

RĪGAS TEHNISKĀ UNIVERSITĀTE

Datorzinātnes un informācijas tehnoloģijas fakultāte

2. praktiskais darbs

mācību priekšmetā

“Mākslīgā intelekta pamati”

Mašīnmācīšanās algoritmu lietojums

**https://github.com/arturskovrigo/MIP\_PD2**

Izstrādāja: Artūrs Kovrigo

St. apl. Nr. 201RDB006

Pārbaudīja:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2021./22. māc. gads

# I daļa - Datu pirmapstrāde/izpēte

Ir izvēlēta datu kopa no Kaggle, tās nosaukums ir “Pumpkin Seeds Dataset”, tajā ir dati par divu šķirņu ķirbju sēklu datiem, kas ir ņemti no Selcuk University, kurā tika no bildēm iegūtas dažādi atribūti, kas apraksta to formas, kopā ir 12 skaitliski atribūti un viens kategoriskais – mērķa atribūts.

Tajā ir 2500 datu objektu, kas iedalās divās klasēs – Çerçevelik (Turpmāk tekstā pirmā klase) un Ürgüp Sivrisi(Turpmāk tekstā otrā klase), kas ir attiecīgās sēklas šķirnes, pirmajai 1300, otrajai 1200 ieraksti. Katram datu punktam ir 12 skaitliski atribūti – Area, Perimeter, Major\_Axis\_Length, Minor\_Axis\_length, Convex\_Area, Equiv\_Diameter, Eccentricity, Solidity, Extent, Roundness, Aspect\_Ration un Compactness. Vairāk par datu kopu var lasīt šeit - <https://link.springer.com/article/10.1007/s10722-021-01226-0>

A screenshot of a computer

Description automatically generated with medium confidence

1. attēls

Pirmie ieraksti datu kopā

|  |  |  |  |
| --- | --- | --- | --- |
| Parametru raksturvērtības | | | |
| Nosaukums | Vidējais | Minimums | Maksimums |
| Area | 80658.22 | 47939 | 136574 |
| Perimeter | 1130.27 | 868.48 | 1559.45 |
| Major\_Axis\_Length | 456.60 | 320.84 | 661.91 |
| Minor\_Axis\_Length | 225.79 | 152.17 | 305.82 |
| Convex\_Area | 81508.08 | 48366 | 138384 |
| Equiv\_Diameter | 319.33 | 247.05 | 417.00 |
| Eccentricity | 0.86 | 0.49 | 0.95 |
| Solidity | 0.99 | 0.91 | 0.99 |
| Extent | 0.69 | 0.46 | 0.83 |
| Roundness | 0.79 | 0.55 | 0.94 |
| Aspect\_Ration | 2.04 | 1.14 | 3.14 |
| Compactness | 0.70 | 0.56 | 0.90 |
| 1. tabula | | | |

|  |  |
| --- | --- |
| Parametru apraksts | |
| Nosaukums | Apraksts |
| Area | Sēklas laukums |
| Perimeter | Sēklas perimetrs |
| Major\_Axis\_Length | Sēklas garums |
| Minor\_Axis\_Length | Sēklas platums |
| Convex\_Area | Izliektas, apvilktas fomas laukums |
| Equiv\_Diameter | Kvadrātsakne no sēklas laukuma reizināta ar 4 un izdalīta ar π |
| Eccentricity | Sēklas formas ekscentriskums |
| Solidity | Laukums dalīts ar Izliektas, apvilktas formas laukumu |
| Extent | Laukums dalīts ar apvilkta taisnstūra laukumu |
| Roundness | Sēklas ovālums ignorējot izkropļojumus |
| Aspect\_Ration | Sēklas garumas dalīts ar platumu |
| Compactness | Kompaktums, jeb sēklas laukums izdalīts ar apvilktas |
| 2. tabula | |

Pirmajā tabulā redzams, ka daži datu objekti ir vērtībās ap 1, kamēr citi ir virs 105 kā rezultātā tika izlemts veikt datu normalizāciju. Izvēlēts normalizēt intervālā [0,1].

Dati nevienā apskatītajā atribūtā un nevienā atribūtu pārī neatdalās pilnīgi, tas ir, visos starp klasēm ir nozīmīga pārklāšanās, bet vienlaikus, lielākajā daļā atribūtu ir arī skaidri redzamas atšķirības. Starp atsevišķiem atribūtiem, Eccentricity bija viens no labākajiem, savukārt no pāriem labākie bija Compactness atkarībā no Solidity, Aspect\_Ration no Minor\_Axis\_Length un Eccentricity no Roundness. Šo sakarību vizualizācijas redzamas 2.;3.;4. un 5. attēlos

A picture containing text, plot, screenshot, line

Description automatically generated

2. attēls

A picture containing text, screenshot, colorfulness

Description automatically generated

3. attēls

A red and blue dots

Description automatically generated with low confidence

4. attēls

A screenshot of a computer screen

Description automatically generated with low confidence

5. attēls

# II daļa - Nepārraudzītā mašīnmācīšanās

***k-Means***

K-Means ir 4 hiperparametri – klasteru skaits, kurš nosaka, cik būs klasteri; re-runs, kurš nosaka, cik reizes mēģināt klasterizēt līdz paliek pie esošās maksimālās silueta koeficienta vērtības; maksimālais iterāciju skaits, kurš nosaka maksimālo centroīdu pārrēķināšanas skaitu; inicializācijas metode, kas ir vai nu nejauša, vai KMeans++, kura pēc pirmā centroīda nejaušas izvēles, otro izvēlās atkarībā no tā attāluma līdz pirmajam – jo tālāk, jo lielāka iespēja izvēlēties.

To izmantojot maksimālā silueta vērtība sasniegta pie diviem klasteriem, kas sakrīt ar mērķa atribūta klašu skaitu, taču tas ir diezgan mazs – 0.3. Apskatot sadalījuma grafiku var novērot, ka pirmais klasteris sastāv no 83.6% no otrās klases, un otrais klasteris 75.5% no pirmās klases. Šos klasterus izmantojot klasifikācijai pareizi klasificēti būtu 78.7% ierakstu

|  |  |  |
| --- | --- | --- |
| Nepārraudzītā mašīnmācīšanās – k-Means | | |
| Maksimālais iterāciju skaits | Klasteru skaits | Silueta koeficients |
| 5 | 2 | 0.307 |
| 3 | 0.290 |
| 4 | 0.269 |
| 5 | 0.239 |
| 25 | 2 | 0.308 |
| 3 | 0.288 |
| 4 | 0.271 |
| 5 | 0.240 |
| 125 | 2 | 0.308 |
| 3 | 0.289 |
| 4 | 0.271 |
| 5 | 0.239 |
| 3. tabula | | |

A picture containing text, screenshot, rectangle, diagram

Description automatically generated

6. attēls

Klašu sadalījums klasteros

|  |  |
| --- | --- |
| Hierarhiskā klasterēšana | |
| Attāluma metrika | Precizitāte |
| Eiklīda | 80.88% |
| Manhatanas | 70.56% |
| Kosīnusa | 72.30%\* |
| Spīrmena | 84.84% |
| Absolūtā Spīrmena | 83.32%\* |
| Pīrsona | 77.96% |
| Hamminga | N/A\*\* |
| 4. tabula | |

\*Pirmie klasteri bija pāris datu objektu lieli, precizitāte iegūta palielinot klasteru skaitu līdz ir 2 lieli klasteri, un mazos klasterus uzskatot par nepareiziem

\*\* Hamminga attālumam neizdevās iegūt precizitāti, jo palielinot klasteru skaitu par līdz 15, bija viens izteikti liels klasteris un pārējie mazi, līdz brīdim, kad atlikušais sasniedz tik pat mazu apjomu, kā pārējie

***Hierarhiskā klasterēšana***

Hierarhiskā klasterēšana izmanto divus parametrus – klasteru skaitu, kas atkal nosaka, cik grupās klasterizēt, un attāluma aprēķina metriku. Vislabākā izrādījās Spīrmena attāluma metrika, ar kuru pirmajā klasterī bija 80.32%, jeb 1220 pirmās klases ieraksti, un otrajā klasterī 91.85%, jeb 901 otrās klases ieraksts, kas kopā dotu 84.84% precizitāti, ja šos klasterus izmantotu klasifikācijai.

A screenshot of a graph

Description automatically generated with low confidence

7. attēls

Klašu sadalījums klasteros

A screenshot of a graph

Description automatically generated with low confidence

8. attēls

Koka vizualizācija

# III daļa - Pārraudzītā mašīnmācīšanās

Tika izvēlēts nejauša meža algoritms, SVM un neirālais tīkls, lai mēģinātu atrast kādu veidu, kā uzticami klasificēt datus. Katrai metodei veikta eksperimentācija ar dažādām hiperparametru kombinācijām, lai iegūtu maksimālo precizitāti.

***Neirālie Tīkli***

Daudzslāņu mākslīgo neironu tīkli būtībā sastāv no 3 dažādiem slāņu tipiem – ieejas slānis, slēptais slānis/slāņi, izvades slānis. Ieejas slānis šajā kontekstā ir neironu kopums, kurā katrs neirons atbilst noteiktam atribūtam. Slēptie slāņi sastāv no vienā virzienā savstarpēji savienotiem neironiem līdz tie nonāk līdz izejas slānim. Salīdzinot izejas slāņa rezultātus ar vēlamo vērtību attiecīgi tiek koriģeti svari neironos ar mērķi tuvināt tīkla pareģojumu šai vēlamajai vērtībai.

Orange rīkā ir pieejami 5 galvenie hiperparametri, tīkla struktūra, kas sastāv no neironu skaita katrā slēptajā slānī, slāņu skaita, aktivācijas funkcijas, solvers, jeb optimizētājs, kurš nosaka kā notiek tīkla trenēšanās un iterāciju, jeb epohu skaits, kas nosaka, cik reizes visi dati tīklam tiek padoti.

No šiem vislabāko precizitāti sasniedza tīkls ar 500 slēptajiem neironiem katrā no 2 slēptajiem slāņiem, izmantojot 100 iterācijas un identitātes aktivācijas funkciju. Vēl datos varēja novērot pārtenēšanos, 3. eksperimentā palielinot iterāciju skaitu samazinājās precizitāte uz testa datiem.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Neironu skaits slēptajos slāņos | Slēpto slāņu skaits | Iterāciju skaits | Aktivācijas funkcija | AUC | CA | F1 | Precision | Recall |
| 100 | 2 | 100 | ReLu | 0.939 | 0.860 | 0.860 | 0.860 | 0.860 |
| 500 | 2 | 100 | ReLu | 0.926 | 0.868 | 0.868 | 0.868 | 0.868 |
| 500 | 2 | 500 | ReLu | 0.914 | 0.856 | 0.856 | 0.857 | 0.856 |
| 100 | 3 | 100 | ReLu | 0.926 | 0.860 | 0.860 | 0.860 | 0.860 |
| 500 | 3 | 100 | ReLu | 0.907 | 0.844 | 0.846 | 0.844 | 0.844 |
| 500 | 2 | 100 | tanh | 0.936 | 0.868 | 0.868 | 0.869 | 0.868 |
| 500 | 2 | 100 | Loģistiskā | 0.933 | 0.860 | 0.859 | 0.862 | 0.860 |
| 500 | 2 | 100 | Identitātes | 0.934 | 0.884 | 0.884 | **0.884** | 0.884 |
| 1000 | 2 | 100 | Identitātes | 0.933 | 0.868 | 0.868 | 0.868 | 0.734 |
| 5. tabula | | | | | | | | |

***Nejauša meža algoritms***

Nejauša meža algoritms, uzģenerē lietotāja izvēlētu skaitu lēmumu kokus nejaušām datu apakškopām, un par pareizu uzskata vispoppulārāko variantu starp izveidoto koku prognozēm. Tam ir tikai viens svarīgākais hipermarametrs – koku skaits, lielāks skaits ir vairāk koki, ir mazāk svārstības, taču eksperimentējot arī ar 100 un 1000 kokiem, tās bija diezgan nozīmīgas, un palaižot atkārtoti nozīmīgi mainījās rezultāti. Labākais ko ar šo algoritmu izdevās sasniegt ir 87.2% precizitāte, un pateicoties lielajām svārstībām, tas tika sasniegts gan ar 10, gan 100 koku eksperimentiem

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Koku skaits | AUC | CA | F1 | Precision | Recall |
| Random fores | 10 | 0.921 | 0.872 | 0.872 | **0.872** | 0.872 |
| 50 | 0.929 | 0.868 | 0.868 | 0.868 | 0.868 |
| 100 | 0.926 | 0.872 | 0.872 | **0.872** | 0.872 |
|  | 1000 | 0.925 | 0.864 | 0.864 | 0.864 | 0.864 |
| 6. tabula | | | | | | |
| ***SVM*** | | | | | | |

SVM algoritms balstās uz robežu novilkšanu starp klasēm izmantojot klašu ekstrēmus. Tā vietā, lai izvēlētos nejaušu līniju, kura atdala maksimāli daudz, tiek izmantoti atbalsta vektori, kas atrod, kur beidzās viena kopa un kur sākās otra, un robežu ieliek tieši pa vidu tiem. Pieejamie hiperparametri ir SVM tips, starp kuriem galvenāatšķirība ir vektoru skaits, v-SVM ir lielāks skaits, kas samazina variāciju, kodols, kurš maina izmantotās funkcijas, oriģinālais ir lineārs, taču bieži labāk strādā citas funkcijas, šajos eksperimentos paturēts Orange noklusējuma variants – RBF, jeb exp(-G|X-Y|2), kā arī iterāciju limits.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SVM | vSVM; Iterāciju limits = 1000 | 0.823 | 0.716 | 0.716 | 0.716 | 0.428 |
| SVM; Iterāciju limits = 1000 | 0.929 | 0.868 | 0.868 | **0.868** | 0.868 |
| vSVM; Iterāciju limits = 100000 | 0.866 | 0.784 | 0.783 | 0.784 | 0.784 |
| SVM; Iterāciju limits = 100000 | 0.929 | 0.868 | 0.868 | **0.868** | 0.868 |
| 7. tabula | | | | | | |

***Secinājumi par pārraudzīto mašīnmāčišanos***

Kopumā labākais sniegums bija neirālajam tīklam 500 neironiem divos slēptajos slāņos, 100 ierācijām un identitātes aktivācijas funkciju, un tas bija 88.4%, kas tomēr ir tālu no datu autoru veiktajā petījumā sasniegtajiem 92.77% ar SVM un 92.31% ar neirālo tīklu.

# Izmantotie informācijas avoti

KOKLU, M., SARIGIL, S., & OZBEK, O. (2021). The use of machine learning methods in classification of pumpkin seeds (Cucurbita pepo L.). Genetic Resources and Crop Evolution, 68(7), 2713-2726. Doi: <https://doi.org/10.1007/s10722-021-01226-0>

Orange Data Mining (2015). *Orange Visual Programming* (skatīts 2023, 11. maijā).<https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/index.html>

Graupe, D. *Principles Of Artificial Neural Networks (3rd Edition)*. Čikāga: World Scientific Publishing Company, 2013. 363 lpp. ISBN 978-981-4522-73-1

# Pielikums

A picture containing text, diagram, line, map

Description automatically generated

Vispārējs darba izvietojums Orange vidē