Operacijų tyrimas ir taikymai“. Pranešimas „Paslėptųjų Markovo modelių parametrų vertinimo algoritmo tyrimas“. Lietuva, Kaunas, 2016 m. balandžio 8 d.
- Konferencija „Kompiuterininkų dienų 2015“. Pranešimas „Automatinio šnekos atpažinimo metodų tyrimas ir taikymas balso įrašams stenografuoti“. Lietuva, Panevėžys, 2015 m. rugsėjo 17–19 d.

Stendiniai pranešimai pristatyti šiose konferencijose:

- Tarptautinė 8-oji konferencija „Data Analysis Methods for Software Systems“. Stendinis pranešimas „Recurrent estimation of homogeneous Hidden Markov model parameters“. Lietuva, Druskininkai, 2016 m. gruodžio 1–3 d.

Pranešimai skaityti šiose tarptautinėse konferencijose:

- Tarptautinė konferencija EURO 2018 (29th European Conference on Operational Research). Pranešimas „Recurrent parameter estimation algorithm in hidden Markov models with application to multivariate data analysis and signal recognition“. Ispanija, Valensija, 2018 m. liepos 8–11 d.

- Tarptautinė konferencija ICIC (3rd International Conference INNOVATIONS AND CREATIVITY). Pranešimas „Recursive Dirichlet Hidden Markov Model Parameter Estimation Algorithm“. Latvija, Liepoja, 2019 m. birželio 6–8 d.

DARBO REZULTATŲ PUBLIKAVIMAS

Straipsniai recenzuojamuose Lietuvos ir užsienio leidiniuose:

- Vaičiulytė J., Sakalauskas L., 2020. Recursive parameter estimation algorithm of the Dirichlet hidden Markov model. *Journal of Statistical Computation and Simulation*. Taylor & Francis Production. (**ISI Web of Science, JIF=0.767**), vol. 90:2, p. 306–323. ISSN 0094-9655. eISSN 1563-5163. https://doi.org/10.1080/00949655.2019.1679144.

- Vaičiulytė J., Sakalauskas L., 2019. Recursive estimation of multivariate hidden Markov model parameters. *Computational statistics*. Heidelberg, Springer. (**ISI Web of Science, JIF=0.680**), vol. 34, p. 1337–1353. ISSN 0943-4062. eISSN 1613-9658. https://doi.org/10.1007/s00180-019-00877-z.

- Vaičiulytė J., Sakalauskas L., 2017. Rekurentinis paslėptųjų Markovo modelių parametrų vertinimo algoritmas. *Computational Science and Techniques* (**Index Copernicus, CEEOL**). Vol 5, No 1 (2017), p. 529–540, eISSN: 2029-9966, https://doi.org/10.15181/csat.v5i1.1561.

- Vaičiulytė J., Felinskas G., 2016. Paslėptųjų Markovo modelių metodo tyrimas ir taikymas balso įrašams stenografuoti. *Jaunųjų Mokslininkų Darbai* (**Index Copernicus, CEEOL**), 1(45), 71–78. https://doi.org/10.21277/jmd.v1i45.40

Santraukos konferencijų leidiniuose:

- Vaičiulytė, Jūratė. Recurrent estimation of homogeneous Hidden Markov model parameters. *Data analysis methods for software systems : 8th international workshop on data analysis methods for software systems* [abstracts book], Druskininkai, 2016 m. gruodžio 1–3 d. Vilnius: Vilniaus universiteto leidykla, 2016. ISBN 9789986680611. p. 62.

Vilniaus universiteto leidykla
Saulėtekio al. 9, LT-10222 Vilnius
El. p. info@leidykla.vu.lt,
www.leidykla.vu.lt
Tiražas 20 egz.