DUOMENŲ ANALIZĖ IR VAIZDAVIMAS

Vektorių kvantavimo metodų ir daugiamačių skalių junginys daugiamačiams duomenims vizualizuoti

Alma Molytė

Matematikos ir informatikos instituto doktorantė
Institute of Mathematics and Informatics, PhD student
Akademijos g. 4, LT-08663 Vilnius
Tel. (8 5) 210 93 22, faks. (8 5) 272 92 09
El. paštas: Alma.Molyte@gmail.com

Olga Kurasova

Matematikos ir informatikos instituto vyresnioji mokslo darbuotoja, daktarė
Institute of Mathematics and Informatics, Senior researcher, PhD
Akademijos g. 4, LT-08663 Vilnius
el. (8 5) 210 93 22, faks. (8 5) 272 92 09
El. paštas: Kurasova@ktl.mii.lt

Darbe pateikiama lyginamoji dviejų vektorių kvantavimo metodų (saviorganizuojančių neuroninių tinklų ir neuroninių dujų) analizė. Neuronai nugalėtojai, kurie gaunami vektorių kvantavimo metodais, yra vizualizuojami daugiamačių skalių metodu. Tirta kvantavimo paklaidos priklausomybė nuo vektorių nugalėtojų skaičiaus. Išsiaiškinta, kuris vektorių kvantavimo metodas yra tinkamesnis jungti su daugiamačių skalių metodu, t. y. vizualizavus neuronus nugalėtojus „atskleidžiama“ analizuojamų duomenų struktūra.

Įvadas

Bet kokia analizuojamų objektų aibė dažnai charakterizuojama bendrais parametrais. Tam tikras visų parametrų reikšmių rinkinys nusako vieną konkretų analizuojamos aibės $X = \{X_1, X_2, ..., X_m\}$ objektą $X_i = (x_{i1}, x_{i2}, ..., x_{in})$; čia m yra analizuojamų objektų skaičius, n – parametrų skaičius ir i – objekto eilės numeris. Taigi, $X_1, X_2, ..., X_m$ yra n-mačiai vektoriai. Dažnai jie interpretuojami kaip taškai n-matėje erdvėje R^n; čia n – erdvės matmenų skaičius. Iš tikrųjų turime analizuojamų duomenų aibės matricą $X = \{X_1, X_2, ..., X_m\} = \{x_{ij}, i = 1, ..., m, j = 1, ..., n\}$, jos eilutės yra vektoriai $X_i = (x_{i1}, x_{i2}, ..., x_{in})$, $i = 1, ..., m$, čia x_{ij} yra i-tojo vektoriaus j-oji komponentė. Duomenų matrica gali būti analizuojama įvairiais statistikos metodais, tačiau kai duomenų kiekis yra didelis, dažnai jų nepakanka. Siekiant gauti daugiau žinių iš analizuojamų duomenų, taikomi įvairūs duomenų tyrybos (angl. *data mining*) metodai: klasifikavimo, klasterizavimo, vizualizavimo ir kt. Šiame darbe tiriamas daugiamačių duomenų klasterizavimas ir vizualizavimas. Klasterizavimas – tai analizuojamų objektų suskirstymas į skirtingas grupes (klasterius) taip, kad grupės viduje esantys objektai būtų panašūs tarpusavyje, o objektai iš skirtingų grupių būtų nepanašūs. Daugiamačių duomenų vizualizavimo, kitaip dar vadinamais matmenų skaičiaus mažinimo, metodais didelio skaičiaus matmenų duomenys transformuojami į mažesnio skaičiaus matmenų erdvę taip, kad išliktų esamos arba būtų atrastos „užslėptos“ analizuojamų duomenų savybės. Klasterizavimo ir vizualizavimo metodų jungimas leidžia gauti daugiau žinių iš analizuojamų duomenų, negu juos taikant atskirai.

1. Vektorių kvantavimas ir vizualizavimas

Vektorių kvantavimas yra klasikinis signalų (vektorių) aproksimavimo metodas. Aproksimuoti X_l, $l=(1,...,m)$ reiškia rasti vektorių $M_i \in R^n$, $i=1,...,N$, aibę tokią, kad N būtų mažiau už m. Aibė turi būti tokia, kad kiekvieną X_l atitiktų vektorius M_i, $i \in (1,...,N)$, iki kurio atstumas būtų kuo mažesnis. Vektoriai $M_i \in R^n$, $i=1,...,N$, yra vadinami kvantuotais vektoriais.

Vektorių kvantavimas yra naudojamas duomenims suspausti, trūkstamiems duomenims koreguoti, taip pat duomenims klasterizuoti. Saviorganizuojantis neuroninis tinklas (angl. *self-organizing map*) (Kohonen, 2001), neuroninių dujų metodas (angl. *neural gas*) (Martinetz, Schulten, 1991), vektorių kvantavimo mokymas (angl. *learning vector quantization*) (Kohonen, 2001) yra neuroniniais tinklais grindžiami vektorių kvantavimo metodai.

Saviorganizuojantis neuroninis tinklas (SOM) yra mokomas mokymo be mokytojo būdu. SOM tinklas gali ne tik vizualizuoti daugiamačius duomenis, bet sykiu juos ir klasterizuoti, be to, nebūtina iš anksto žinoti klasterių skaičių. SOM tinklai gali būti naudojami siekiant vizualiai pateikti duomenų klasterius ir ieškant daugiamačių duomenų projekcijų į mažesnio skaičiaus matmenų erdvę. Neuroninės dujos (ND) – tai vienas iš kvantavimo metodų, pagrįstas neuroninio tinklo mokymu (Martinetz, Schulten, 1991). Sakykime, turime duomenų vektorių $X_1, X_2, ..., X_m$ rinkinį. Kvantavimo metodų tikslas – rasti vektorių $M_1, M_2, ..., M_N$ $(N < m)$ rinkinį tokį, kad gauti kvantuoti vektoriai M_i, $i=1,...,N$, atspindėtų vektorių X_l, $l=1,...,m$, savybes.

Siekiant gauti daugiau žinių iš analizuojamų duomenų, kvantuotus vektorius tikslinga vizualizuoti. Yra sukurta nemažai daugiamačių duomenų vizualizavimo metodų, padedančių nustatyti arba įvertinti daugiamačių duomenų struktūrą: susidariusius klasterius, itin išsiskiriančius objektus (taškus atsiskyrėlius), analizuojamų objektų ar jų grupių panašumus ir pan. (Dzemyda ir kt., 2008; Chen ir kt., 2008). Kvantuotiems vektoriams vizualizuoti šiame straipsnyje taikytas vienas populiariausių vizualizavimo metodų – daugiamačių skalių metodas (Borg, Groenen, 2005).

1.1. Saviorganizuojantis neuroninis tinklas ir neuroninių dujų metodas

Tiek neuroninių dujų metodu (ND), tiek saviorganizuojančiu neuroniniu tinklu (SOM) yra sukuriamas kvantuotų vektorių masyvas M. Dažnai kvantuoti vektoriai yra vadinami tiesiog neuronais. ND metodu sukuriamas neuronų tinklas yra vienmatis $M=\{M_1, M_2, ..., M_N\}$; čia $M_i \in R^n$ $(M_i=(m_{i1}, m_{i2}, ..., m_{in}))$, $i=1,...,N$, N – neuronų skaičius. SOM tinklas yra dvimatis $M=\{M_{ij}, i=1,\ldots,r, j=1,\ldots,s\}$; čia $M_{ij} \in R^n$ $(M_{ij}=(m_{ij}^1, m_{ij}^2, ..., m_{ij}^n))$, r yra eilučių skaičius, s – stulpelių, neuronų skaičius $N=r \times s$. Metodų tikslas – pakeisti neuronų reikšmes taip, kad jie atspindėtų analizuojamos duomenų aibės vektorių X_l, $l=1,...,m$, savybes, t. y. mokymo pabaigoje neuronai taptų vektorių X_l kvantuotais vektoriais. Bendra algoritmo schema yra tokia: prieš tinklo mokymą generuojamos atsitiktinės pradinės neuronų komponenčių reikšmės intervale $(-0,5 \cdot 10^{-5}, 0,5 \cdot 10^{-5})$ (ND) arba $(0, 1)$ (SOM). Mokymo metu vienas po kito mokymo aibės X vektoriai pateikiami į tinklą nustatytą kiekį kartų. Kiekvienas vektorius į tinklą pateikiamas $\hat{e}$ kartų. Analizuojamų vektorių skaičius yra lygus m, todėl mokymo iteracijų skaičius $t_{\max} = \hat{e} \times m$.

Kvantuojant vektorius ND metodu kiekvienos mokymo iteracijos metu į tinklą pateikiamas vis kitas vektorius X_l, $l \in \{1,...,m\}$, skaičiuojamas Euklido atstumas nuo jo iki visų tinklo neuronų M_i, $i=1,...,N$. Gauti atstumai $\|M_1 - X_l\|, ..., \|M_N - X_l\|$ surūšiuojami didėjimo tvarka. Gaunama neuronų seka $W_1, W_2, ..., W_N$, čia $W_k \in \{M_1, M_2, ..., M_N\}$, $k=1,...,N$, tokia, kad $\|W_1 - X_l\| \leq, ..., \leq \|W_N - X_l\|$. Tada atstumas nuo X_l iki pirmo neurono W_1 yra mažiausias. Šis neuronas vadinamas neuronu nugalėtoju. Visų neuronų reikšmės keičiamos pagal formulę: $W_k(t+1) = W_k(t) + E(t) \cdot h_\lambda \cdot (X_l - W_k(t))$; čia t yra iteracijos numeris $(t=0,...,t_{\max})$,

ND mokymo algoritmas:

FOR $t=0$ TO $t_{\max}$

  FOR $l=1$ TO m

    FOR $i=1$ TO N

      $\|M_i - X_l\| = \sqrt{\sum_p^n (m_{ip} - x_{lp})^2}$ // *skaičiuojamas Euklido atstumas*

    END

    $\{W_1, W_2, ..., W_N\}$ = SORT_ASCENDING($\|M_1 - X_l\|, ..., \|M_N - X_l\|$)

    // *čia* $W_k \in \{M_1, M_2, ..., M_N\}$, $k = 1, ..., N$, *ir* $\|W_1 - X_l\| \le, ..., \le \|W_N - X_l\|$

    $E(t) = E_i (E_f / E_i)^{(t/t_{\max})}$

    $\lambda(t) = \lambda_i (\lambda_f / \lambda_i)^{(t/t_{\max})}$

    FOR $k=1$ TO N

      $h_\lambda = e^{-(k-1)/\lambda(t)}$

      $W_k(t+1) = W_k(t) + E(t) \cdot h_\lambda \cdot (X_l - W_k(t))$ // *ND mokymo taisyklė*

    END

  END

END

SOM mokymo algoritmas

FOR $t=1$ TO $\hat{e}$

  FOR $l=1$ TO m

    FOR $i=1$ TO r

      FOR $j=1$ TO s

        $\|M_{ij} - X_l\| = \sqrt{\sum_p^n (m_{ij}^p - x_{lp})^2}$ // *skaičiuojamas Euklido atstumas*

      END

    END

    $c = \arg\min_{i,j} \{\| X_l - M_{ij} \|\}$, $\hat{M}_c$ *– vektoriaus* X_l *neuronas nugalėtojas*

    FOR $i=1$ TO r

      FOR $j=1$ TO s

        $M_{ij}(t+1) = M_{ij}(t) + h_{ij}^c(t)(X_l - M_{ij}(t))$ // *SOM mokymo taisyklė*

      END

    END

  END

END

čia t yra iteracijos numeris, h_{ij}^c – vadinamoji kaimynystės funkcija, kurios reikšmė priklauso nuo vykdomos iteracijos numerio t ir perskaičiuojamo neurono vietos tinkle neurono nugalėtojo atžvilgiu. Procesui konverguoti būtina, kad $h_{ij}^c(t) \to 0$, kai $t \to \infty$.

Kai tinklas išmokytas, būtina įvertinti jo kokybę. Nadojant vektorių kvantavimo metodus dažniausiai vertinama kvantavimo paklaida E_{QE}, kuri apskaičiuojama pagal formulę:

$$E_{QE} = \frac{1}{m} \sum_{l=1}^{m} \left\| X_l - \hat{M}_c \right\|; \quad (1)$$

čia $\hat{M}_c$ yra vektoriaus X_l neuronas nugalėtojas, ND metode $\hat{M}_c = W_1$.

1.2. Daugiamačių skalių metodas

Naudojantis daugiamačių skalių (angl. *multidimensional scaling*, MDS) metodu, ieškoma daugiamačių duomenų projekcijų mažesnio skaičiaus matmenų erdvėje (dažniausiai R^2, R^3), siekiant išlaikyti analizuojamos aibės objektų artimumus – panašumus arba skirtingumus (Borg, Groenen, 2005). Gautuose vaizduose panašūs objektai išdėstomi arčiau vieni kitų, o skirtingi – toliau. Vienas daugiamačių skalių metodų tikslų yra rasti optimalų daugiamačius objektus atitinkančių taškų (vektorių) vaizdą mažo skaičiaus matmenų erdvėje.

Tarkime, kiekvieną n-matį vektorių $X_i \in R^n$, $i \in \{1, ..., m\}$, atitinka mažesnio skaičiaus matmenų vektorius $Y_i \in R^p$, $p < n$. Atstumą tarp vektorių X_i ir X_j pažymėkime $d(X_i, X_j)$, o atstumą tarp vektorių Y_i ir Y_j – $d(Y_i, Y_j)$, $i, j = 1, ..., m$. Naudojantis MDS algoritmu, bandoma atstumus $d(Y_i, Y_j)$ priartinti prie atstumų

$E(t) = E_i (E_f / E_i)^{(t/t_{\max})}$, $h_\lambda = e^{-(k-1)/\lambda(t)}$, $\lambda(t) = \lambda_i (\lambda_f / \lambda_i)^{(t/t_{\max})}$, parametrų λ_i, λ_f, E_i, E_f reikšmės parenkamos prieš tinklo mokymą.

Jei taikomas SOM metodas, į tinklą pateikus vektorių X_l, skaičiuojamas Euklido atstumas nuo jo iki visų tinklo neuronų M_{ij}, $i = 1, \ldots, r$, $j = 1, \ldots, s$, randamas neuronas nugalėtojas $\hat{M}_c$, iki kurio atstumas nuo X_l yra mažiausias; čia $c = \arg\min_{i,j} \{\| X_l - M_{ij} \|\}$. Neuronų reikšmės keičiamos pagal formulę:

$$M_{ij}(t+1) = M_{ij}(t) + h_{ij}^c(t)(X_l - M_{ij}(t));$$

$d(X_i, X_j)$. Jei naudojama kvadratinė paklaidos funkcija, tai minimizuojama tikslo funkcija E_{MDS} gali būti užrašyta taip:

$$E_{MDS} = \sum_{i<j} \left(d\left(X_i, X_j\right) - d\left(Y_i, Y_j\right) \right)^2 .$$

Šiame darbe naudojamas vienas populiariausių MDS minimizavimo algoritmų – SMACOF (angl. *Scaling by Majorizing a Complicated Function*) įtempimo funkcijai E_{MDS} minimizuoti.

1.3. Kvantavimo metodų jungimas su daugiamačių skalių metodu

Neuroninių dujų ir saviorganizuojantys neuroniniai tinklai mokomi daug kartų jiems pateikiant skirtingų objektų $X_1, X_2, ..., X_m$, nusakomų *n*-mačiais vektoriais. Kiekvienas įėjimo vektorius yra susijęs su artimiausiu neuronu. Dalis neuronų gali būti susiję su kai kuriais analizuojamais įėjimo vektoriais, o dalis ne. Neuronai, kurie yra susiję su analizuojamais įėjimo vektoriais, vadinami neuronais nugalėtojais. Dažniausiai neuronų nugalėtojų skaičius *r* yra mažesnis nei neuronų skaičius *N* ($r \le N$). SOM stačiakampės tinklo topologijos atveju galima nubraižyti lentelę, kurios langeliai atitinka neuronus, tačiau iš jos neaišku, kaip arti kaimyniniuose langeliuose esantys vektoriai yra *n*-matėje erdvėje. Kartais gautus rezultatus sudėtinga interpretuoti, todėl kilo mintis juos analizuoti vienu iš daugiamačių duomenų projekcijos metodu. Tuo tikslu gali būti naudojamas daugiamačių skalių (MDS) metodas. Keletas SOM ir MDS junginių yra analizuojami J. Bernatavičienės ir kt. (2006) darbe. Neuronai nugalėtojai, kurie gaunami neuroninių dujų metodu, taip pat gali būti vizualizuojami daugiamačių skalių metodu (Kurasova, Molytė, 2009).

Šiame darbe analizuojamas neuroninių dujų ir saviorganizuojančio neuroninio tinklo jungimas su daugiamačių skalių metodu. Projekcijos paklaida minimizuojama SMACOF algoritmu. Neuronų nugalėtojų vizualizavimo daugiamačių skalių metodu schema pateikiama 1 paveiksle.

2. Eksperimentinio tyrimo rezultatai

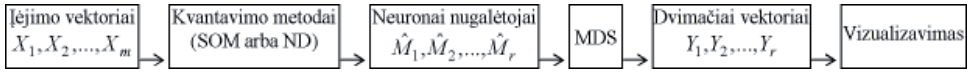
Tyrimuose naudotos kelios duomenų aibės, turinčios tam tikrų specifinių savybių (1 lentelė). Dvi pirmosios duomenų aibės paimtos iš (Fundamental clustering...), o kitų aibių vektorių koordinatės generuotos intervale (0, 1) taip, kad kiekvienos aibės vektoriai sudaro penkis klasterius. Aibės rand_clust* viena nuo kitos skiriasi arba matmenų skaičiumi, arba klasterių artimumu.

Neuronų skaičiaus parinkimo strategija pasiūlyta ir ištirta O. Kurasovos ir A. Molytės (2008) darbe. Šiame tyrime nagrinėjama neuronų nugalėtojų skaičiaus įtaka vizualizavimo rezultatams.

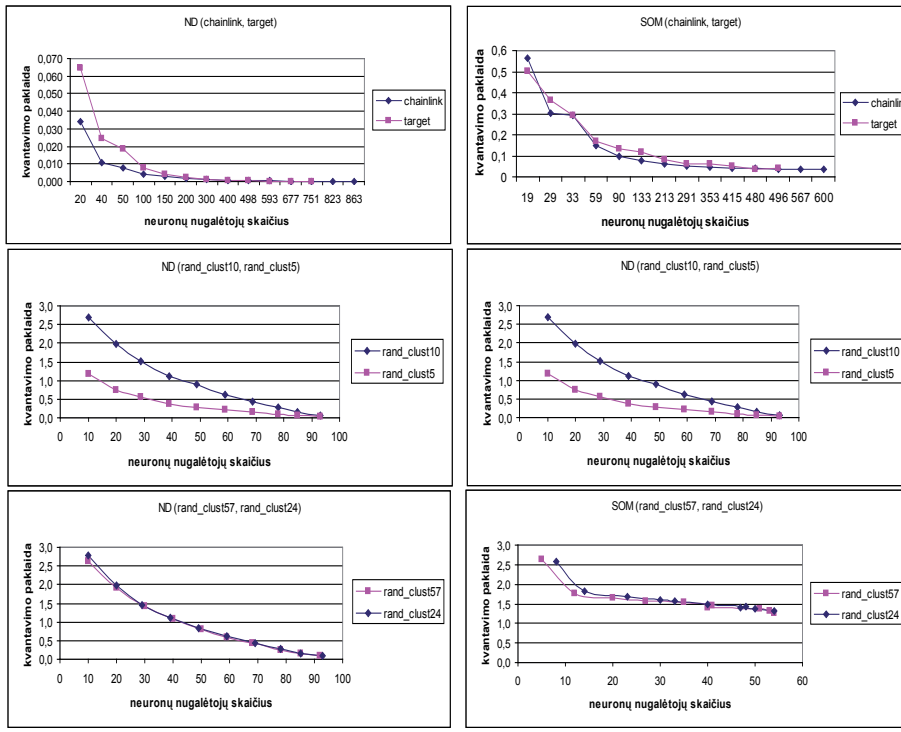
L e n t e l ė . ***Duomenų aibės***

Nr.	Pavadinimas	*m*	*n*	Klasių skaičius	Apibūdinimas
1	chainlink	1000	3	2	tiesiškai neatskiriamos
2	target	770	2	6	taškai atsiskyrėliai
3	rand_clust5	100	5	5	
4	rand_clust10	100	10	5	
5	rand_clust57	100	10	5	tolimesni klasteriai
6	rand_clust24	100	10	5	artimesni klasteriai

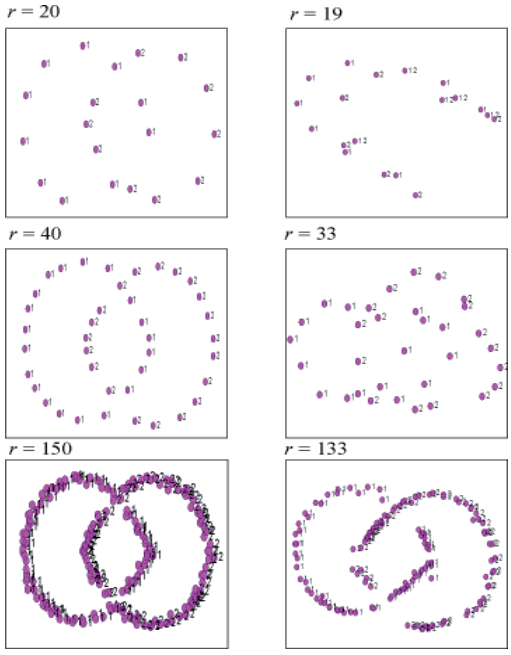
Norint įvertinti kvantavimo kokybę yra skaičiuojama kvantavimo paklaida E_{QE} (1). Kuo neuronų nugalėtojų yra daugiau, tuo kvantavimo paklaida yra mažesnė (2 pav.). Iš 2 paveikslo matome, kad neuroninių dujų metodu gauta kvantavimo paklaida yra mažesnė negu gauta saviorganizuojančiu neuroniniu tinklu, kai neuronų nugalėtojų skaičius mažai skiriasi. Galime daryti išvadą, kad neuroninių dujų metodas yra tinkamesnis vektoriams kvantuoti negu saviorganizuojantys neuroniniai tinklai.



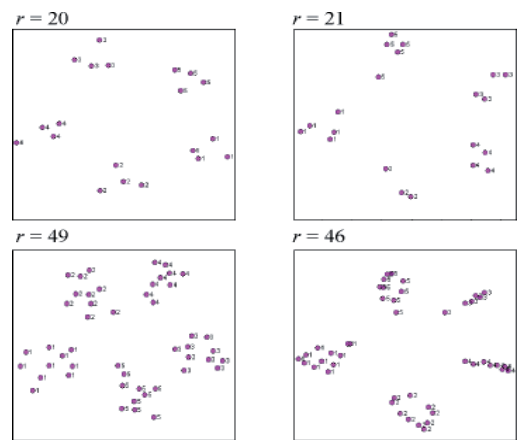
1 p a v . ***Neuronų nugalėtojų vizualizavimo schema***



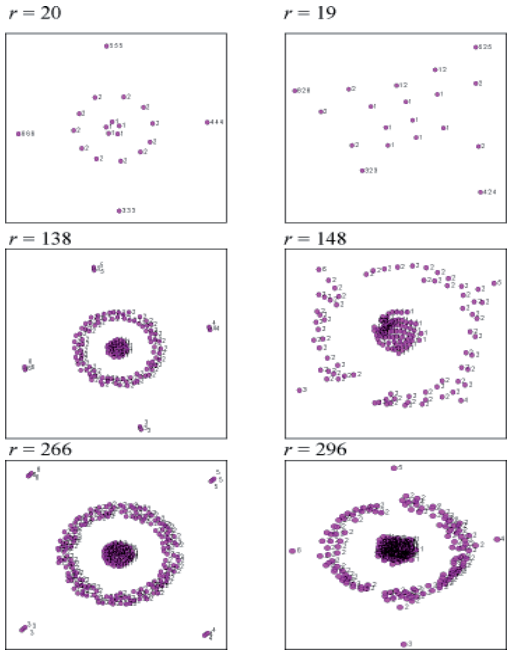
2 p a v. ***Kvantavimo paklaida: ND (kairėje) ir SOM (dešinėje)***



3 p a v. ***Duomenų „chainlink“ vaizdai: ND (kairėje) ir SOM (dešinėje)***



4 p a v. ***Duomenų „rand_clust10“ vaizdai: ND (kairėje) ir SOM (dešinėje)***



5 pav. ***Duomenų „target“ vaizdai: ND (kairėje) ir SOM (dešinėje)***

Neuronai nugalėtojai, kurie yra daugiamačiai vektoriai $\hat{M}_1, \hat{M}_2, ..., \hat{M}_r$, atvaizduojami į dvimačius vektorius $Y_1, Y_2, ..., Y_r$ naudojant daugiamačių skalių metodą. Neuronų nugalėtojų skaičius yra *r*. Gauti dvimačiai vektoriai pateikiami 3–5 paveiksluose. Skaičiai prie taškų reiškia klasės, kuriai jie priklauso, numerį. Iš 3–5 paveikslų matome, kaip keičiasi vaizdai didėjant neuronų nugalėtojų skaičiui. ND metodu duomenų struktūra atskleidžiama, kai neuronų nugalėtojų skaičius *r* yra gana mažas ($r = 20$). SOM metodu duomenų struktūra matoma tik kai neuronų nugalėtojų skaičius yra didelis ($r = 133; 296; 46$).

Išvados

Straipsnyje nagrinėti keli duomenų tyrybos metodai. Neuroninių dujų metodas ir saviorganizuojantys neuroniniai tinklai naudojami siekiant sumažinti analizuojamos aibės duomenų skaičių. Neuronai nugalėtojai vizualizuojami daugiamačių skalių metodu. Tirta kvantavimo paklaida ir daugiamačių duomenų vizualizavimo kokybė.

Eksperimentai parodė, kad kvantavimo paklaida mažėja, kai neuronų nugalėtojų skaičius didėja. Kvantavimo paklaida ND metodu yra mažesnė negu SOM, kai neuronų nugalėtojų skaičius yra beveik lygus. Tai rodo, kad vektoriams kvantuoti neuroninių dujų metodas yra tinkamesnis.

ND metodu gautų vizualizuojamų duomenų struktūra gerai matoma, kai neuronų nugalėtojų skaičius *r* yra gana mažas, o SOM duomenų struktūra – tik kai neuronų nugalėtojų skaičius yra didelis.

LITERATŪRA

BERNATAVIČIENĖ, J.; DZEMYDA, G.; KURASOVA, O.; MARCINKEVIČIUS, V. (2006). Optimal Decisions in Combining the SOM with Nonlinear Projection Methods. *European Journal of Operational Research*, vol. 173, p. 729–745.

BORG, I.; GROENEN, P. (2005). *Modern Multidimensional Scaling*. New York: Springer-Verlag. 616 p. ISBN 0387251502.

CHEN, CH. H.; HRDLE, W.; UNWIN, A. (2008). *Handbook of Data Visualization* (Springer Handbooks of Computational Statistics), CA, USA: Springer-Verlag, 938 p. ISBN 9783540330363.

DZEMYDA, G.; KURASOVA, O.; ŽILINSKAS, J. (2008). *Daugiamačių duomenų vizualizavimo metodai*. Vilnius: Mokslo aidai. 206 p. ISBN 9789986680420.

Fundamental Clustering Problems Suite (FCPS) [interaktyvus] [žiūrėta 2009 m. gegužės 19 d.]. Prieiga per internetą: <http://www.uni-marburg.de/fb12/datenbionik/data/>.

KOHONEN, T. (2001). *Self-organizing Maps*. 3rd ed. Springer series in information sciences. Berlin: Springer-Verlag. 506 p. ISBN 3540679219.

KURASOVA, O.; MOLYTĖ, A. (2008). Neuronų skaičiaus parinkimas vektorių kvantavimo metoduose. *Lietuvos matematikos rinkinys, LMD darbai*, t. 48/49, p. 354–359. ISSN 01322818.

KURASOVA, O.; MOLYTĖ, A. (2009). Investigation of the Quality of Mapping Vectors Obtained by Quantization Methods. Iš: *Proceedings of XIII International Conference on Applied Stochastic Models and Data Analysis*, ASMDA 2009. Vilnius: Technika, p. 269–273. ISBN 9789955284635.

MARTINETZ, T. M.; SCHULTEN, K. J. (1991). A Neural-Gas Network Learns Topologies. Iš: *Artificial Neural Networks*. T. Kohonen, K. Mäkisara, O. Simula, and J. Kangas (Eds.). Amsterdam: North-Holland, p. 397–402.

COMBINATION OF VECTOR QUANTIZATION AND MULTIDIMENSIONAL SCALING

Alma Molytė, Olga Kurasova

S u m m a r y

In this paper, we present a comparative analysis of a combination of two vector quantization methods (self-organizing map (SOM) and neural gas (NG)), based on neural networks and multidimensional scaling that is used for visualization of codebook vectors obtained by vector quantization methods. The dependence of neuron-winners, quantization and mapping qualities, and preserving of a data structure in the mapping image are investigated. It is established that the quantization errors of NG are smaller than that of the SOM when the number of neurons-winners is approximately equal. It means that the neural gas is more suitable for vector quantization. The data structure is visible in the mapping image even when the number r of neurons-winners of NG is small enough. If the number r of neurons-winners of the SOM is larger, the data structure is visible, as well.