Mākslīgā intelekta pamati

2. Praktiskais darbs

Mašīnmācīšanās algoritmu lietojums

Github - https://github.com/Synhro2001/M-ksl-g-intelekta-pamati.git

Vadims Krasovskis

201DDB005

**I daļa - Datu pirmapstrāde/izpēte**

**Datu kopas apraksts**

**Map

Description automatically generated**

* 1. Attēls

**Datu kopas nosaukums, avots, izveidotājs un/vai īpašnieks.**

Šajā praktiskajā darbā tiek pielietota datu kopa ar nosaukumu **Date Fruit Datasets,** kura ir ņemta <https://www.muratkoklu.com/datasets/> . Datu kopa izveidotājs ir Murat Koklu (Academician at Selcuk University, Konya, Konya, Turkey).

**Datu kopas problēmsfēras apraksts.**

Visā pasaulē audzē milzīgu skaitu augļu, no kuriem katram ir dažādi veidi. Faktori, kas nosaka augļu veidu, ir izskata pazīmes, piemēram, krāsa, garums, diametrs un forma.

Šī pētījuma mērķis ir klasificēt datuma augļu sugas, proti: Barhi, Deglet Nur, Sukkari, Rotab Mozafati, Rutana, Safavi un Sagai, izmantojot trīs dažādas mašīnmācīšanās metodes.

**Veids, kā datu kopa tika savākta.**

Computer Vision System (CVS) izveidoja 898 attēlus ar septiņiem dažādu veidu dateļu augļiem. Kopumā no šiem attēliem, izmantojot attēlu apstrādes metodes, tika iegūtas 34 pazīmes, tostarp morfoloģiskās pazīmes, forma un krāsa.

(Atsauce no kurienes ir paņemta informācijas KOKLU, M., KURSUN, R., TASPINAR, Y. S. and CINAR, I. (2021). Classification of Date Fruits into Genetic Varieties Using Image Analysis. Mathematical Problems in Engineering, Vol.2021, Article ID: 4793293. <https://doi.org/10.1155/2021/4793293> )

**Datu kopas satura apraksts.**

**Datu objektu skaits datu kopā.**

Datu kopā objektu skaits ir 898.

Datu kopā klašu skaits ir 7 ( BERHI ( 65 ), DEGLET ( 98 ), DOKOL ( 204 ), IRAQI ( 72 ),

ROTANA ( 166 ), SAFAVI ( 199 ), SOGAY ( 94 ) ).

Datu kopā atribūtu skaits ir 34 (AREA , PERIMETER, MAJOR\_AXIS, MINOR\_AXIS, ECCENTRICITY, EQDIASQ, SOLIDITY, CONVEX\_AREA, EXTENT, ASPECT\_RATIO, ROUNDNESS, COMPACTNESS, SHAPEFACTOR\_1, SHAPEFACTOR\_2, SHAPEFACTOR\_3, SHAPEFACTOR\_4, MeanRR, MeanRG, MeanRB, StdDevRR, StdDevRG, StdDevRB, SkewRR, SkewRG, SkewRB, KurtosisRR, KurtosisRG, KurtosisRB, EntropyRR, EntropyRG, EntropyRB, ALLdaub4RR, ALLdaub4RG, ALLdaub4RB ).

**Bet praktiskajā darbā tiek izmantoti:**

Datu kopā objektu skaits ir 264.

Datu kopā klašu skaits ir 3 ( DEGLET ( 98 ), IRAQI ( 72 ), SOGAY ( 94 ) ). Šīm klasēm ir laba atdalāmība histogramā.

Datu kopā atribūtu skaits ir 8 ( AREA , PERIMETER, MAJOR\_AXIS, MINOR\_AXIS, ECCENTRICITY, EXTENT, ROUNDNESS, COMPACTNESS ) Es paņēmu tādus atribūtus, kuriem ir morfoloģiskās pazīmes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Atribūtus | Diapazons | Vērtību tips | Atribūtu apraksts |
| 1 | Area | No 1987 līdz 546063 | Skaitlisks | Atgriež pikseļu skaitu augļa robežās |
| 2 | Perimeter | No 911.8280 līdz 2811.9971 | Skaitlisks | Aprēķina apkārtmēru, aprēķinot attālumu starp pikseļiem ap augļa apmalēm |
| 3 | Major axis | No 336.7227 līdz 1222.7230 | Skaitlisks | Dod garāko līniju, kādu var novilkt uz rīsu augļa, t.i. attālums līdz galvenajai asij. |
| 4 | Minor axis | No 2.2832 līdz 766.4536 | Skaitlisks | Īsākā līnija, ko var novilkt uz rīsu augļa t.i. neliels attālums starp asīm, dod. |
| 5 | Eccentricity | No 0.3448 līdz 1.0000 | Skaitlisks | Mēra, cik apaļa ir elipse, kurai ir tādi paši momenti kā auglim. |
| 6 | Roundness | No 0.0048 līdz 0.9773 | Skaitlisks | Aprēķināta pēc formulām |
| 7 | Equivalent diameter | No 50.2984 līdz 833.8279 | Skaitlisks | Rīsu augļa diametrs ir vienāds ar apļa diametru |
| 8 | Solidity | No 0.8366 līdz 0.9974 | Skaitlisks | Izliekts apvalks līdz augļu daudzumam(PX) |
| 9 | Convex\_area | No 2257 līdz 552598 | Skaitlisks | Atgriež pikseļu skaitu mazākā izliektajā apgabalā, ko veido auglis. |
| 10 | Extent | No 0.5123 līdz 0.8562 | Skaitlisks | Atgriež augļa izveidotā laukuma attiecību pret ierobežojošā lodziņa pikseļiem |
| 11 | Aspect ratio | No 1.0653 līdz 535.5257 | Skaitlisks | Attiecība starp galvenās ass garumu un mazās ass garumu |
| 12 | Compactness | No 0.0411 līdz 0.9681 | Skaitlisks | Augļa apaļuma pēc formulam |
| 13 | Shapefactor\_1 | No 0.0017 līdz 0.6154 | Skaitlisks | Aprēķināta pēc formulas: |
| 14 | Shapefactor\_2 | No 0.0011 līdz 0.0043 | Skaitlisks | Aprēķināta pēc formulas: |
| 15 | Shapefactor\_3 | No 0.0017 līdz 0.9373 | Skaitlisks | Aprēķināta pēc formulas: |
| 16 | Shapefactor\_4 | No 0.7568 līdz 0.9995 | Skaitlisks | Aprēķināta pēc formulas: |
| 17 | Mean RR | No 30.3826 līdz 159.6494 | Skaitlisks | red color |
| 18 | Mean RG | No 41.0480 līdz 166.1354 | Skaitlisks | green color |
| 19 | Mean RB | No 44.2569 līdz 148.2114 | Skaitlisks | blue color |
| 20 | Std. dev RR | No 7.3038 līdz 48.5712 | Skaitlisks | red color |
| 21 | Std. dev RG | No 8.6557 līdz 45.5107 | Skaitlisks | green color |
| 22 | Std. dev RB | No 5.1486 līdz 42.4228 | Skaitlisks | blue color |
| 23 | Skew RR | No -1.7242 līdz 3.2236 | Skaitlisks | red color |
| 24 | Skew RG | No -1.8344 līdz 3.6971 | Skaitlisks | green color |
| 25 | Skew RB | No -1.0291 līdz 3.0923 | Skaitlisks | blue color |
| 26 | Kurtosis RR | No 1.7082 līdz 26.1711 | Skaitlisks | red color |
| 27 | Kurtosis RG | No 1.6076 līdz 26.7367 | Skaitlisks | green color |
| 28 | Kurtosis RB | No 1.7672 līdz 32.2495 | Skaitlisks | blue color |
| 29 | Entropy RR | No -109122420736 līdz -162731632 | Skaitlisks | Entropy of the current channel |
| 30 | Entropy RG | No -92616974336 līdz -562772672 | Skaitlisks | Entropy of the current channel |
| 31 | Entropy RB | No -87471767552 līdz -437043456 | Skaitlisks | Entropy of the current channel |
| 32 | All Daub4 RR | No 15.1911 līdz 79.8289 | Skaitlisks |  |
| 33 | All Daub4 RG | No 20.5247 līdz 83.0649 | Skaitlisks |  |
| 34 | All Daub4 RB | No 22.1300 līdz 74.1046 | Skaitlisks |  |

1.1 tabula

Nav informācijas par All Daub4 !

Avoti, no kurienes informācija ņemta.

* <https://research.ijcaonline.org/volume83/number15/pxc3892915.pdf>
* <http://thejaps.org.pk/docs/v-24-1/47.pdf>
* <https://agro.icm.edu.pl/agro/element/bwmeta1.element.agro-20f5b343-f1df-4785-93b1-1f28c3d79222/c/Wheat_class.pdf>

**Table

Description automatically generated with medium confidenceStatistiskie vidējie rādītāji, kas iegūti no dateļu augļiem**

* 1. Attēls

**Table

Description automatically generated with medium confidence**

* 1. Attēls

**Graphical user interface, text, application, email

Description automatically generatedPētījumā izmantotie dateļu augļi un to īpašības ( Augļi, kuri izmantoti šajā darbā).**

1.4 Attēls

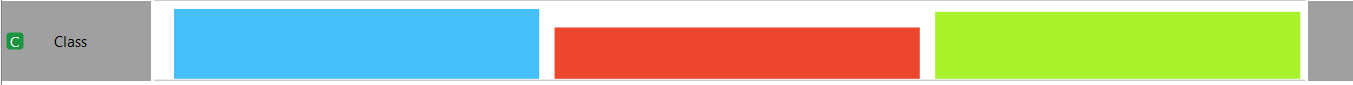
**Vai klases datu kopā ir līdzsvarotas, vai dominē viena klase (vai vairākas klases)? Tas tiek noteikts, spriežot pēc tā, cik datu objektu pieder katrai kopai.**

**Datu kopa, kur ir visi dati un visas klases.**

****

1.5 Attēls

**Mana datu kopa, kuru es izmantoju darbā.**

****

1.6 Attēls

Secinājums: Datu kopa, kur visas klases nav sabalansētas, jo ir trīs klases, kur ir vairāk datu nekā pārējiem, tas ir attēlots pirmajā attēlā.

Datu kopa, kura izmantota šajā praktiskajā darbā, ir labi sabalansēta, tas ir attēlots otrajā attēlā.

**Vai datu vizuālais atspoguļojums ļauj redzēt datu struktūru? Runa ir par to, vai datu objekti, kuri pieder dažādām klasēm, ir skaidri atdalāmi.**

**Izkliedes diagramma**

**Chart, scatter chart

Description automatically generated**

1.7 Attēls

Secinājums : Y asi – Minor\_Axes , X asi – AREA. Šajā diagrammā ir parādīts, ka atdalījums starp 3 klasēm ir ne īpaši labs. Klase DEGLET un SOGAY saplūst kopā, bet klase IRAQI ir labi atdalīta no pārējiem.

**Cik datu grupējums ir iespējams identificēt, pētot datu vizuālo atspoguļojumu? Runa ir par to, vai ir kaut cik atdalāmi datu grupējumi, ja gadījumā dažādu klašu datu objekti saplūst kopā.**

**Vai identificētie datu grupējumi atrodas tuvu viens otram vai tālu viens no otra?**

**Izkliedes diagramma**

**(Y – MINOR\_ASIS , X – MAJOR\_ASIS )**

**Chart, scatter chart

Description automatically generated**

1.8 Attēls

**Izkliedes diagramma**

**(Y – COMPACITINESS , X – EXTENT )**

**Chart, scatter chart

Description automatically generated**

1.9 Attēls

**Izkliedes diagramma**

**(Y – ECCENTIRICITY , X – AREA )**

**Background pattern

Description automatically generated**

1.10 Attēls

Secinājums: Apskatot diagrammas, redzams, ka klase IRAQI ir dominējoša. Lai uzzināt, kuras klases ir dominējošas, bija izmantots (color regresion ).

Ir redzams, ka DEGLET un SOGAY grupas atrodas tuvāk viens otram nekā grupa IRAQI.

**Chart

Description automatically generatedFeature Statistics**

1.11 Attēls

Lai uzzināt vidējo vērtību un dispersijas vērtību bija izmantota (Feature Statistics).

**Pilnai datu kopai histogramma ( AREA atribūts )**

**Chart, histogram

Description automatically generated**

1.12 Attēls

**Datu kopai histogramma, kura ir izmantota (AREA atribūts )**

**Chart, line chart, histogram

Description automatically generated**

1.13 Attēls

**Pilnai datu kopai histogramma (EXTENT atribūts )**

**Chart, histogram

Description automatically generated**

1.14 Attēls

**Datu kopai histogramma, kurai ir izmantots (EXTENT atribūts )**

**Chart, histogram

Description automatically generated**

1.15 Attēls

Aplūkojot histogrammu, varam secināt, ka sadalījums nav īpaši labs.

**II daļa – Nepārraudzītā mašīnmācīšanās**

**Hierarhiskās klasterizācijas algoritmam ir jāveic vismaz 3 eksperimenti, brīvi mainot hiperparametru vērtības, un analizējot algoritma darbību.**

Single linkage - aprēķina attālumu starp diviem tuvākajiem klasteru elementiem.

Average linkage - aprēķina vidējo attālumu starp diviem tuvākajiem klasteru elementiem.

**Chart, scatter chart

Description automatically generated**

2.1 Attēls

**Hierarchical Clustering**

**Chart, box and whisker chart

Description automatically generated**

2.2 Attēls

Secinājums: Lai izveidot dendogrammu, tika izmantots Eikildiāna distance. Pārbaudām klasterus pēc attāluma, ja starp klasteriem ir liels attālums, tas tiks uzskatīts par atšķirīgu objektu savienību. Izmantojot Single linkage, varam secināt, ka dati vislabāk klastirizējas trīs klasteros. Pirmajā klasterī dominē SOGAY objekti, otrajā klasterī dominē DEGLET objekti, bet trešajā visi pārējie objekti.

**Hierarchical Clustering**

**Chart

Description automatically generated**

2.3 Attēls

Ja mēģināt izveidot 2 klasterus, var secināt, ka pirmajā klasterī dominē objekti SOGAY, bet otrajā visi pārējie.

**Hierarchical Clustering**

**Chart

Description automatically generated**

2.4 Attēls

Izmantojot Average Linkage, dati mainījušies. Pirmajā klasterī SOGAY objekti palika tāpat kā bija, bet otrajā klasterī dominē DEGLET objekti. Trešajā klasterī dominē visi pārējie objekti.

**Chart, bar chart

Description automatically generated**

2.5 Attēls

Ja objekti tiek sagrupēti divos klasteros, var secināt, ka pirmajā klasterī, kā tas bija, dominē SOGAY objekti, bet otrajā paliek visi pārējie objekti.

**K-vidējo algoritmam ir jāveic eksperimenti ar vismaz 5 k vērtībām, jāaprēķina Silhouette Score, un jāanalizē algoritma darbība.**

Par k-Means algoritmu. Pirmais centrs tiek izvēlēts nejauši, nākamie tiek izvēlēti no nākamajiem punktiem ar varbūtību, kas ir proporcionāla attāluma kvadrātam no tuvākā centra.

Re-runs - cik reizes algoritms tiek palaists no nejaušām sākuma pozīcijām.

Maximum iterations - maksimālais iterāciju skaits katru reizi, kad tiek palaists algoritms.

**Graphical user interface

Description automatically generated**

2.6 Attēls

Tika izmantota Sihoulette Score, lai noskaidrotu, cik kopu vislabāk atbilst. Izmantojot jebkurus hiperparametrus, var secināt, ka visi sadalījumi parāda ne pārāk labus rezultātus. Silueta koeficients parāda, ka vislabākais ir divu klasteru sadalījums, jo koeficients ir tuvāk 1. Ja paskatās uz citiem klasteriem, var redzēt, ka koeficients tuvojas 0.

Izmantoju Select Rows, lai interaktīvi izpētītu kopu un displeju izkliedes diagrammā.

1)Graphical user interface, text, application

Description automatically generated

2.7 Attēls

2) Graphical user interface, text, application

Description automatically generated

2.8 Attēls

3) Graphical user interface

Description automatically generated with medium confidence

2.9 Attēls

1. **Izkliedes diagrammā par DEGLET dati.**

Chart, scatter chart

Description automatically generated

2.10 Attēls

1. **Izkliedes diagrammā par IRAQI dati**

Chart, scatter chart

Description automatically generated

2.11 Attēls

1. **Izkliedes diagrammā par SOGAY dati**

Chart, scatter chart

Description automatically generated

2.12 Attēls

**Silhouette Plot**

Chart, histogram

Description automatically generated

2.13 Attēls

Rezultāta analīzē, mēs redzam ( 2.13 attēls ), ka klases DEGLET sadalījums ir labs, jo dominē pozitīvā puse. Var secināt, ka dažas vērtības atšķiras, dažas ir līdzīgas citu klašu vērtībām.

Graphical user interface

Description automatically generated

2.14 Attēls

Klases IRAQI klasificēts ļoti labi, jo dominē pozitīvā puse. Tas nozīme, ka vērtības klase IRAQI atšķiras no pārējo klašu vērtībām. ( Skatīties uz 2.14 attēls ).

Chart

Description automatically generated

2.15 Attēls

Izmantojot Siluetu diagramma, var secināt, ka datu objekti sadalīti diezgan labi, jo daudzi objekti klasificētie ar plusu. Mēs varām redzēt izklaides diagrammā, ka klasēs SOGAY dati ir nepareizi klasificēti, jo dominēja vairāk uz negatīvo pusi. ( Skatīties uz 2.15 attēls ). Balstoties uz algoritmiem var secināt, ka vieglāk bija izpildīt darbu ar K-vidēju algoritmu.

**Chart, scatter chart

Description automatically generatedTas redzams izklaides diagrammā**.

Piemēram izmantojām izklaides diagrammu, kur Y – Minor\_Axes un X – AREA, varam secināt, ka dati IRAQI atdalīti ļoti labi, jo IRAQI klases vērtība ļoti atšķiras no DEGLET un SOGAY klašu vērtības. Bet klases DEGLET un SOGAY punkti ir samaisīti kopā, un tos ir ļoti grūti atdalīt citu no cita, jo vērtības ir ļoti līdzīgas.

2.16 Attēls

**Balstoties uz abu algoritmu darbības analīzi, ir jādod studenta secinājumi par to, vai datu kopā esošās klases ir labi vai slikti atdalāmas.**

Analizējot abus algoritmus var secināt, ka ar k-Means algoritmu klases labāk atdalīti nekā ar Hierarchical clustering algoritmu to var redzēt jā analizēt 2.13 – 2.15 attēlus ar 2.2 – 2.5 attēlus.

**III daļa – Pārraudzītā mašīnmācīšanās**

**Īss apraksts (1/3 no A4 lapas) diviem brīvi izvēlētajiem algoritmiem un to izvēles motivācijai (izņemot mākslīgo neironu tīklu), norādot arī atsauces uz izmantotajiem informācijas avotiem.**

Pirmais algoritms SVM ( Support Vector Machine ). Šis algoritms tika izvēlēts, jo tas ir viens no populārākajiem uzraudzītās mācīšanās algoritmiem, ko izmanto gan klasifikācijas, gan regresijas problēmām. SVM algoritmu vispirms izmanto, lai risināt klasifikācijas uzdevumus mašīnmācībā. Kā strādā algoritms SVM? Vispirms, vajag izveidot hiperplānu , kas var sadalīt n-dimensiju telpu klasēs ( lai ievietotu jauno datu punktu pareizajā kategorijā ). Ko dara SVM ? SVM algoritms izvēlas ekstremālos punktus, lai izveidot hiperplānus.

**(**[**https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm**](https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm)**)**

Otrais algoritms kNN (K – Nearest Neighbour ). Šis algoritms tika izvēlēts, tas ir viens no vienkāršākajiem mašīnmācīšanās algoritmiem, kuru pamatā ir uzraudzīta mācību metode. Šo algoritmu visbiežāk izmanto klasifikācijas problēmām. Izmantojot kNN, mēs varam viegli noteikt konkrētas datu kopas kategoriju vai klasi. Kāds ir šī algoritma mērķis? Izveidot labāko lēmumu līniju vai robežu, kas var sadalīt n-dimensiju telpu klasēs. Mēs to darām, lai ievietotu jauno datu punktu pareizajā kategorijā. Mēs dabūjam hiperplānu. Hiperplāns ir labākā lēmuma robeža.

**Katram algoritmam ir jāapraksta Orange rīkā pieejamie hiperparametri un to nozīme.**

**Neural Network (3.1 Attēls )**

Neurons per hidden layer: definēts kā i-tais elements, kas apzīmē neironu skaitu i-tajā slēptajā slānī.

Activation: Bija izvēlēts ReLu, tā ir rektificēta lineārās identitātes funkcija.

Solver: Bija izvelēts Adam, tas ir stohastiskā gradienta optimizētājs.

Regularization: soda parametrs.

Max iterations: iterāciju skaits.

**Graphical user interface, application

Description automatically generated**

3.1 Attēls

**SVM (3.2 Attēls)**

Cost: zaudējumu sods, tas tiek izmantots klasifikācijas un regresijas problēmām.

Regression loss epsilon: Norāda attālumu no patiesajām vērtībām, kurā sods netiks saistīts ar prognozētajām vērtībām.

Kernel: ir funkcija, kas pārvērš atribūtu vietu jaunā objekta telpā.

, kur

g - gamma konstantes kodola funkcijā (ieteicamā vērtība ir 1/k, kur k ir atribūtu skaits).

c - konstantam c0 kodola funkcijā (noklusējums 0).

d - kodola pakāpes (noklusējums 3)

Optimization parametrs: skaitliskajā pielaidē iestatiet pieļaujamo novirzi no paredzamās vērtības.

**Graphical user interface, text, application

Description automatically generated**

3.2 Attēls

**kNN (k – Nearest Neighbour) algorithm (3.3 Attēls )**

Neighbors - norāda tuvāko kaimiņu skaitu, attāluma parametru (metriku) un svarus kā modeļa kritērijus.

Bija izvēlēts Eiklīda metric (“taisna līnija”, attālums starp diviem punktiem).

**Graphical user interface, application, email

Description automatically generated**

3.3 Attēls

**Informācija par testu un apmācību datu kopām**

Pēc nejaušības principa, sadalām datu kopu apmācības un testa komplektus noteiktās proporcijās. Pārbauda algoritma veiktspēju un noteikt, kurš algoritms ir labāks.

**Graphical user interface, text, application, email

Description automatically generated**

3.4 Attēls

Apskatot visus rezultātus varam secināt, ka labākie rezultāti būs SVM algoritmā.

AUC – Salīdzinot trīs algoritmus, var redzēt, ka SVM algoritma vislabākais rezultāts sanāca (0.933) . Apskatot parametru AUC, varam uzzināt divdimensiju laukumu zem visas līknes ROC.

**Datu kopas, kuras sadalītas divās kopās. ( 3.5 Attēls )**

Pirmajā kopā, datu objektu skaits ir 198, no kuriem DEGLET klasei ir 71 jeb 35.86% , IRAQI klasei ir 59 jeb 29.80 %, SOGAY klasei ir 68 jeb 34.34%.

**Chart, bar chart

Description automatically generated**

3.5 Attēls

Otrajā kopā (3.6 Attēls ) , datu objektu skaits ir 66, no kuriem DEGLET klasei ir 27 jeb 40.91%, IRAQI klasei ir 13 jeb 19.70%, SOGAY klasei ir 26 jeb 39.39%

**Chart, bar chart, waterfall chart

Description automatically generated**

3.6 Attēls

1. **Eksperimenta hiperparametri**

**Neural Network (3.1 tabula)**

|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| Neurons in hidden layers | 10, 5, 10 |
| Activation | ReLu |
| Solver | Adam |
| Regularization | 0.0001 |
| Max number of iterations | 200 |

3.1 tabula

**SVM (3.2 tabula)**

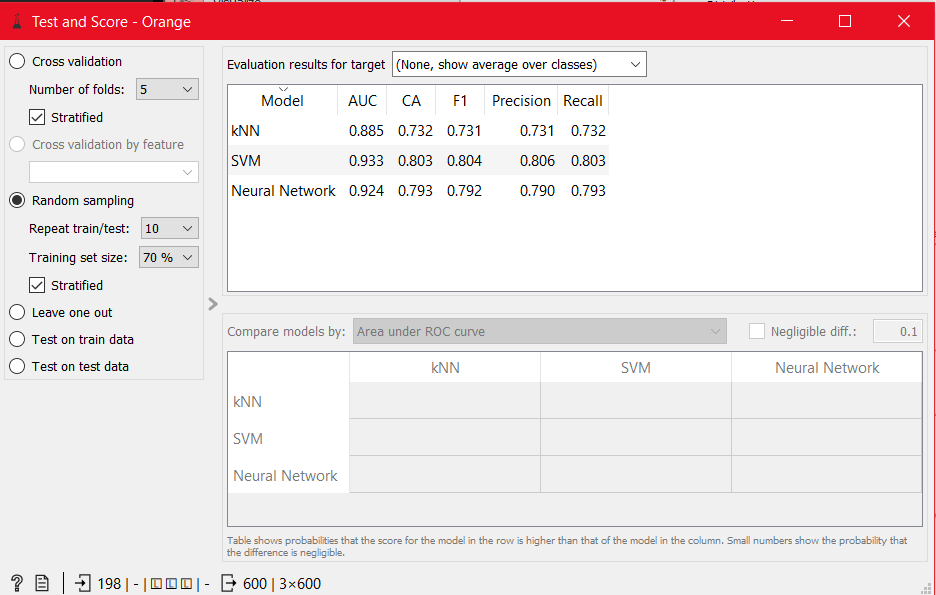
|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| SVM Cost ( C ) | 10.00 |
| Regression loss epsilon | 0.5 |
| Kernel | RBF |
| Numerical tolerance | 0.0010 |
| Iteration limit | 100 |

3.2 tabula

**kNN ( K – Nearest Neighbour ) (3.3 tabula)**

|  |  |
| --- | --- |
| **Hiperparematra nosaukums** | **Vērtība** |
| Number of neighbors | 5 |
| Metric | Euclidean |
| Weight | Uniform |

3.3 tabula

****

3.7 Attēls

Pēc pirmā eksperimenta var secināt, ka visi algoritmi sniedz diezgan labus rezultātus, bet vislabākais rezultāts parādījās SVM algoritmam, jo tā precizitāte, pēc visiem novērtēšanas metrika, ir vienāda, apmēram 0.803 – 0.806. Vissliktākais rezultāts ir kNN algoritmam, jo tā precizitāte pēc visiem novērtēšanas metrika ir vienāda, 0.731 – 0.732.

1. **Eksperimenta hiperparametri**

**Neural Network (3.4 tabula)**

|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| Neurons in hidden layers | 4, 6, 10 |
| Activation | ReLu |
| Solver | Adam |
| Regularization | 0.05 |
| Max number of iterations | 200 |

3.4 tabula

**SVM ( 3.5 tabula )**

|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| SVM Cost ( C ) | 5.00 |
| Regression loss epsilon | 0.10 |
| Kernel | RBF |
| Numerical tolerance | 0.010 |
| Iteration limit | 100 |

3.5 tabula

**kNN ( K – Nearest Neighbour ) ( 3.6 tabula)**

|  |  |
| --- | --- |
| **Hiperparematra nosaukums** | **Vērtība** |
| Number of neighbors | 10 |
| Metric | Euclidean |
| Weight | Manhattan |

3.6 tabula

Graphical user interface, text, email

Description automatically generated

3.8 Attēls

Pēc otrā eksperimenta var secināt, ka visi algoritmi sniedz labākus rezultātus nekā pirmajā eksperimentā. Vislabākais rezultāts šajā eksperimentā parādījās Neural Network algoritmam, jo tā precizitāte pēc visiem novērtēšanas metrika ir vienāda apmēram 0.827. Vissliktākais rezultāts ir kNN algoritmam, jo tā precizitāte pēc visiem novērtēšanas metrika ir vienāda 0.770 – 0.777, bet labāk nekā pirmajā eksperimentā.

1. **Eksperimenta hiperparametri**

**Neural Network (3.7 tabula )**

|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| Neurons in hidden layers | 8, 6, 10 |
| Activation | ReLu |
| Solver | Adam |
| Regularization | 0.01 |
| Max number of iterations | 200 |

3.7 tabula

**SVM (3.8 tabula )**

|  |  |
| --- | --- |
| **Hiperparemetra nosaukums** | **Vērtība** |
| SVM Cost ( C ) | 40 |
| Regression loss epsilon | 0.50 |
| Kernel | RBF |
| Numerical tolerance | 0.010 |
| Iteration limit | 100 |

3.8 tabula

**kNN ( K – Nearest Neighbour ) ( 3.9 tabula )**

|  |  |
| --- | --- |
| **Hiperparematra nosaukums** | **Vertība** |
| Number of neighbors | 30 |
| Metric | Euclidean |
| Weight | Mahalanobis |

3.9 tabula

Graphical user interface, text, application, email

Description automatically generated

* 1. Attēls

Pēc trešā eksperimenta var secināt, ka visi algoritmi sniedz labākus rezultātus nekā pirmajā un otrajā eksperimentā. Visi rezultāti augstāki nekā 0.8 bet vislabākais rezultāts šajā eksperimentā parādījās SVM algoritmam, jo tā precizitāte pēc visiem novērtēšanas metrika ir vienāda apmēram 0.825. Vissliktākais rezultāts ir kNN algoritmam, jo tā precizitāte pēc visiem novērtēšanas metrika ir vienāda 0.80 – 0.802 , bet labāk nekā iepriekšējās reizēs.

Graphical user interface

Description automatically generated

3.10 Attēls

Graphical user interface, application

Description automatically generated

3.11 Attēls

Pēc visus eksperimentus var secināt, ka vislabākie rezultāti paradās trešā eksperimentā. Trešā eksperimenta algoritmu precizitāte ir ļoti laba. Visliktākais eksperiments ir pirmais, jo rezultāti, kuri biji atšķiras katrā algoritmā. SVM un Neural Network otrā un trešā eksperimentā paradīja ļoti tuvi rezultāti. Lai pārbaudīt visus algoritmus izmantojot TestDate, bija izvelēts trešais eksperiments, jo tur bija vislabākie rezultāti. Var secināt, ka izmantojot 3. eksperimenta hiperparamentrus, rezultāti atšķiras. Vislabākais algoritms 3.11 attēlā ir kNN algoritms, jo tur paradījās labākie rezultāti, nekā SVM un Neural Network. Piemēram, kNN pēc visiem novērtēšanas metrika ir vienāda 0.788

**Avotu saraksts**

* [**https://www.muratkoklu.com/datasets/**](https://www.muratkoklu.com/datasets/)
* [**https://www.hindawi.com/journals/mpe/2021/4793293/**](https://www.hindawi.com/journals/mpe/2021/4793293/) **-** Informācija par datu kopu
* [**https://research.ijcaonline.org/volume83/number15/pxc3892915.pdf**](https://research.ijcaonline.org/volume83/number15/pxc3892915.pdf) **-** A. Beyaz, R. Ozturk, and U. Turker, “Assessment of mechanical damage on apples with image analysis,” Journal of Food Agriculture and Environment, vol. 8, no. 3&4, pp. 476–480, 2010.
* [**https://research.ijcaonline.org/volume83/number15/pxc3892915.pdf**](https://research.ijcaonline.org/volume83/number15/pxc3892915.pdf) **-** A. M. Vyas, B. Talati, and S. Naik, “Colour feature extraction techniques of fruits: a survey,” *International Journal of Computer Applications*, vol. 83, no. 15, 2013.
* [**https://ijisae.org/index.php/IJISAE/article/view/1068/599**](https://ijisae.org/index.php/IJISAE/article/view/1068/599) **-** I. Cinar and M. Koklu, “Classification of rice varieties using artificial intelligence methods,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 7, no. 3, pp. 188–194, 2019.
* [**https://agro.icm.edu.pl/agro/element/bwmeta1.element.agro-20f5b343-f1df-4785-93b1-1f28c3d79222/c/Wheat\_class.pdf**](https://agro.icm.edu.pl/agro/element/bwmeta1.element.agro-20f5b343-f1df-4785-93b1-1f28c3d79222/c/Wheat_class.pdf) **-** A. Arefi, A. M. Motlagh, and R. F. Teimourlou, “Wheat class identification using computer vision system and artificial neural networks,” *International Agrophysics*, vol. 25, no. 4, 2011.
* [**http://thejaps.org.pk/docs/v-24-1/47.pdf**](http://thejaps.org.pk/docs/v-24-1/47.pdf) **-** A. Pazoki, F. Farokhi, and Z. Pazoki, “Classification of rice grain varieties using two artificial neural networks (MLP and neuro-fuzzy),” *The Journal of Animal & Plant Sciences*, vol. 24, no. 1, pp. 336–343, 2014.
* [**https://www.sciencedirect.com/science/article/pii/S0889157520314769?via%3Dihub**](https://www.sciencedirect.com/science/article/pii/S0889157520314769?via%3Dihub) **-** M. M. Oliveira, B. V. Cerqueira, S. Barbon, and D. F. Barbin, “Classification of fermented cocoa beans (cut test) using computer vision,” *Journal of Food Composition and Analysis*, vol. 97, Article ID 103771, 2021.
* [**https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm**](https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm) **-** ParSVM ( Support Vector Machine) algoritmu .
* [**https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning**](https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning) **-** Par K-Nearest Neighbor(KNN) algoritmu**.**