

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS**

**Vytautas Kraujalis**

Studijų modulio

P160M121 Duomenų tyrybos metodai ir programinės priemonės

**Laboratorinis darbas nr. 3**

Dėstytojas: dr. M. Kavaliauskas

**KAUNAS, 2022**

# DUOMENŲ RINKINYS

Pasirinkite duomenų rinkinį su vaizdais priklausančiais vienai iš kelių klasių. Rekomenduojamas vaizdų dydis nuo 50x50 iki 250x250 taškų dydžio ir šimto ar daugiau vaizdų kiekvienai klasei. Pasirinktą duomenų rinkinį pasižymėkite Moodle pateiktame dokumente. Duomenų rinkiniai tarp studentų negali sutapti.

# UŽDUOTIS

1. **Neuroniai tinklai**

Savo pasirinktam duomenų rinkiniui iš bandykite bent 2 tiesioginio sklidimo (feed-forward) ir bent 4 skirtingas sąsūkų neuroninių tinklų architektūras. Stenkitės parinkti tinklą tokį, kad gautumėte kuo geresnį klasifikavimo tikslumą. Pateikite gautus rezultatus kiekvienam iš neuroninių tinklų. Tinklą nusakykite sluoksnių kiekiu, neuronų kiekiu sluoksniuose, filtrų kiekiu ar dydžiu, aktivacijos funkcijos naudojimu, apjungimo sluoksniu, etc., o jų gautą rezultatą nusakykite klasifikavimo tikslumu (ant test imties), sumaišymo matrica ir apmokymo trukme. Pateikite tikslumo kitimo mokymosi metu grafiką. Apibūdinkite, ar tinklai linkę persimokyti. Aprašykite sukumus su kuriais susidūrėte ir kaip juos sprendėte.

1. **Neuroninių tinklų perpanaudojimas**

Sudarykite neuroninį tinklą savo pasirinktiems vaizdamas klasifikuoti. Tinklą sudarykite naudodami *imagenet* tinklo sąsūkų sluoksnius bei papildomai apmokydami savo sukurtą pilnai sujungtą tinklo dalį. Išbandykite dvi pilnai sujungto tinklo dies alternatyvas. Analogiškai, kaip ir pirmoje užduotyje, pateikite gautus rezultatus.

Apibendrinkite gautus rezultatus, pateikite išvadas.

# BENDRI REIKALAVIMAI

1. Pasižymėkite savo naudojamą duomenų rinkinį Moodel dokumente.
2. Projektas ataskaita turi būti patalpinama Moodle aplinkoje.
3. Projekto atskaitos prieduose turi būti pateikiamas pilnas R kodas, kurį galima būtų įvykdyti be klaidų ir gauti projekto aprašyme pateiktus rezultatus.

Galimi duomenų rinkinių šaltiniai:

* https://www.kaggle.com/tags/image-data

**ATASKAITA**

**DUOMENŲ RINKINYS**

....

**UŽDUOTIS 1. Sąsūkų neuroniniai tinklai**

**Tiesioginio sklidimo pilnai sujungto neuroninio tinklo architektūra nr. 1.**

Sudarėme paprastą tiesioginio sklidimo neuroninį tinklą, sudarytą iš 4 sluoksnių (lentelė žemiau). Pirmasis sluoksnis mūsų duomenis tik paverčia 1d vektoriumi, todėl jis parametrų ir neturi. Kadangi mūsų duomenys yra vaizdai, o juos visus konvertuoju į 150x150 dimensijas turinčius spalvotus vaizdus. Todėl gauname, kad mūsų įėjimo sluoksnyje yra 150 \* 150 \* 3 (kadangi RGB vaizdai) = 67500 įvestys.

Antrame sluoksnyje turime 32 neuronus, kurie yra pilnai sujungti su įvesties neuronais. Todėl šiame žingsnyje turime 32 (neuronai) \* (67500 (įvesties neuronai) + 1 (konstanta)) = 2160032 parametrus. Naudojome “ReLu” aktyvacijos funkciją.

Trečiame sluoksnyje turime 16 neuronų, kurie yra pilnai sujungti su 2 sluoksniu. Šiame sluoksnyje turime 16 (neuronai) \* (32 (2 sluoksnio neuronai kaip įvestys) + 1 (konstanta)) = 528 parametrus. Naudojome “ReLu” aktyvacijos funkciją.

Ketvirtas sluoksnis yra išėjimo sluoksnis, kuris susideda iš 9 neuronų (kadangi turime 9 klases), šis sluoksnis taip pat pilnai sujungtas su 3 sluoksniu, todėl turėsime 9 \* (16 + 1) = 153 parametrus. Naudojome “Softmax” aktyvacijos funkciją.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

===========================================================

flatten\_11 (Flatten) (None, 67500) 0

dense\_28 (Dense) (None, 32) 2160032

dense\_27 (Dense) (None, 16) 528

dense\_26 (Dense) (None, 9) 153

===========================================================

Total params: 2,160,713

Trainable params: 2,160,713

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Neuroninis tinklas iš architektūros yra labai paprastas, tačiau net ir paprastas tinklas su tokiu įvesties formatu turi net 2160713 parametrus.

Apmokydami tinklą, eigoje gauname:

Chart, line chart

Description automatically generated

pav. 1 Pirmo tiesioginio sklidimo pilnai sujungto neuroninio tinklo mokymosi procesas

Apmokymas truko 565,45 sek. (9,4 min.). Kaip matome, eigoje tinklui visiškai nepavyksta rasti tinkamų parametrų reikšmių. Kiekviena epocha beveik niekuo nesiskiria nuo kitos.

Išbandę modelį ant patvirtinamosios imties, gauname 11,11% tikslumą, tai jau yra žymiai blogiau nei atsitiktinis spėjimas. Kaip matome iš sumaišymo matricos (paveiksliukas žemiau), tinklas kiekvieną vaizdą klasifikuoja kaip 1 klasę (Gilt-Head Bream). Todėl net neapsimoka skaičiuoti kitų statistikų, kai tinklas yra visiškai nepanaudojamas.

Calendar

Description automatically generated

pav. 2 Sumaišymo matrica nr. 1

Sekantį tinklą pamėginsime sudaryti sudėtingesnį.

**Tiesioginio sklidimo pilnai sujungto neuroninio tinklo architektūra nr. 2.**

Sudarėme tiesioginio sklidimo pilnai sujungtą neuroninį tinklą su 4 sluoksniais. Pirmasis sluoksnis sudarytas tik iš vaizdų konvertavimo į vektorių, o paskutinis sluoksnis – išėjimo sluoksnis. Iš viso šiame tinkle turime 17314313 parametrų. Viduriniuose sluoksniuose naudojome „ReLu“ aktyvacijos funkcijas, o išėjimo sluoksnyje – „Softmax“.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

===========================================================

flatten\_14 (Flatten) (None, 67500) 0

dense\_40 (Dense) (None, 256) 17280256

dense\_39 (Dense) (None, 128) 32896

dense\_38 (Dense) (None, 9) 1161

===========================================================

Total params: 17,314,313

Trainable params: 17,314,313

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Stebėdami mokymosį procesą (paveikslas žemiau), matome žymiai geresnį vaizdą nei prieš tai naudotojo architektūroje. Matome, kad tikslumas ant apmokymo imties vis didėja, tačiau tikslumas ant patvirtinamosios imties po kelių epochų pradeda mažėt, dar ankščiau pradeda didėti patvirtinamosios imties nuostolių funkcijos reikšmės. Galim teigti, kad modelis per daug apsimoko ant apmokymo imties.

Chart, line chart

Description automatically generated

pav. Antro tiesioginio sklidimo pilnai sujungto neuroninio tinklo mokymosi procesas

Šį tinklą apmokyti prireikė 614,39 sek. (10,24 min.). Modelis sudėtingesnis, nei pirmoje dalyje naudota architektūra, apmokymas truko tik šiek tiek ilgiau, o tikslumas ant patvirtinamosios imties siekia 54,8%. Tačiau toks tikslumas yra tik šiek tiek geresnis nei atsitiktinis sprendimas.

Calendar

Description automatically generated

pav. Sumaišymo matrica nr. 2

7 klasės teisingų spėjimų buvo tik 21 (iš 200), šios klasės jautrumas siekia tik 0,10. Šią klasę tinklas dažniausiai sumaišė su 5 klase.

A fish on a blue surface

Description automatically generated with medium confidence

pav. 7 klasės pavyzdys

A picture containing text, fish

Description automatically generated

pav. 5 klasės pavyzdys

A picture containing text, plaque

Description automatically generated

pav. Klasifikavimo statistikos nr. 2

Apskritai, šis tinklas rodo gana didelį specifiškumą, kadangi visų klasių specifiškumas siekia >0,94 ir tik 2 klasė (0,81) ir 5 klasė (0,77) turi mažesnį specifiškumą.

**Tiesioginio sklidimo pilnai sujungtų neuroninių tinklų išvada.**

Tokio tipo architektūra neparodė gerų rezultatų, sudėtingesnės architektūros nepavyko sudaryti dėl resursų trūkumo, kuomet R programa neleido sudaryti sudėtingesnio tinklo su daugiau neuronų per kelis sluoksnis dėl atminties trūkumo. Kas buvo pastebėta, jog tinklas sugeba klasifikuoti teisinga linkme tik tuomet, kada sluoksnyje prieš išėjimą turime mažiausiai 128 neuronų sluoksnį (jei naudoju tik neuronų skaičių, kurie gaunami keliant 2 laipsniu). Taip pat rezultatas nemažai priklausė ir nuo „batch size“ parinkimo, kuomet naudojame didesnį vaizdų kiekį, rezultatai yra prastesni, bei nuo vaizdų dydžio. Buvo bandyta naudoti 200x200 RGB vaizdus, tačiau apmokymas trūko pernelyg ilgai.

**Sąsūkų neuroninio tinklo architektūra nr. 1.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

===========================================================

conv2d\_1 (Conv2D) (None, 150, 150, 32) 896

max\_pooling2d\_1 (MaxPool ing2D) (None, 75, 75, 32) 0

dropout\_2 (Dropout) (None, 75, 75, 32) 0

conv2d (Conv2D) (None, 75, 75, 16) 4624

max\_pooling2d (MaxPoolin g2D) (None, 37, 37, 16) 0

dropout\_1 (Dropout) (None, 37, 37, 16) 0

flatten (Flatten) (None, 21904) 0

dense\_1 (Dense) (None, 16) 350480

dropout (Dropout) (None, 16) 0

dense (Dense) (None, 9) 153

===========================================================

Total params: 356,153

Trainable params: 356,153

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Chart, line chart

Description automatically generated

Apmokymo accuracy 37,6%

839.75 sec elapsed

Calendar

Description automatically generated

Text

Description automatically generated

6 klasės jautrumas – 0. Todėl nevertinsim

**Sąsūkų neuroninio tinklo architektūra nr. 2.**

odel: "sequential\_1"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

===========================================================

conv2d\_3 (Conv2D) (None, 150, 150, 32) 896

max\_pooling2d\_3 (MaxPool ing2D) (None, 75, 75, 32) 0

dropout\_6 (Dropout) (None, 75, 75, 32) 0

conv2d\_2 (Conv2D) (None, 75, 75, 64) 18496

max\_pooling2d\_2 (MaxPool ing2D) (None, 37, 37, 64) 0

dropout\_5 (Dropout) (None, 37, 37, 64) 0

flatten\_1 (Flatten) (None, 87616) 0

dense\_4 (Dense) (None, 128) 11214976

dropout\_4 (Dropout) (None, 128) 0

dense\_3 (Dense) (None, 64) 8256

dropout\_3 (Dropout) (None, 64) 0

dense\_2 (Dense) (None, 9) 585

===========================================================

Total params: 11,243,209

Trainable params: 11,243,209

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Chart, line chart

Description automatically generated

Apmokymo acc 74,5%

961.98 sec elapsed

Calendar

Description automatically generatedS

Text

Description automatically generated

A picture containing text, plaque

Description automatically generated

Val\_acc didesnis:

This indicates the presence of high bias in your dataset. It is underfitting. The solutions to issue are:-

1. Probably the network is struggling to fit the training data. Hence, try a little bit bigger network.
2. Try a different Deep Neural Network. I mean to say change the architecture a bit.
3. Train for longer time.
4. Try using advanced optimization algorithms.

**Sąsūkų neuroninio tinklo architektūra nr. 3.**

**Sąsūkų neuroninio tinklo architektūra nr. 4.**

**UžDUOTIS 2. Neuroninių Tiklų perpanaudojimas**

....