Liet. matem. rink., T. 42, spec. nr., 235–240

Daugiamačių duomenų vizualizavimas įvertinant savireguliuojančių neuroninių tinklų mokymo eigą

Gintautas DZEMYDA, Olga KURASOVA (MII)
el. paštas: dzemyda@ktl.mii.lt, kurasova@ktl.mii.lt

1. Įvadas

Daugiamačių duomenų vizualizavimas – sudėtinga problema. Vienas iš daugiamačių duomenų vizualizavimo metodų – Sammono algoritmas [1]. Daugiamatę erdvę projektuojant į plokštumą neišvengiamos projekcijos paklaidos. Būtina kurti metodus, minimizuojančius šias paklaidas. Anksčiau atlikti tyrimai [2] parodė, kad savireguliuojančių neuroninių tinklų (SOM) kombinacija su Sammono algoritmu yra efektyvus projektavimo būdas. Čia po neuroninio tinklo apmokymo gauti vektoriai (neuronai-nugalėtojai) analizuojami ir vizualizuojami Sammono algoritmu. Šiame straipsnyje pasiūlyta nauja SOM ir Sammono algoritmo kombinacija, kai daugiamačiai duomenys projektuojami į plokštumą pagal Sammono algoritmą, įvertinant savireguliuojančių neuroninių tinklų mokymo eigą. Parodyta, kad naujuoju algoritmu gaunama mažesnė daugiamačių vektorių projektavimo į dvimatę plokštumą vidutinė paklaida.

2. Daugiamačių duomenų vizualizavimo būdai

Sammono algoritmas. Sammono projekcija [1] yra netiesinio daugelio kintamųjų objektų (vektorių) atvaizdavimo žemesnio matavimo erdvėje metodas. Nagrinėsime atvejį, kai projektinės erdvės, į kurią atvaizduojame, dimensija yra 2, t. y. atvaizduojame į plokštumą.

Tarkime, turime daugiamačius vektorius $x_1, x_2, \ldots, x_m$, priklausančius erdvei R^n. Čia $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $i = \overline{1,m}$. Sprendžiamas uždavinys – šiuos n-mačius vektorius $x_1, x_2, \ldots, x_m$ atvaizduoti (gauti projekciją) plokštumoje R^2. Juos atitiks dvimačiai vektoriai $y_1, y_2, \ldots, y_m \in R^2$. Čia $y_i = (y_{i1}, y_{i2})$, $i = \overline{1,m}$. Pažymėkime d^*_{ij} atstumą tarp daugiamačių vektorių x_i ir x_j, d_{ij} – atstumą tarp vektorius x_i ir x_j atitinkančių dvimačių vektorių y_i ir y_j $(i, j = \overline{1,m})$. Sammono algoritmas minimizuoja projekcijos iškraipymą (paklaidą):

$$E_s = \frac{1}{\sum\limits_{\substack{i,j=1 \\ i<j}}^{n} d^*_{ij}} \sum_{\substack{i,j=1 \\ i<j}}^{n} \frac{(d^*_{ij} - d_{ij})^2}{d^*_{ij}}. \qquad (1)$$

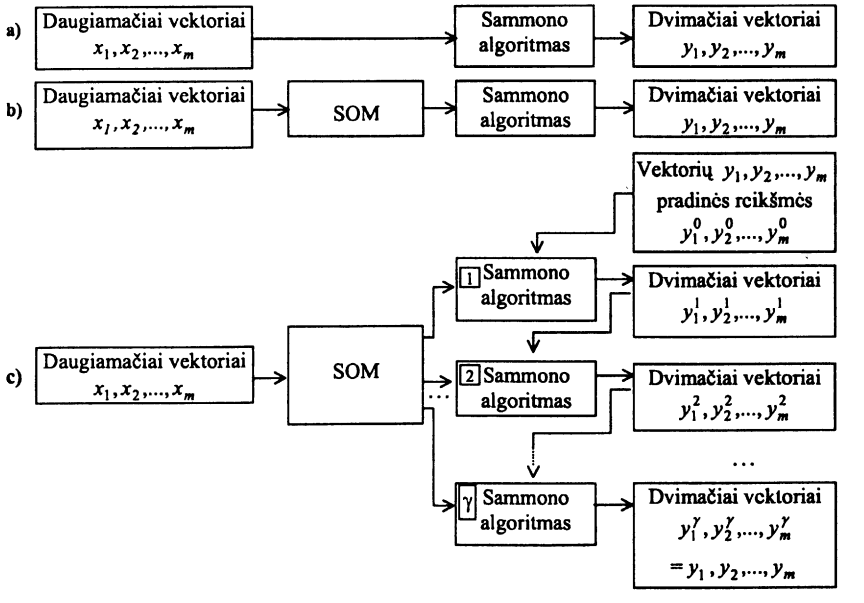
Dvimačių vektorių $y_i \in R^2$ komponentės y_{ik}, $i = \overline{1,m}$, $k = \overline{1,2}$ randamos naudojantis iteracine formule:

$$y_{ik}(m'+1) = y_{ik}(m') - \alpha \frac{\frac{\partial E_s(m')}{\partial y_{ik}(m')}}{\left|\frac{\partial^2 E_s(m')}{\partial y_{ik}^2(m')}\right|}, \tag{2}$$

čia m' yra iteracijos numeris, o α vadinamas „magiškuoju faktoriumi", kadangi nuo jo priklauso projekcijos paklaida.

Savireguliuojantys neuroniniai tinklai. Šiame darbe nagrinėsime savireguliuojančius neuroninius tinklus dar vadinamus Kohoneno neuroniniais tinklais. Kohoneno tinklas [3] yra neuronų masyvas, paprastai išdėstytų dvimačio tinklelio, dar vadinamo žemėlapiu arba lentele, mazguose. Kiekvieną žemėlapio elementą atitinka n-matis vektorius. Stačiakampis tinklelis (žemėlapis) yra sudarytas iš $q \times r$ elementų – n-mačių vektorių: q eilučių ir r stulpelių. Neuroninis tinklas apmokomas, jam daug kartų pateikiant v skirtingų objektų, nusakomų n-mačiais vektoriais. Apmokant tinklą apskaičiuojami žemėlapio vektoriai ir tuos vektorius atitinkančių objektų numeriai, t.y. objektai pasiskirsto tarp žemėlapio elementų. Netuščius lentelės langelius atitinkančius vektorius, t.y. neuronus-nugalėtojus, galima analizuoti Sammono algoritmu.

Sammono algoritmo kombinacijos su savireguliuojančiais neuroniniais tinklais. Atvaizduojant n-mačius vektorius $x_1, x_2, \ldots, x_m$ plokštumoje, naudojant SOM kombinaciją su Sammono algoritmu, galimi trys scenarijai pateikti 1 pav. 1a pav. pavaizduota daugiamačių vektorių vizualizavimo algoritmo struktūra, kai vektoriai $x_1, x_2, \ldots, x_m$ atvaizduojami plokštumoje naudojantis Sammono algoritmu. Vektorius $x_1, x_2, \ldots, x_m$



1 pav. Daugiamačių vektorių projekcijos plokštumoje scenarijai.

plokštumoje atitinka dvimačiai vektoriai $y_1, y_2, \ldots, y_m$. 1b pav. pavaizduoto algoritmo pradžioje n-mačiais vektoriais apmokamas neuroninis tinklas, vėliau netuščius lentelės langelius atitinkantys vektoriai (neuronai)-nugalėtojai analizuojami Sammono algoritmu. Vektorių-nugalėtojų skaičius paprastai būna mažesnis nei m, o tada tarp vektorių $y_1, y_2, \ldots, y_m$ būna ir sutampančių, t. y. kelis vektorius iš $x_1, x_2, \ldots, x_m$ atitiks vienas plokštumos taškas. 1a ir 1b atvejai detaliau aptarti darbe [2]. Kad būtų galima geriau palyginti 1 pav. algoritmus, šiame darbe tyrimuose naudota 1b algoritmo modifikacija (toliau – 1b algoritmas). Trumpai aptarsime. Neuroninio tinklo mokymo rezultate kiekvieną netuščią lentelės langelį atitinka vienas ar keli vektoriai iš $x_1, x_2, \ldots, x_m$ ir vienas n - matis vektorius-nugalėtojas. Sammono algoritmu analizuojami ne vektoriai-nugalėtojai, bet m n-mačių vektorių $z_1, z_2, \ldots, z_m$ sistema, susidedanti iš vektorių-nugalėtojų, kiekvienas kurių yra pakartotas tiek kartų, kiek yra tą vektorių atitinkančių vektorių tarp $x_1, x_2, \ldots, x_m$. Toliau bus išsamiau nagrinėjamas algoritmas, kurio struktūra pateikta 1c pav. Tai naujas savireguliuojančių neuroninių tinklų ir Sammono atvaizdavimo apjungimo algoritmas. Tyrimai parodė, kad būtent tokia SOM ir Sammono algoritmo kombinacijos struktūra tinkamiausia norint rasti tikslesnę daugiamačių vektorių projekciją plokštumoje kriterijaus (1) prasme.

Algoritmo struktūra

- Neuroninį tinklą apmokysime n-mačiais vektoriais $x_1, x_2, \ldots, x_m$ naudodami e epochų. Epocha – tai mokymo proceso dalis, kai visus vektorius atsitiktine tvarka pateikiame tinklui po vieną kartą.
- Prieš neuroninio tinklo apmokymą pasirenkame į kelis blokus γ skaidysime mokymo eigą.
- Po p mokymo epochų ($p = e \bmod \gamma$) gautus vektorius-nugalėtojus analizuojame Sammono algoritmu. Pradinės dvimačių vektorių $y_i^0 = (y_{i1}^0, y_{i2}^0)$, $i = \overline{1,m}$, koordinatės imamos tokios: $y_{i1}^0 = i + \frac{1}{3}$, $y_{i2}^0 = i + \frac{2}{3}$. Sammono algoritmu apskaičiuojame vektorių-nugalėtojų dvimates projekcijas, o tuo pačiu ir n-mačių vektorių $x_1, x_2, \ldots, x_m$ dvimates projekcijas $y_1^1, y_2^1, \ldots, y_m^1$ ($y_i^1 = (y_{i1}^1, y_{i2}^1)$, $i = \overline{1,m}$).
- Neuroninio tinklo mokymas tęsiamas toliau. Po sekančių p epochų gautus vektorius-nugalėtojus vėl analizuojame Sammono algoritmu. Dabar pradines dvimačių vektorių $y_1^2, y_2^2, \ldots, y_m^2$ koordinates imame lygias prieš tai buvusio atvaizdavimo dvimačių vektorių $y_1^1, y_2^1, \ldots, y_m^1$ koordinatėms.
- Po γ tokių žingsnių gauname dvimačius vektorius $y_1^\gamma, y_2^\gamma, \ldots, y_m^\gamma$, kurie atitinka n-mačių vektorių $x_1, x_2, \ldots, x_m$ dvimates projekcijas $y_1, y_2, \ldots, y_m$.

Norint parodyti, kad, kokia bebūtų α reikšmė (2) formulėje, naujuoju algoritmu gaunama vidutiniškai mažesnė paklaida, negu algoritmu 1b, tyrime buvo analizuojami atvejai su įvairiomis α reikšmėmis. Minimali projekcijos paklaida E_s ieškoma gradientiniu metodu (2). Iteracijoms augant, paklaidos reikšmė mažėja, tačiau pasitaiko atvejų, kai kažkurios iteracijos metu paklaida staigiai išauga ir vėl sumažėja [4]. Tokia situacija gali būti ir paskutinės iteracijos metu. Norint neužfiksuoti tokios paklaidos reikšmės kaip galutinio rezultato, imama mažiausia paklaida, apskaičiuota visų iteracijų metu. Projekcijos paklaidos gali skirtis ir esant kitoms tinklo neuronų pradinėms reikšmėms, generuojamoms

atsitiktinai. Vadinasi, atsitiktinai gali gautis daug didesnė arba daug mažesnė paklaida ir naujuoju algoritmu, ir 1b algoritmu. Kad to išvengti, eksperimentai buvo vykdomi 25 kartus, gauti rezultatai suvidurkinti. 25 eksperimentų pilnai pakanka, kad vidurkiai pakliūtų į savo pasikliautinuosius intervalus su pakankamai didele tikimybe. Atliekant eksperimentus pastebėta, kad vykdant 1b algoritmą pasitaikydavo atvejų, kad, iteraciniu būdu skaičiuojant projekcijos koordinates, projekcijos paklaidos E_s antrosios eilės išvestinės tapdavo lygios nuliui. Toliau vykdyti skaičiavimų nebuvo galima, kadangi (2) formulėje antrosios eilės išvestinės yra vardiklyje. Tai atsitikdavo, kai neuroninis tinklas būdavo mažas (2×2), t. y. vieną neuroną-nugalėtoją atitikdavo keli apmokymo aibės vektoriai. Naujajame algoritme šią problemą pavyko nesunkiai išspręsti: jeigu analizuojant kažkurio mokymo bloko rezultatus antroji išvestinė lygi nuliui, tai tas tinklo mokymo blokas turi būti praleidžiamas, o analizuojant sekantį mokymo bloką, pradinės dvimačių vektorių koordinates reikia imti iš prieš tai buvusio „gero" bloko.

3. Tyrimų rezultatai

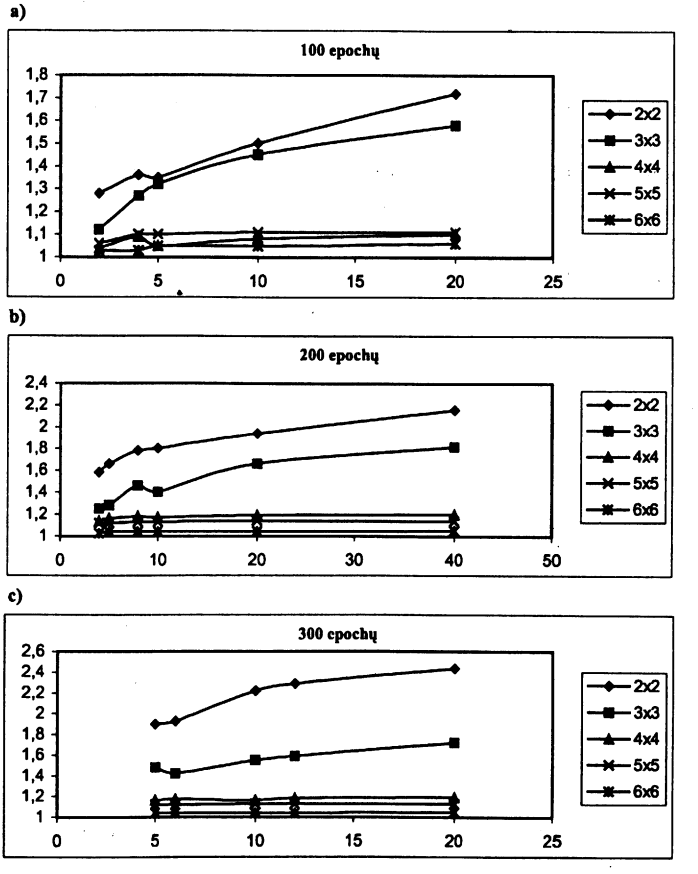
Naujojo algoritmo (1c pav.) pranašumas prieš 1b algoritmą, parodytas analizuojant realius duomenis. Analizuoti ekologiniai duomenys, nusakantys Suomijos pajūrio kopas ir jų vegetaciją [5]. Kopas charakterizuoja šie parametrai: x_1 – atstumas nuo kranto; x_2 – aukštis virš jūros lygio; x_3 – dirvožemio PH; x_4, x_5, x_6, x_7 – kalcio (CA), fosforo (P), kalio (K), magnio (Mg) kiekis; x_8; x_9 – vidutinis smėlio skersmuo ir jo rūšis; x_{10} – šiaurumas pagal suomišką koordinačių sistemą; x_{11} – žemės kilimo greitis; x_{12} – jūros lygio svyravimas; x_{13} – dirvožemio drėgnumas; x_{14} – šlaito tangentas; x_{15} – smėlio paviršiaus dalis; x_{16} – medžiais apaugusi dalis. Darbe [5] yra pateikta šių 16 parametrų koreliacinė matrica. Naudojantis darbe [2] pasiūlytu metodu, gauti 16 objektų-vektorių ($v = 16$), atitinkančių parametrams $x_1 - x_{16}$, sudarytų iš 16 komponenčių ($n = 16$). Visi šie 16-mačiai vektoriai yra vienetinio ilgio.

Analizuoti atvejai, esant įvairiam neuroninio tinklo dydžiui (2×2, 3×3, 4×4, 5×5, 6×6), tinklo mokymo epochų skaičiui (100, 200, 300), Sammono algoritmo „magiškajam faktoriui" α (0, 1; 0, 11; ...; 0, 89; 0, 9). Esant toms pačioms pradinėms sąlygoms, su visais minėtais parametrais apskaičiuotos projekcijos paklaidos ir 1b, ir naujuoju algoritmu. Kaip minėta anksčiau, eksperimentai pakartoti 25 kartus, imant kitas atsitiktines pradines SOM neuronų-vektorių komponenčių reikšmes. Apskaičiuotas santykis tarp suvidurkintų projekcijos paklaidų, gautų 1b ir naujuoju algoritmais. Iš 1 lentelės ir 2 pav. matyti, kad šis santykis visada didesnis už vienetą. Vadinasi naujuoju algoritmu gautos vidutinės paklaidos yra mažesnės. Didinant neuroninio tinklo mokymo blokų γ skaičių (2 pav. abscisių ašis), šis santykis didėja. Tai ypač ryšku, kai analizuojamas mažas tinklas. Taip pat matyti, kad, didėjant tinklo dydžiui, santykis mažėja.

3 pav. parodyta, kad naujuoju algoritmu gauta vidutinė projekcijos paklaida mažiau priklauso nuo α reikšmės negu gautoji 1b algoritmu. Čia imtos suvidurkintos paklaidos, kai atlikti 25 eksperimentai. Pateiktame pavyzdyje neuroninio tinklo 2×2 apmokymui naudojama 200 epochų ir atsižvelgiama į 40 tarpinių mokymo rezultatų. Panašūs rezultatai gauti ir kitais atvejais.

1 lentelė. Santykis tarp projekcijos paklaidos, gautos 1b ir naujuoju algoritmais

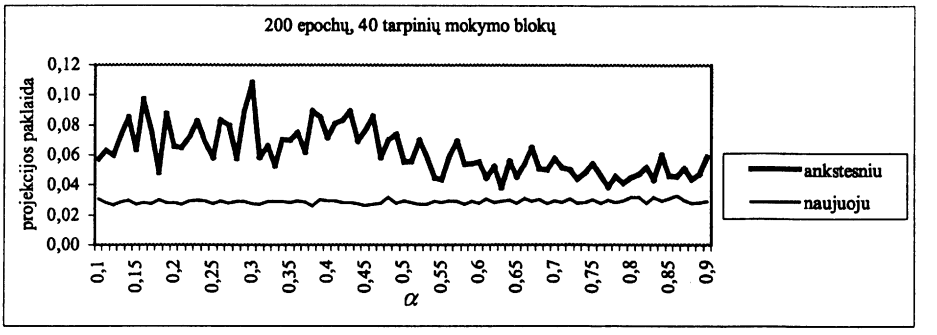
e	100					200						300				
p	50	25	20	10	5	50	40	25	20	10	5	60	50	30	25	15
γ	2	4	5	10	20	4	5	8	10	20	40	5	6	10	12	20
2×2	1,28	1,36	1,35	1,5	1,72	1,58	1,66	1,78	1,8	1,94	2,16	1,9	1,93	2,22	2,29	2,44
3×3	1,12	1,27	1,32	1,45	1,58	1,25	1,28	1,46	1,4	1,66	1,82	1,48	1,42	1,55	1,59	1,72
4×4	1,04	1,09	1,05	1,08	1,1	1,14	1,16	1,18	1,17	1,19	1,2	1,16	1,17	1,16	1,18	1,19
5×5	1,06	1,1	1,1	1,11	1,11	1,12	1,12	1,13	1,13	1,14	1,14	1,12	1,12	1,13	1,13	1,13
6×6	1,03	1,03	1,05	1,05	1,06	1,03	1,04	1,04	1,04	1,04	1,05	1,04	1,04	1,04	1,04	1,05



2 pav. Projekcijos paklaidų santykis, kai SOM epochų skaičius yra: a) 100, b) 200, c) 300.

4. Išvados

Lyginant vidutines projekcijos paklaidas, gautas savireguliuojančių neuroninių tinklų ir Sammono atvaizdavimo kombinacija 1b, su šiame straipsnyje pasiūlytu nauju kombinavimo algoritmu, kai atsižvelgiama į neuroninio tinklo mokymosi eigą, matyti, kad nau-



3 pav. Paklaidos priklausomybė nuo α reikšmės.

juoju algoritmu gautos mažesnės projekcijos paklaidos, t. y. gaunama tikslesnė projekcija. Kuo daugiau vertinama tarpinių neuroninio tinklo rezultatų, tuo pagerėjimas didėja. Gal būt analizuojant po kiekvienos neuroninio tinklo epochos gautus vektorius-nugalėtojus gautume dar geresnius rezultatus. Bet čia susiduriame su labai išaugančiu skaičiavimų laiku. Naujuoju algoritmu pavykdavo išvengti nulinių antros eilės išvestinių, kurios naudojamos dvimačių vektorių koordinačių skaičiavimui. Be to, naujuoju algoritmu gautų projekcijos paklaidų priklausomybė nuo α reikšmės yra mažesnė negu 1b algoritmu.

Literatūra

[1] J.W. Sammon, A nonlinear mapping for data structure analysis, *IEEE Transactions on Computers*, **C–18**, 401–409 (1969).

[2] G. Dzemyda, Visualization of a set of parameters characterized by their correlation matrix, *Computational Statistics and Data Analysis*, **36**(1), 15–30 (2001).

[3] T. Kohonen, *Self-Organizing Maps*, 3rd ed., Springer Series in Information Sciences, **30**, Springer–Verlag (2001).

[4] I. Apostal, W. Szpankowski, Indexing and mapping of proteins by Sammon's projection algorithm, *Journal of Computational Chemistry*, **20**, 1049–1059 (1999).

[5] P. Hellemaa. *The Development of Coastal Dunes and their Vegetation in Finland*, Dissertation, Fenia 176:1, Helsinki (1998), `http://ethesis.helsinki.fi/julkaisut/mat/maant/vk/hellemaa/index.html`

Visualization of multidimensional data taking into account the learning flow of the self organizing neural network

G. Dzemyda, O. Kurasova

In the paper we discuss the visualization of multidimensional vectors taking into account the learning flow of the self organizing neural network. A new algorithm realizing a combination of the self-organizing map (SOM) and Sammon's mapping has been proposed. It takes into account the intermediate learning results of the SOM. The experiments showed that the algorithm gives lower average projection errors compared with a consequent application of the SOM and Sammon's mapping.