

RĪGAS TEHNISKĀ UNIVERSITĀTE

Datorzinātnes un informācijas tehnoloģijas fakultāte

Mākslīgā intelekta un sistēmu inženierijas katedra

**Mašīnmācīšanās algoritmu lietojums**

**2. praktiskais darbs mācību priekšmetā  
“Mākslīgā intelekta pamati”**

**Izstrādāja:** Kristiāna Heniņa

Studenta apliecības nr. 201RDB196

2.kurss 8.grupa

**Pārbaudīja:**

Dr.sc.ing. , Dr.paed.asoc.prof. Alla

Anohina-Naumeca

**Izveidotais projekts:** [**https://github.com/kikkiioo/AI\_klasifikacija\_201RDB196**](https://github.com/kikkiioo/AI_klasifikacija_201RDB196)

**Datu kopa:** [**https://www.kaggle.com/datasets/deepu1109/star-dataset**](https://www.kaggle.com/datasets/deepu1109/star-dataset)

2021/22. māc. gads

**Orange rīka darbplūsmas atspoguļojums** *(att. 1.45)*

**Diagram, schematic

Description automatically generated**

*att. 1.45 Orange rīks*

**Saturs**

[**Orange rīka darbplūsmas atspoguļojums** 2](#_Toc104495231)

[**I daļa - Datu pirmapstrāde un izpēte** 4](#_Toc104495232)

[**1.** **Datu kopas apraksts** 4](#_Toc104495233)

[**2.** **Datu kopas satura apraksts** 4](#_Toc104495234)

[**3.** **Datu kopa vizuāli** 6](#_Toc104495235)

[**3.1** **Izkliedes diagrammas** 6](#_Toc104495236)

[**3.2** **Histogrammas un sadalījuma analīze** 8](#_Toc104495237)

[**3.3** **Statiskie rādītāji** 10](#_Toc104495238)

[**II daļa – Nepārraudzītā mašīnmācīšanās** 11](#_Toc104495239)

[**1.** **Hierarhiskā klasterizacija** 11](#_Toc104495240)

[**1.1** **Orange rīka apraksts** 11](#_Toc104495241)

[**1.2** **Pirmais ekspriments** 12](#_Toc104495242)

[**1.3** **Otrais eksperiments** 13](#_Toc104495243)

[**1.4** **Trešais eksperiments** 14](#_Toc104495244)

[**2.** **K – vidējo algoritms** 15](#_Toc104495245)

[**2.1** **Orange rīka apraksts** 15](#_Toc104495246)

[**2.2** **Pirmais eksperiments** 15](#_Toc104495247)

[**2.3** **Otrais eksperiments** 16](#_Toc104495248)

[**2.4** **Trešais eksperiments** 17](#_Toc104495249)

[**2.5** **Ceturtais eksperiments** 17](#_Toc104495250)

[**2.6** **Piektais eksperiments** 18](#_Toc104495251)

[**3.** **Algoritmu darbības analīze** 19](#_Toc104495252)

[**III daļa - Pārraudzītā mašīnmācīšanās** 20](#_Toc104495253)

[**1.** **kNN klasifikācijas algoritms** 20](#_Toc104495254)

[**1.1** **Orange rīka apraksts** 20](#_Toc104495255)

[**1.2** **Testa un apmācību datu kopas** 21](#_Toc104495256)

[**1.3** **Hiperparametru vērtības eksperimentos** 23](#_Toc104495257)

[**1.4** **Secinājumi** 24](#_Toc104495258)

[**2.** **SVM klasifikācijas algoritms** 25](#_Toc104495259)

[**2.1** **Orange rīka apraksts** 25](#_Toc104495260)

[**2.2** **Testa un apmācību datu kopas** 27](#_Toc104495261)

[**2.3** **Hiperparametru vērtības eksperimentos** 27](#_Toc104495262)

[**2.4** **Secinājumi** 28](#_Toc104495263)

[**3. Neironu tīkli** 28](#_Toc104495264)

[**3.1** **Orange rīka apraksts** 28](#_Toc104495265)

[**3.2** **Testa un apmācību datu kopas** 29](#_Toc104495266)

[**3.4** **Secinājumi** 30](#_Toc104495267)

[**Izmantotie avoti** 31](#_Toc104495268)

**I daļa - Datu pirmapstrāde un izpēte**

## **Datu kopas apraksts**

Darbā tiks izmantota datu kopa, kas satur informāciju par dažādu zvaigžņu iezīmēm. Datu kopu platformā kaggle [[1]](#footnote-1) 29. oktrobrī 2019. gadā plkst. 22:47 izveidoja Deepraj Baidya ar nosaukumu “Star dataset to predict star type”(tulkojums - Zvaigžņu datu kopa, lai prognozētu zvaigžņu veidus). Datu kopa tika izveidota, lai pierādītu to, ka zvaigznes debesu telpā seko noteiktam grafikam – HR diagrammai, lai varētu klasificēt zvaigznes pamatojoties uz šādu grafiku. Datu kopai ir licencēšanas nosacījums – “Data files © Original Authors”. Dati datu kopai tika savākti 3 nedēļu laikā un tie satur 240 zvaigžņu ierakstus, dati tika savākti no Wikipedia[[2]](#footnote-2) un grāmatas[[3]](#footnote-3). Trūkstošie dati tika aprēķināti pēc astrofizikas formulām. [[4]](#footnote-4)

## **Datu kopas satura apraksts**

Datu objektu skaits datu kopā – 240. Klašu skaits datu kopā – 6. Katra klase raksturo

konkrētu zvaigzni un tās šajā datu kopā tiek apzīmētas ar cipariem : 0 – Brūnais punduris, 1 – Sarkanais punduris, 2 – Baltais punduris, 3 – Pamat-kārtības zvaigzne, 4 – Supergiants, 5 – Hipergiants. Tā kā klases kopā ir 6, tad katrai klasei pieder 40 objekti. Pazīmju (atribūtu) atspoguļojums parādīts zemāk tabulā. (tab. 1.1) [[5]](#footnote-5)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pazīmes (atribūti) | | | | | |
| N.p.k | Nosaukums | Nozīme | Apzīmējums | Diapazons | Vērtību tips |
| 1. | Absolūtā temperatūra | Zvaigznes virsmas absolūtā temperatūra | K (Kelvini) | 1939 –  40000 | Numeriska,  float |
| 2. | Relatīvais spilgtums | Zvaigznes spilgtums attiecībā pret sauli | L/Lo | 8E-05 –  849420 |  |
| 3. | Relatīvs rādiuss | Zvaigžņu rādiuss attiecībā pret saules rādiusu | R/Ro | 0,00840 –  1948,5 | Numeriska,  float |
| 4. | Absolūtā spilgtuma pakāpe ( magnitude ) | Zvaigznes spilgtuma pakāpe, kura nosaka vai zvaigzni iespējams redzēt ar neapbruņotu aci. | Mv | 11,920 –  20,060 | Numeriska,  float |
| 5. | Zvaigznes krāsa | Zvaigznes krāsa noteikta pēc spektrālās analīzes | Vairāki: white,Red,Blue,Yellow,yellow-orange un citas krāsas | 17 vērtības | Kategoriska,  String |
| 6. | Spektrālā  klase | Zvaigznes noteiktā spektrālā klase | Vairāki:  O,B,A,F,G,K,M | 7 vērtības | Kategoriska,  String |
| 7. | Zvaigznes  tips | Zvaigznes tips no 6 dažādām klasēm | Vairāki:  0,1,2,3,4,5 | No 0 līdz 5, veseli skaitļi | Numeriska,  Int |

*tab. 1.1. Pazīmju raksturojums*

Failu struktūras fragments: (att. 1.1) [[6]](#footnote-6)

Table, calendar

Description automatically generated

*att. 1.1 Faila struktūras fragments no kaggle.com*

## **Datu kopa vizuāli**

### **Izkliedes diagrammas**

Tika izveidotas divas izkliedes diagrammas. Pirmajā izkliedes diagrammā uz x ass

atspoguļota temperatūra Kelvinos, bet uz y absolūtā zvaigznes spilgtuma pakāpe ( att 1.2). Otrajā izkliedes diagrammā uz x ass atspoguļota absolūtā zvaigznes spilgtuma pakāpe, bet uz y ass zvaigznes krāsa, kura noteikta pamatojoties pēc spektrālās analīzes ( att 1.3).

**A picture containing chart

Description automatically generated**

*att. 1.2. Izkliedes diagramma*

**Chart

Description automatically generated**

*att. 1.3. Izkliedes diagramma*

Apskatot izkliedes diagrammas var veikt secinājumus. Diagrammās var redzēt to, ka

datu objekti, kas pieder dažādām klasēm ir skaidri atdalāmi, jo diagrammas vizuālais atspoguļojums ļauj skaidri redzēt katru datu struktūru. Vizuālajā atspoguļojumā var viegli izskatīt, ka ir kopā ir 6 klases, jo katra klase atspoguļota savā krāsā – gaiši dzeltens, dzeltens, gaiši zaļš, tumši zaļš, gaiši zils un tumši zils. Var secināt, ka vizualizētie datu grupējumi pārsvarā savā starpā neatrodas tālu no viens otra, taču atsevišķi datu grupējumi viens no otra atrodas lielā attālumā.

### **Histogrammas un sadalījuma analīze**

Tika izveidotas divas histogrammas pamatojoties uz ieinteresējošām pazīmēm –

zvaigznes virsmas temperatūra (att. 1.4) un absolūtā zvaigznes spilgtuma pakāpe ( att. 1.5), kas atspoguļo šo pazīmju sadalījumu datu kopā. Kā arī tika izveidotas 2 histogrammas, kas parāda klašu atdalīšanu pamatojoties uz atribūtiem – zvaigznes spilgtuma pakāpe (att. 1.6) un zvaigznes virsmas temperatūra ( att. 1.7).

Chart, histogram

Description automatically generated

*att. 1.4 Histogramma zvaigžņu spilgtuma pakāpēm*

Histogram

Description automatically generated with medium confidence

*att. 1.5 Histogramma zvaigžņu virsmas temperatūrai*

Var secināt, ka histogrammā, kura atspoguļo zvaigžņu spilgtuma pakāpes eksistē

normālais sadalījums un datu kopā objektiem ir dažādas šīs pazīmes vērtības, taču histogrammā, kas atspoguļo zvaigžņu virsmas temperatūru vērtība, kas atrodas starp 3000 K un 4000 K piemīt trešdaļai no visas datu kopas objektiem.

A picture containing chart

Description automatically generated

*att. 1.6. Histogramma klašu atdalīšanai pēc atribūta temperatūra*

*Chart, histogram

Description automatically generated*

*att. 1.7. Histogramma klašu atdalīšanai pēc zvaigžņu spilgtuma pakāpes*

Var secināt, ka pie šīm pazīmēm ir dažas klases, kas izteikti dominē. Pēc šīs sadalījuma analīzes nevar labi identificēt datu grupējumus, jo tie mēdz arī saplūst kopā.

### **Statiskie rādītāji**

Chart

Description automatically generated with low confidenceTika aprēķināti statiskie rādītāji datu kopas atribūtiem ( att. 1.8).

*att. 1.8. Statiskie rādītāji*

Atribūti :

1. Zvaigznes virsmas temperatūra (K)

* Dispersija = 0,91
* Vidējā vērtība = 10497,46

1. Zvaigznes relatīvais spilgtums (L/Lo)

* Dispersija = 1,6705
* Vidējā vērtība = 107188

1. Zvaigznes rādiuss (R/Ro)

* Dispersija = 2,1760923
* Vidējā vērtība = 237,1577814

1. Zvaigznes spilgtuma pakāpe (Mv)

* Dispersija = 2,39836
* Vidējā vērtība = 4,38240

1. Zvaigznes krāsa

* Dispersija = 1,7

1. Zvaigznes spektrālā klase

* Dispersija = 1,48

Var secināt, ka kvadrātiskā novirze no izlases vidējā lieluma datu kopas objektiem nav liela, jo dispersijas vērtība visiem atribūtiem nav liela. Vidējās vērtības vairākām pazīmēm atrodas tuvāk minimālajai vērtībai, kas nozīmē, ka pārējā diapazonā līdz maksimālajai vērtība klašu objekti nav vienmērīgi sadalīti.

**II daļa – Nepārraudzītā mašīnmācīšanās**

## **Hierarhiskā klasterizacija**

### **Orange rīka apraksts**

Lai izveidotu hierarhisko klasterizāciju datu tabula tika savienota ar atribūtu

“distances”un pēc tam šis atribūts tika savienots ar “Hierarchical Clustering”. “Distances” atribūts aprēķina attālumu starp rindām un kolonām datu kopā. Tiek izņemts ķeksis no izvēles normalized, jo normalizācija notiek kolonu starpā un vērtības tiek tuvinātas nullei un mērogotas. [[7]](#footnote-7) “Hierarchial Clustering” atribūts sagrupē datus balstoties uz hierarhiskās klasterizācijas algoritmu. Šis atribūts gala rezultātā parāda dendogrammu. Šim rīkam piemīt vairāki hiperparametri. Tiek piedāvāti dažādi veidi, kā mērīt attālumus starp klasteriem ar izvēlni “Linkage”. “Signle linkage” – aprēķina distanci starp tuvākajiem elementiem starp diviem klasteriem, “Average linkage” – aprēķina vidējo distanci starp diviem klasteriem, “Weighted linkage” – lieto WPGMA metodi [[8]](#footnote-8), “Complete linkage” – aprēķina distanci starp divu klasteru vistālākajiem datu objektiem, “Ward linkage” – aprēķina kļūdu kvadrātu summas pieaugumu. “Annotation” izvēlnē tiek izvēlēti dendogrammas etiķešu uzraksti. “Pruning ” izvēlne ļauj izvēlēties maksimālo dendogrammas dziļumu. “Selection” ļauj izvēlēties kā dati tiks parādīti. “Manual” – var izvēlēties vienu klasteri, “Height ratio” – ja izvēlas šo, tad ir iespēja bīdīt līniju un mainīt klasteru skaitu, “Top N” – atļauj izvēlēties N klasteru skaitu. [[9]](#footnote-9) Šajā darbā tiks mainīti “Linkage” un “Selection” hiperparametri, kā arī tiks izvēlēts dendogrammas maksimālais dziļums.

### **Pirmais ekspriments**

Etiķetes visiem eksperimentiem būs zvaigznes tips, jo zvaigznes tips ir klase. Šajā eksperimentā “Linkage” tika izvēlēts, kā “Single”. Tas nozīmē, ka klasteri tiks veidoti pamatojoties uz attālumu tuvākajiem elementiem starp diviem klasteriem. “Height ratio” tiks izvēlēts – 40,0% (att 1.9.).

Chart, box and whisker chart

Description automatically generated

*att. 1.9. Pirmais eksperiments - Hierarhiskā klasterizācija*

Gala rezultātā pēc dendogrammas nolasās, ka ir izveidoti 6 klasteri. Visvairāk datu objektu satur klasteris C6. Var redzēt, ka C5 klasterī datu objekti ir vislīdzīgākie ņemot vērā to attālumu, jo stabiņu augstums ir vismazākais starp visiem klasteriem.

### **Otrais eksperiments**

Otrajā eksperimentā “Linkage ” izvēlēts kā “Average”. Tas nozīmē, ka tiks aprēķināta

vidējā distance starp viena klastera vidējo attālumu starp datu objektiem un otra klastera datu objektiem. “Height ratio” tiks izvēlēts – 30,0%, lai iegūtu lielāku klasteru skaitu (att 1.10).

Chart, box and whisker chart

Description automatically generated

*att. 1.10. Otrais eksperiments – Hierarhiskā klasterizācija*

Datu objekti šajā eksperimentā tika sadalīti 4 klasteros. Vismazāk datu objektu ir klasteros C1 un C2, taču savā starpā klasteri ir līdzīgi. Vismazākie attālumi, līdz ar to arī vislielākā līdzība starp datu objektiem ir klasterī C4, turpretim klasterī C4 var redzēt to, ka objekti nav tik līdzīgi un attālumi šajā sadalījumā starp datu objektiem ir vislielākie.

### **Trešais eksperiments**

Trešajā eksperimentā “Linkage” tiks norādīts kā “Complete”. Tas nozīmē, ka veidojot klasterus, tiks ņemts vērā attālums starp klasteru vistālākajiem elementiem. “Selection” sadaļā tika izvēlēts “Top N”, kura vērtība tika izvēlēta 6. Tas nozīmē, ka tiks izveidoti seši klasteri. Šādu vērtību izvēlējos, jo šāds ir klašu skaits datu kopā (att. 1.11).

Chart, box and whisker chart

Description automatically generated

*att. 1.11. Trešais eksperiments – Hierarhiskā klasterizācija*

Šajā eksperimentā datu objekti tika sadalīti 6 klasteros, kā to nosaka “Top N”. Visvairāk datu objektu klasterizēti klasterī C3, kā arī šeit objekti ir vislīdzīgākie pēc to garākajiem attālumiem. Klasterī C6 pēc dendogrammas nolasās, ka objekti šeit salīdzinot ar pārējiem klasteriem ir visnelīdzīgākie. Vismazāk datu objektu ir klasteros C1 un C2 un tiem arī savā starpā nav lielas līdzības, jo stabiņi salīdzinājumā ar citiem klasteriem ir augsti, taču savā starpā šie klasteri ir līdzīgi.

## **K – vidējo algoritms**

### **Orange rīka apraksts**

Rīks “k-Means” grupē datu objektus izmantojot K – vidējo algoritmu. Ieejas dati šim rīkam ir datu tabula. Šis algoritms arī aprēķina silueta koeficientu, kas nosaka cik klasteri ir piemērotāki sadalījumam. Šajā rīkā ir iespēja izmainīt klasteru skaitu. Iestatīt kā “fixed” – specifisks klasteru skaits vai arī “From X to Y”, kas atļauj izvēlēties klasteru skaitu no X līdz Y un aprēķina silueta koeficientus. Kā arī ir iespēja izvēlēties inicializācijas metodi. Šajā gadījumā tā būs k-Means++, jo šī metode centroīdu izvēlas nejauši un pēc tam nākamie tiek izvēlēti pamatojoties uz attālumu no centroīdiem. [[10]](#footnote-10) Šajā darbā tiks mainītas k vērtības, kas nozīmē, ka tiks mainīts klasteru skaits.

### **Pirmais eksperiments**

Pirmajā eksperimentā k = 5 jeb klasteru skaits ir 5 (att. 1.11). No iegūtās silueta vērtības pie k = 5 var secināt, ka šis klasteru skaits ir pārāk mazs, jo silueta vērtība ir maza un ir tuvāk 0 nekā 1.

Graphical user interface, application

Description automatically generated

*att. 1.11. k = 5, K – vidējo algoritms*

### **Otrais eksperiments**

Otrajā eksperimentā k = 10 jeb klasteru skaits ir 10 (att. 1.12). Šajā eksperimentā tika

iegūta lielāka silueta vērtība, kas nozīmē, ka datu kopu ir jēdzīgi dalīt 10 klasteros.

Graphical user interface

Description automatically generated

*att. 1.12. k = 10, K – vidējo algoritms*

### **Trešais eksperiments**

Šajā eksperimentā tika palielināta k vērtība, k = 15(att.1.13).Tas tika darīts lai noskaidrotu vai ir iespējams iegūt vēl labāku sadalījumu klasteros aprēķinot silueta vērtību. Rezultātā var redzēt, ka sadalījums vēl vairāk klasteros deva labāku silueta vērtību.

Graphical user interface

Description automatically generated

*att. 1.13. k = 15, K – vidējo algoritms*

### **Ceturtais eksperiments**

Šajā eksperimentā tika palielināta k vērtība, k = 20 (att.1.14).Tas tika darīts, lai noskaidrotu vai ir iespējams iegūt vēl labāku sadalījumu klasteros aprēķinot silueta vērtību. Rezultātā var redzēt, ka sadalījums 19 klasteros dod vēl labāku silueta vērtību nekā iepriekšējos eksperimentos norādītie klasteri.

Graphical user interface

Description automatically generated

*att. 1.14. k = 20, K – vidējo algoritms*

### **Piektais eksperiments**

Pēdējā eksperimentā klasteru skaits tika palielināts līdz maksimumam, k = 30 (att. 1.15).

Šajā eksperimentā aprēķinātās silueta vērtības vairs nepalielinās lielākam klasteru skaitam un vislabākā silueta vērtība paliek pie k = 19. Tas nozīmē, ka vislabāk datu kopu būtu dalīt 19 klasteros.

Graphical user interface

Description automatically generated

*att. 1.15. k = 30, K – vidējo algoritms*

## **Algoritmu darbības analīze**

Pēc abu algoritmu apskates un eksperimentēšanas var secināt, ka hierarhiskās

klasterizācijas algoritms pēc rezultātiem ir noderīgāks, jo netika veidoti tik daudz klasteri kā K – vidējo algoritmam. Klases datu kopā ir tikai 6, taču K – vidējo algoritms piedāvā sadalījumu 19 klasteros ar vislabāko silueta vērtību. Šāds klasteru skaits tika piedāvāt tāpēc, ka vērtības atrodas diezgan tālu viena no otras. “Silhouette Plot” var redzēt, ka ir pazīmes ar kurām ļoti labi var atdalīt datu objektus taču ir pazīmes ar kurām ir ļoti grūti atdalīt pazīmes šajā datu kopā (att. 1.16). Taču, ja izvēlamies datus no K – vidēja algoritma apskatīt kā tie sadalīti izkliedes diagrammā un uz asīm novietot pazīmes, kas jau iepriekš darbā tika apskatītas un secinātas, ka šīs pazīmes labi atdala datu objektus , tad ir iespēja redzēt šo klasteru normālu sadalījumu (att. 1.17).

**Chart, funnel chart

Description automatically generated**

*att. 1.16. “Silhouette Plot”*

A picture containing chart

Description automatically generated

*att. 1.17. Izkliedes diagramma K – vidējo algoritmam*

**III daļa - Pārraudzītā mašīnmācīšanās**

## **kNN klasifikācijas algoritms**

Klasifikācijai izvēlējos kNN klasifikācijas algoritmu jeb k-tuvāko kaimiņu metodi. Šo

algoritmu pirmkārt izvēlējos, jo to pielieto klasifikācijas uzdevumā. Šis algoritms meklējot datu objektam atbilstošu klasi no sākuma atrod līdzīgāko kaimiņu šim datu objektam un piešķir datu objektam atrasto kaimiņa klasi. Kaimiņu skaitu apzīmē burts k . Šis algoritms pielieto līdzības jēdzienu, piemēram, tiek atrasts ģeometriskais attālums starp datu objektiem jeb Eiklīda attālums. Pēc tam šis algoritms visus aprēķinātos attālumus sagrupē augošā secībā un paņem no attālumu sarasta k datu objektus. No šiem objektiem algoritms iegūst objektu iezīmes un pēc tam jau tiek atgriezta visbiežākā klases iezīme kā tas ir klasifikācijas uzdevumā. Šim algoritmam nav apmācības fāzes un klasifikācijai tiek izmantoti visi dati datu kopā. [[11]](#footnote-11)

### **Orange rīka apraksts**

Lai izveidotu kNN klasifikācijas algoritmu orange rīkā tika pievienots rīks “Test and

Score”, lai varētu veikt secinājumus par algoritma darību, kā arī pats kNN algoritms. Šajā rīkā ir iespēja iestatīt kaimiņu skaitu “Neighbors” , metrikas kādā rēķināsies attālums “Metric” un svarus, kuri tiks izmantoti “Weight”. Šajā darbā tiks mainīts kaimiņu skaits, taču metrikas tiks izmantots Eiklīda attālums (“taisna līnija”, attālums starp diviem datu objektiem) un svari tiks uzstādīti kā uniform – visi punkti katrā kaimiņu apgabalā tiks svērti vienlīdzīgi. [[12]](#footnote-12) Dati tiks testēti un rezultāti apskatīti rīkā – “Test and score”. Eksperimenti tiks veikti atzīmējot izvēlni – test on train, kas nozīmē, ka tiks izmantota datu kopa no sākuma mācīšanās nolūkam un pēc tam testēšanai, kā arī eksperimenti tiks veikti atzīmējot izvēlni – test on test data, kas nozīmē, ka tiks izmantoti tikai rīkam padotie dati. [[13]](#footnote-13)

### **Testa un apmācību datu kopas**

Lai izveidotu testa un apmācību datu kopas tika izmantots rīks “Data sampler” un, lai

uzzinātu datu objektu skaitu testa un apmācību datu kopās un katrā klasē atsevišķi tika izmantots rīks “Distributions”. “Data sampler” tika fiksēts tas, ka apmācību datu kopu saturēs 60% no datu objektiem, taču testa datu kopu saturēs 40 % no datu objektiem ( att. 1.18). Tas nozīmē, ka apmācību datu kopa saturēs 144 datu objektus, taču testa datu kopa saturēs 96 datu objektus. Lai iegūtu informāciju par to, cik datu objektu no katras klases ir iekļauts katrā datu kopā katrā klasē tika izmantots rīks “Distributions” (att. 1.19 – apmācību dati un att. 1.20 – testa dati). Apmācību datu kopā klasēs objekti tika sadalīti – “0” klase 22 objekti (15,28%), “1” klase 24 objekti (16,67%), “2” klase 18 objekti (12,50%), “3” klase 25 objekti (17,36%), “4” klase 25 objekti (17,36%) un “5” klase 40 objekti (20,83%). Testa datu kopā klasēs objekti tika sadalīti – “0” klase 28 objekti (18,75%), “1” klase 16 objekti (16,67%), “2” klase 22 objekti (22,92%), “3” klase 15 objekti (15,62%), “4” klase 15 objekti (15,62%) un “5” klase 10 objekti (10,42%).

Graphical user interface, application

Description automatically generated

*att. 1.18 “Data sampler”*

*Chart, bar chart

Description automatically generated*

*att. 1.19 Apmācību dati klasēs*

*Chart, bar chart

Description automatically generated*

*att. 1.20 Testa dati klasēs*

* 1. **Hiperparametru vērtības eksperimentos (tab 1.2)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.p.k | k – kaimiņu vērtība | Metrika | Svari | Apmācību/testa dati |
| 1. | 2 | Eiklīda | Uniform | apmācību |
| 2. | 5 | Eiklīda | Uniform | apmācību |
| 3. | 10 | Eiklīda | Uniform | apmācību |
| 4. | 2 | Eiklīda | Uniform | testa |
| 5. | 5 | Eiklīda | Uniform | testa |
| 6. | 10 | Eiklīda | Uniform | testa |

*tab. 1.2 Hiperparametru vērtības*

Eksperimentu iegūtie rezultāti ar apmācību datiem (att. 1.21, att. 1.22, att. 1.23). Eksperimentu iegūtie rezultāti ar testa datiem (att. 1.24, att 1.25, att. 1.26).

Graphical user interface, application, table

Description automatically generated

*att. 1.21. 1.eksperiments*

*Graphical user interface, application, table

Description automatically generated*

*att. 1.22. 2.eksperiments*

*Graphical user interface, application, table

Description automatically generated*

*att. 1.23. 3.eksperiments*

**

*att. 1.24. 4.eksperiments*

*Graphical user interface, application

Description automatically generated*

*att. 1.25. 5.eksperiments*

**

*att. 1.26. 6.eksperiments*

### **Secinājumi**

Secinājumi tiks veiktie pamatojoties uz dažādām piedāvātajām statistikām. AUC – laukums zem ROC līknes, CA – pareizi klasificētu datu objektu proporcija, F-1 – precizitātes harmoniskais vidējais rādītājs, Precision – patieso pozitīvo gadījumu skaits starp visiem, kas klasificēti kā pozitīvi[[14]](#footnote-14), recall – jūtīguma metrika, kas aprēķina proporciju starp tiem datu objektiem, kas klasificēti kā pozitīvi salīdzinājumā ar kopējo datu objektu ar pozitīvām iezīmēm. [[15]](#footnote-15) Pirmie trīs eksperimenti tika veikti izmantojot apmācību datus.

Var secināt to, ka vislabākie algoritma rezultāti tika iegūti, kad tika izmantoti apmācības dati un tika izmantoti 2 kaimiņi algoritmā. Kā arī rezultāti tika iegūti labāki, tad, kad tika izmantoti apmācību dati, jo modelis no sākuma sevi apmācīja un tad pēc tam šos datus izmantoja kā rezultātā tika iegūti labāki rezultāti, jo modelim jau bija zināšanas. Vissliktākie rezultāti tika iegūti, kad tika izmantoti testa dati un maz kaimiņu – 2.

## **SVM klasifikācijas algoritms**

SVM jeb atbalsta vektora mašīna darbā tika izmantots kā otrs klasifikācijas algoritms. Ir interesanti apskatīt šo algoritmu, jo tas ir ļoti piemērots klasificēt lielus teksta gabalus, taču šajā darbā netiek izmantoti lieli teksta fragmenti, bet vairākas iezīmes gan skaitliskas, gan tekstveida, tāpēc būs interesanti apskatīt kā algoritms tiks galā ar šāda veida klasifikāciju. Algoritms darbojas attēlojot testa datus plaknē un pēc tam no šiem visiem punktiem izvada hiperplakni, kas divās dimensijās ir līnija. Šī līnija atspoguļo robežu un katrā pusē dati tiek klasificēti. Vislabākā hiperplakne ir tā, kuras attālums līdz katram datu objektam ir vislielākais.[[16]](#footnote-16)

### **Orange rīka apraksts**

Orange rīkā tika pievienots šī algoritma rīks “SVM”, kas tika savienots ar iepriekš

aprakstīto “Test and Score”. Rīkam “SVM” piemīt vairāki hiperparametri. Izvēlnē SVM tips tiks izvēlēts parametrs v-SVM, jo šo parametru var izmantot klasifikācijas uzdevumā. Augšējais parametrs norāda soda termiņu par zaudējumiem, taču tas attiecas tikai uz regresijas uzdevumu. Apakšējais parametrs norāda apmācības kļūdas augšējo robežu un atbalsta vektoru daļas apakšējo robežu (Complexiy bound (v)). “Kernel” ir funkcija, kas pārveido iezīmes par funkciju telpu tā lai atbilstu maksimālās robežas hiperplaknei, tādējādi ir iespēja izvēlēties ar ko algoritms veidos modeli – “linear”, “polynomial”, “RBF” vai “Sigmoid” funkcijām.[[17]](#footnote-17) (att. 1.27) Funkciju automātiski ģenerētie parametri un to formulas pievienotas kā attēli ( att. 1.28, att. 1.29, att. 1.30, att. 1.31) Ir iespēja arī izmainīt optimizācijas parametrus.

Graphical user interface, application

Description automatically generated

*att. 1.27. SVM rīks*

**

*att. 1.28. Lineāra funkcija*

*Graphical user interface, application

Description automatically generated*

*att. 1.29. Polinomu funkcija*

*Graphical user interface, text, application, Word

Description automatically generated*

*att. 1.30. RBF funkcija*

*Graphical user interface, text, application, Word

Description automatically generated*

*att. 1.31. Sigmoid funkcija*

### **Testa un apmācību datu kopas**

Testa un apmācību datu kopas tiks izmantotas tādas pašas kā kNN algoritmā ( skatīt III

daļas 1. nodaļas 1.2 apakšnodaļu)

* 1. **Hiperparametru vērtības eksperimentos (tab. 1.3)**

|  |  |  |  |
| --- | --- | --- | --- |
| N.p.k | Testa/apmācību dati | Izmantotā funkcija | Complexity Bound (v) |
| 1. | Apmācību | Lineāra | 0,5 |
| 2. | Apmācību | Polinomu | 0,5 |
| 3. | Apmācību | Sigmoid | 0,5 |
| 4. | Testa | Lineāra | 0,5 |
| 5. | Testa | Polinomu | 0,5 |
| 6. | Testa | Sigmoid | 0,5 |

*tab. 1.3 Hiperparametru vērtības*

Eksperimentu iegūtie rezultāti ar apmācību datiem (att. 1.32, att. 1.33, att. 1.34). Eksperimentu iegūtie rezultāti ar testa datiem (att. 1.35, att 1.36, att. 1.37).



*att. 1.32 1.eksperiments*

**

*att. 1.33 2.eksperiments*

**

*att. 1.34 3.eksperiments*

**

*att. 1.35 4.eksperiments*

**

*att. 1.36 5.eksperiments*

**

*att. 1.37 6.eksperiments*

* 1. **Secinājumi**

Kā jau kNN algoritmā tika secināts – apmācību datu kopu izmantojot tiks iegūti labāki modeļa veiktspējas rezultāti. Izmantojot testa datu kopu modelis netiek vispirms apmācīts līdz ar to ir sliktāki rezultāti nekā apmācītam modelim. Vislabākie modeļa veiktspējas rezultāti abās funkcijās tika iegūti izmantojot sigmoid funkciju.

## **3. Neironu tīkli**

* 1. **Orange rīka apraksts**

Lai izveidotu neironu tīklu orange rīkā tiek pievienots rīks “Neural network”. Šajā rīkā ir iespējas definēt neironu tīkla struktūru un definēt cik neironu ir katrā slānī. “Activaton” ir funkcija, kura tiek izmantota lai aktivizētu slēpto slāni. Šajā darbā tiks izmantota “Logistic” funkcija, kas ir sigmoida funkcija. Šajā rīkā ir arī hiperparametrs “solver”, kuru norāda kā “Adam” – stohastisks gradienta bāzēts optimizētājs. Ir iespēja arī mainīt maksimālo iterāciju skaitu.[[18]](#footnote-18) Šajos eksperimentos tiks mainīts hiperparamets – regularizaton, kas norāda mācīšanās ātrumu.[[19]](#footnote-19) Ieejas slānis tiks norādīts kā ieejas pazīmju skaits tātad – 6 (neskaita klasi). Izejas slānis ir noteikts 6, jo tas ir klašu skaits. Slēpto slāņu skaits eksperimentēšanas laikā tiks mainīts vairākas reizes eksperimentējot. (att. 1.38)

Graphical user interface, application

Description automatically generated

*att. 1.38 Neironu tīkli rīks*

### **Testa un apmācību datu kopas**

Testa un apmācību datu kopas tiks izmantotas tādas pašas kā kNN algoritmā ( skatīt III

daļas 1. nodaļas 1.2 apakšnodaļu)

* 1. **Hiperparametru vērtības eksperimentos ( tab 1.4)**

|  |  |  |  |
| --- | --- | --- | --- |
| N.p.k | Testa/apmācību dati | Mācīšanās ātrums | Slēpto slāņu skaits |
| 1. | Apmācību | 0,0001 | 3 |
| 2. | Apmācību | 0,005 | 2 |
| 3. | Apmācību | 0,1 | 4 |
| 4. | Testa | 0,0001 | 3 |
| 5. | Testa | 0,005 | 2 |
| 6. | Testa | 0,1 | 4 |

*tab. 1.4. Hiperparametru vērtības*

Eksperimentu iegūtie rezultāti ar apmācību datiem (att. 1.39, att. 1.40, att. 1.41). Eksperimentu iegūtie rezultāti ar testa datiem (att. 1.42, att 1.43, att. 1.44).

**Text

Description automatically generated with low confidence**

*att. 1.39 1.eksperiments*

*Graphical user interface

Description automatically generated*

*att. 1.40 2.eksperiments*

**

*att. 1.41 3.eksperiments*

**

*att. 1.42 4.eksperiments*

**

*att. 1.43 5.eksperiments*

**

*att. 1.44 6.eksperiments*

### **Secinājumi**

Var secināt to, ka neatkarīgi no tā vai modelis ir apmācīts vai nē statiskie vērtējumi īpaši nemainījās. Vislabākais rezultāts tika iegūts 4. eksperimentā, kad tika izmantoti testa dati un 3 slēptie slāņi, līdz ar to tika iegūts augstākais precizitātes rādītājs. Visaugstākā jūtīguma metrika tika novērota pirmajā eksperimentā, kad tika izmantoti apmācību dati un vismazākais mācīšanās ātrums. Var secināt, ka palielinoties mācīšanas ātrums samazinās modeļa veiktspēja visos statiskajos parametros. Tātad neironu tīklos ir svarīgi izvēlēties mazāku mācīšanās ātrumu, lai iegūtu pēc iespējas labāku modeļa veiktspēju. Kā arī ir nepieciešams izvēlēties pareizus ieejas, izejas un slēptos slāņus.

# **Izmantotie avoti**

[1] <https://www.kaggle.com> [ skatīts 22.05.2022]

[2] <https://www.wikipedia.org/> [skatīts 22.05.2022]

[3] Grāmata – “Stars and Galaxies”, autori - Michael A. Seeds un Dana Backman

[4] <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022]

[5] <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022]

[6] <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022]

[7] <https://orangedatamining.com/widget-catalog/unsupervised/distances/> [skatīts 24.05.2022]

[8] <https://en.wikipedia.org/wiki/WPGMA> [skatīts 24.05.2022]

[9] <https://orangedatamining.com/widget-catalog/unsupervised/hierarchicalclustering/> [skatīts 24.05.2022]

[10] <https://orangedatamining.com/widget-catalog/unsupervised/kmeans/> [skatīts 24.05.2022]

[11] <https://estudijas.rtu.lv/pluginfile.php/2928885/mod_resource/content/2/4_2_temas_konspekts_2019.pdf> [skatīts 25.05.2022]

[12] <https://orangedatamining.com/widget-catalog/model/knn/> [skatīts 25.05.2022]

[13] <https://orangedatamining.com/widget-catalog/evaluate/testandscore/>[skatīts 25.05.2022]

[14] <https://orangedatamining.com/widget-catalog/evaluate/testandscore/> [skatīts 25.05.2022]

[15] <https://estudijas.rtu.lv/pluginfile.php/3227426/mod_resource/content/2/4_4_temas_konspekts_2022.pdf> [skatīts 25.05.2022]

[16] <https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/> [skatīts 25.05.2022]

[17] <https://orangedatamining.com/widget-catalog/model/svm/> [skatīts 25.05.2022]

[18] <https://orangedatamining.com/widget-catalog/model/neuralnetwork/> [skatīts 25.05.2022]

[19] <https://estudijas.rtu.lv/pluginfile.php/3241895/mod_resource/content/0/Neironu_tikli_Orange.pdf> [skatīts 25.05.2022]

1. <https://www.kaggle.com> [ skatīts 22.05.2022] [↑](#footnote-ref-1)
2. <https://www.wikipedia.org/> [skatīts 22.05.2022] [↑](#footnote-ref-2)
3. Grāmata – “Stars and Galaxies”, autori - Michael A. Seeds un Dana Backman [↑](#footnote-ref-3)
4. <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022] [↑](#footnote-ref-4)
5. <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022] [↑](#footnote-ref-5)
6. <https://www.kaggle.com/datasets/deepu1109/star-dataset> [ skatīts 22.05.2022] [↑](#footnote-ref-6)
7. <https://orangedatamining.com/widget-catalog/unsupervised/distances/> [skatīts 24.05.2022] [↑](#footnote-ref-7)
8. <https://en.wikipedia.org/wiki/WPGMA> [skatīts 24.05.2022] [↑](#footnote-ref-8)
9. <https://orangedatamining.com/widget-catalog/unsupervised/hierarchicalclustering/> [skatīts 24.05.2022] [↑](#footnote-ref-9)
10. <https://orangedatamining.com/widget-catalog/unsupervised/kmeans/> [skatīts 24.05.2022] [↑](#footnote-ref-10)
11. <https://estudijas.rtu.lv/pluginfile.php/2928885/mod_resource/content/2/4_2_temas_konspekts_2019.pdf> [skatīts 25.05.2022] [↑](#footnote-ref-11)
12. <https://orangedatamining.com/widget-catalog/model/knn/> [skatīts 25.05.2022] [↑](#footnote-ref-12)
13. <https://orangedatamining.com/widget-catalog/evaluate/testandscore/> [skatīts 25.05.2022] [↑](#footnote-ref-13)
14. <https://orangedatamining.com/widget-catalog/evaluate/testandscore/> [skatīts 25.05.2022] [↑](#footnote-ref-14)
15. <https://estudijas.rtu.lv/pluginfile.php/3227426/mod_resource/content/2/4_4_temas_konspekts_2022.pdf> [skatīts 25.05.2022] [↑](#footnote-ref-15)
16. <https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/> [skatīts 25.05.2022] [↑](#footnote-ref-16)
17. <https://orangedatamining.com/widget-catalog/model/svm/> [skatīts 25.05.2022] [↑](#footnote-ref-17)
18. <https://orangedatamining.com/widget-catalog/model/neuralnetwork/> [skatīts 25.05.2022] [↑](#footnote-ref-18)
19. <https://estudijas.rtu.lv/pluginfile.php/3241895/mod_resource/content/0/Neironu_tikli_Orange.pdf> [skatīts 25.05.2022] [↑](#footnote-ref-19)