

VILNIAUS UNIVERSITETSS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
PROGRAMŲ SISTEMŲ BAKALAURO STUDIJŲ PROGRAMA

Saviorganizuojantys neuroniniai tinklai

Self organizing neural networks

Laboratorinio darbo ataskaita

Atliko: Armintas Pakenis

Darbo vadovas: prof. dr. Olga Kurasova

Vilnius – 2023

TURINYS

1. UŽDUOTIES TIKSLAS	2
1.1. Duomenys.....	2
1.2. Įranga.....	2
2. METODAI	3
2.1. Naudotos funkcijos	3
2.2. Mokymas.....	3
3. REZULTATAI	4
4. IŠVADOS	6

1. Užduoties tikslas

Užduoties tikslas — suprogramuoti saviorganizuojančio neuroninio tinklo (žemėlapių, SOM) mokymo algoritmą, apmokyti jį naudojant pasirinktus duomenis. Naudotos programos kodą galima rasti GitHub repositorijoje: <https://github.com/ArmintasP/Computational-intelligence/tree/main/Lab5>.

1.1. Duomenys

Naudotas irisų duomenų rinkinys <https://archive.ics.uci.edu/ml/datasets/iris>.

Irisų duomenų rinkinyje buvo 150 pavyzdžių, iš kurių po 50 įrašų kiekvienos klasės (iš viso 3 klasės). Požymių skaičius — 4.

1.2. Įranga

Modelio treniravimas vyko ant asmeninio kompiuterio su AMD Ryzen 5 5600G APU (CPU) ir 32 GB RAM.

Taip pat ilgesniems skaičiavimas buvo naudoti MIF superkompiuterio resursai.

2. Metodai

Saviorganizuojančio neuroninio žemėlapiu mokymas buvo atliekamas dvejais skirtingais mokymo būdais, tačiau naudojant tas pačias pagrindines skaičiavimo funkcijas.

2.1. Naudotos funkcijos

Atstumams apskaičiuoti buvo naudojama Euklido atstumų funkcija:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2},$$

kur n — požymių skaičius, x — duomenų rinkinio pavyzdžio vektorius, o y — žemėlapiu neurono svorių vektorius.

Kaimynų atstumui įvertinti naudota Gauso kaimynų atsutimų funkcija:

$$h_{ij}^c = \alpha(t, T) \cdot \exp\left(\frac{-d(R_c - R_{ij}^2)}{2 \cdot (n_{ij}^c(t))^2}\right),$$

kur c — nugalėtojo neurono indeksas, t — iteracijos (epochos) numeris, T — iteracijų (epochų) skaičius, $n_{ij}^c(t)$ — kaimynystės eilės numeris nugalėtojo atžvilgiu, R_c, R_{ij} — nugalėtojo neurono ir lyginimamo neurono indeksų vektoriai, čia $\alpha(t, T) = 1 - \frac{t}{T}$.

2.2. Mokymas

Mokymas buvo atliktas 2 būdais. Pirmu būdu atliekant mokymą duomenų rinkinys nebuvo permaišomas. Per vieną epochą kiekvienam pavyzdžiui X_i buvo randamas neuronas nugalėtojas ir atitinkamai pakeičiamos žemėlapiu neuronų reikšmės.

Antruoju būdu mokymas buvo atliekamas iteracijomis, kur per kiekvieną iteraciją yra imamas atsitiktinis vektorius X_i , kuriam randamas neuronas nugalėtojas ir atitinkamai pakeičiamos žemėlapiu neuronų reikšmės.

1 lentelė. 1 mokymo būdu mokyto SOM kvantavimo paklaidos

Epochos	Žemėlapių dydis		
	5x5	10x10	15x15
	Kvantavimo paklaida		
10	2,23	2,17	2,16
50	1,63	1,52	1,51
1000	1,49	1,39	1,35
10000	1,49	1,38	1,34

2 lentelė. 2 mokymo būdu mokyto SOM kvantavimo paklaidos

Iteracijos	Žemėlapių dydis		
	5x5	10x10	15x15
	Kvantavimo paklaida		
50	1,52	1,36	1,43
100	1,5	1,44	1,40
1500	1,45	1,39	1,39
7500	1,49	1,38	1,32
150000	1,49	1,39	1,36

3. Rezultatai

1 mokymo būdu buvo atlikti 4 stebėjimai su kintančiu epochų skaičiumi (žr. 1 lentelę). Matosi tendencija, jog didėjant epochų skaičiui, mažėja kvantavimo paklaida. Tiesa, mažėjimas pastebimas mažesnis arba nebepastebimas, kai epochų skaičius pasiekia 1000. Taip pat didėjant žemėlapių dydžiui, mažėja ir kvantavimo paklaida, tačiau jei lygintume 15 × 15 žemėlapi su 10 × 10 dydžio žemėlapi, tai kvantavimo paklaidos pokytis nebuvo toks ryškus kaip tarp 10 × 10 ir 5 × 5 dydžio SOM.

Antru mokymo būdu buvo atlikti 5 stebėjimai su kintančiu iteracijų skaičiumi (žr. 2 lentelę). Galioja tos pačios tendencijos kaip minėtos anksčiau aptariant 1 lentelę. Tiesa, labai išaugus iteracijų skaičiui, (šiuo atveju pasiekus 150000 iteracijų), kvantavimo paklaida gavosi prastesnė nei su ankstesnėmis iteracijomis. Kitąvertus, antru mokymu būdu pavyko pasiekti neženkliai geresnes arba vienodas kvantavimo paklaidas kiekvienam žemėlapiui.

3 ir 4 lentelėse pateikiamas žemėlapis lentelės formatu, kur skaičiai žymi žemėlapių eilutės ar stulpelio numerį. Žemėlapiui sukurti buvo naudojami tie patys mokymo duomenys, kurie sudaryti iš trijų klasių.

Abejuose žemėlapiuose matyti, kad duomenys klasterizavosi, o geriausiai tai nutiko su "Iris-setosa" ir "Iris-virginica" klasių duomenimis. Šiuo atveju, saviorganizuojančio žemėlapių dydis reikšmės neturėjo. Tačiau dera atkreipti dėmesį, kad duomenų rinkinys ganėtinai paprastas, neturi daug atributų, klasių ne daug, duomenų kiekis nedidelis. Visa tai leidžia saviorganizuojantį neuorininį tinklą mokyti daug epochų, kad mažai besiskiriantys dydžiu žemėlapiai stipriai varijuotų.

3 lentelė. 1 mokymo būdu mokyto 5x5 dydžio SOM žemėlapis po 1000 epochų

	1	2	3	4	5
1					Iris-versicolor, Iris-virginica
2				Iris-versicolor, Iris-virginica	Iris-versicolor
3		Iris-versicolor	Iris-versicolor	Iris-versicolor	
4	Iris-versicolor	Iris-versicolor			
5	Iris-setosa, Iris-versicolor				

4 lentelė. 1 mokymo būdu mokyto 10x10 dydžio SOM žemėlapis po 1000 epochų

	1	2	3	4	5	6	7	8	9	10
1	Iris-setosa, Iris-versicolor	Iris-versicolor								
2		Iris-versicolor								
3		Iris-versicolor	Iris-versicolor							
4				Iris-versicolor						
5					Iris-versicolor					
6						Iris-versicolor				
7							Iris-versicolor			
8								Iris-versicolor, Iris-virginica		
9								Iris-versicolor	Iris-versicolor	Iris-versicolor
10										Iris-versicolor, Iris-virginica

4. Išvados

Kvantavimo paklaida priklauso nuo epochų (iteracijų) skaičiaus, pasirinkto SOM mokymo būdo (algoritmo). Paklaidos mažėjimas taip pat priklausomas ir nuo žemėlapių didėjimo, tačiau stipriai didinant žemėlapių matmenis paklaidos mažėjimas ima lėtėti. Su irisų duomenų rinkiniu duomenys aiškiai klasterizavosi, tačiau "iris-versicolor" buvo labiausiai išsisklaidęs. Skirtingo dydžio apmokyti žemėlapiai savo rezultatais neišskysrė stipriai, tiek 10×10 , tiek 5×5 dydžio SOM rezultatų interpretacijos yra artimos.