

**Rīgas Tehniskā universitāte Datorzinātnes un informācijas tehnoloģijas
fakultāte**

Mākslīga intelekta pamati

Otrā praktiskā darba atskaite

Izpildītājs: Nikita Dolzenkovs
7.grupa 191RBC111

Saite uz projektu: https://github.com/tinuzu/2_pr_dr/upload/main

2023. / 2024. m. g.

SATURS

| | |
|---|----|
| IEVADS | 3 |
| I. DATU PIRMAPSTRĀDE | 4 |
| 1.1. Datu kopas apraksts | 4 |
| 1.2. Datu kopas satura apraksts | 4 |
| 1.3. I daļas statistiskie rādītāji | 6 |
| I daļa (jautājumi un atbildes) | 10 |
| 1.5. Rādītāju analīze | 11 |
| II. NEPĀRRAUDZĪTĀ MAŠĪNMĀCĪŠANĀS | 13 |
| 2.1. K-vidējo algoritms | 13 |
| 2.2. hierarhiskā klasterizācija | 17 |
| 2.3. Nepārraudzītās mašīnmācīšanās secinājumi | 22 |
| III. PĀRRAUDZĪTĀ MAŠĪNMĀCĪŠANĀS | 23 |
| 3.1. kNN algoritms | 23 |
| 3.2. Grādienta palielināšana | 24 |
| 3.3. Apmācība un testēšana | 24 |
| 3.4. Modeļu izveidošana | 26 |
| SECINĀJUMI | 28 |
| IZMANTOTIE AVOTI UN LITERATŪRA | 29 |

IEVADS

Mums (studentiem) priekšmetā “Mākslīga intelekta pamati” bija piedāvāts apskatīt un izpildīt uzdevums ar jēbkur datu kopu pēc mūsu izvēles (no uzdevuma piedāvātajiem resursiem). Tā pat mums ir nepieciešams izmantot nepārraudzītos un pārraudzītos mašīnmācīšanās algoritmus, par kuriem mums bija pastāstīts lekcijas laikā. Darbs jābūt izpildīts izmantojot Orange rīku.

Darbs būs sadalīts uz trīs daļām.

1. Datu kopas analīze, apraksts, apstrādāšanas sagatavošana.
2. Nepārraudzītās mašīnmācīšanās
3. Pārraudzītās mašīnmācīšanās

I. DATU PIRMAPSTRĀDE

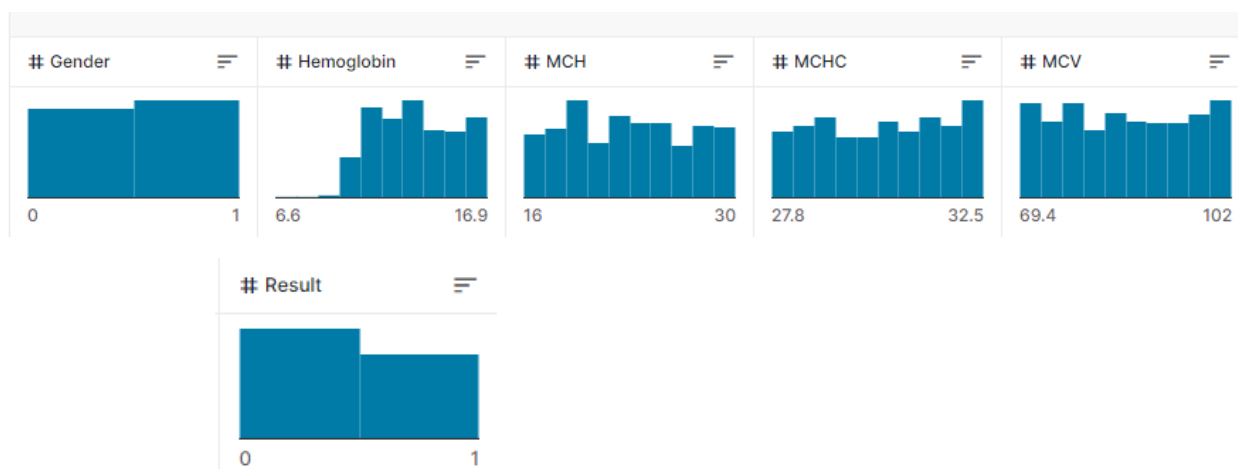
1.1. Datu kopas apraksts

Manim izvēlēta datu kopa ir “ANEMIA DATASET”. Šo datu kopu izveidoja un publicēja Bisva Ranžan Rao (angl. Biswa Ranjan Rao). Ar šo datu kopu, kura satur 1421 objektu, var prognozēt Anemijas diagnozi. Šī datu kopa satur tādus datus, kā (dzimums, hemoglobīns, MCV – vidējais eritrocītu tilpums; MCHC – vidējā hemoglobīna koncentrācija eritrocītā; MCH – vidējais hemoglobīns saturs eritrocītā; rezultāts). (License: Unknown).(1)

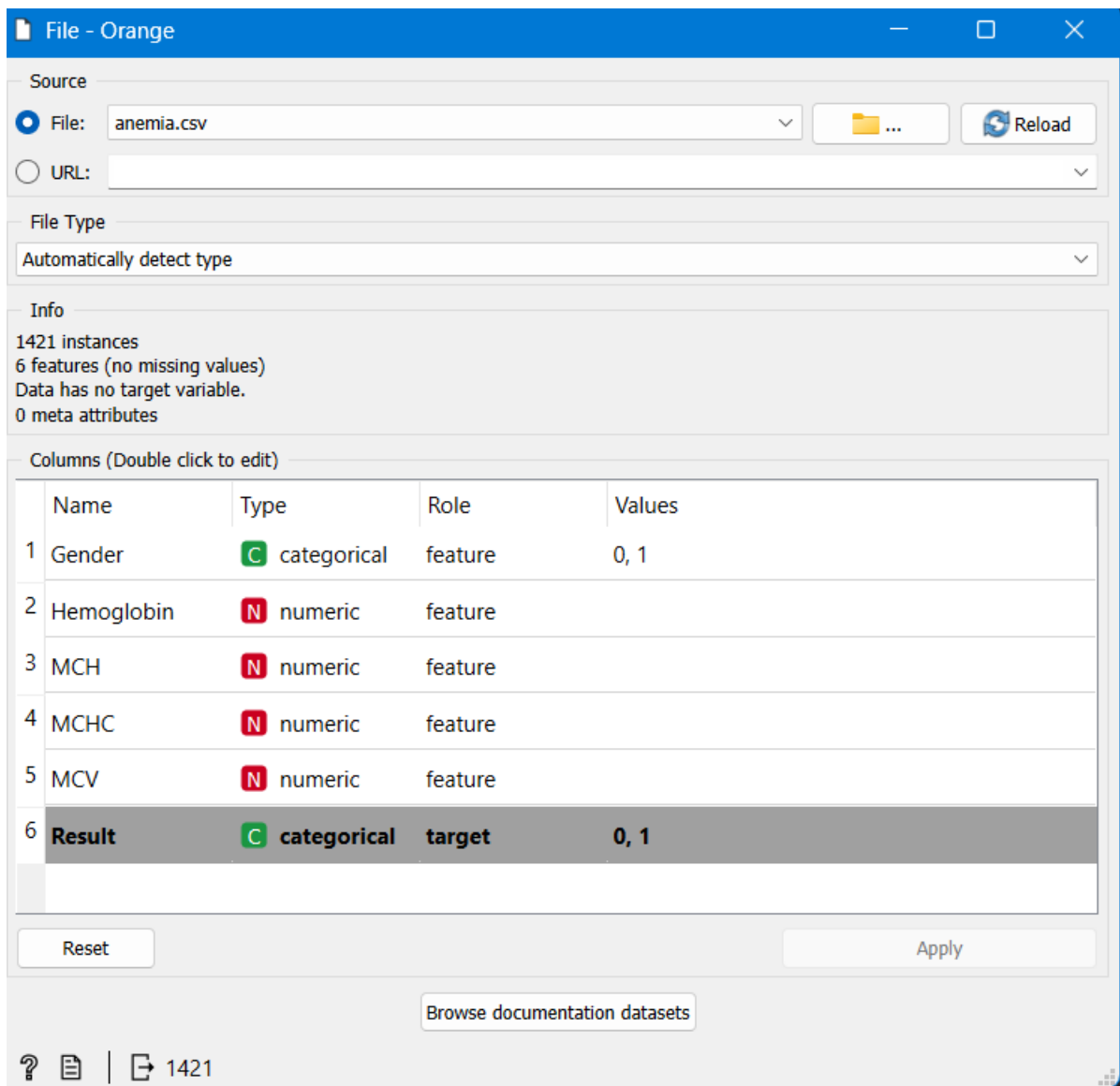
1.2. Datu kopas satura apraksts

| Atribūts | Vērtība | Tips |
|------------------------------|---|-------------|
| Dzimums (Gender) | No 0 līdz 1, kur 0 ir vīrietis un 1 ir sieviete | Kategorisks |
| Hemoglobīns (Hemoglobīns) | No 6,6 līdz 16,9 | Numerisks |
| MCV | No 69,4 līdz 102 | Numerisks |
| MCH | No 16 līdz 30 | Numerisks |
| MCHC | No 27,8 līdz 32,5 | Numerisks |
| Result (Rezultāts) | No 0 līdz 1, kur 0 ir negatīvs un 1 ir pozitīvs | Kategorisks |

1.1.tab. Datu kopas pazīmes, vērtības un tipi



1.1.att. Kaggle datu kopas atribūtu apraksti.



1.2.att. Orange rīkā kopas realizācija.

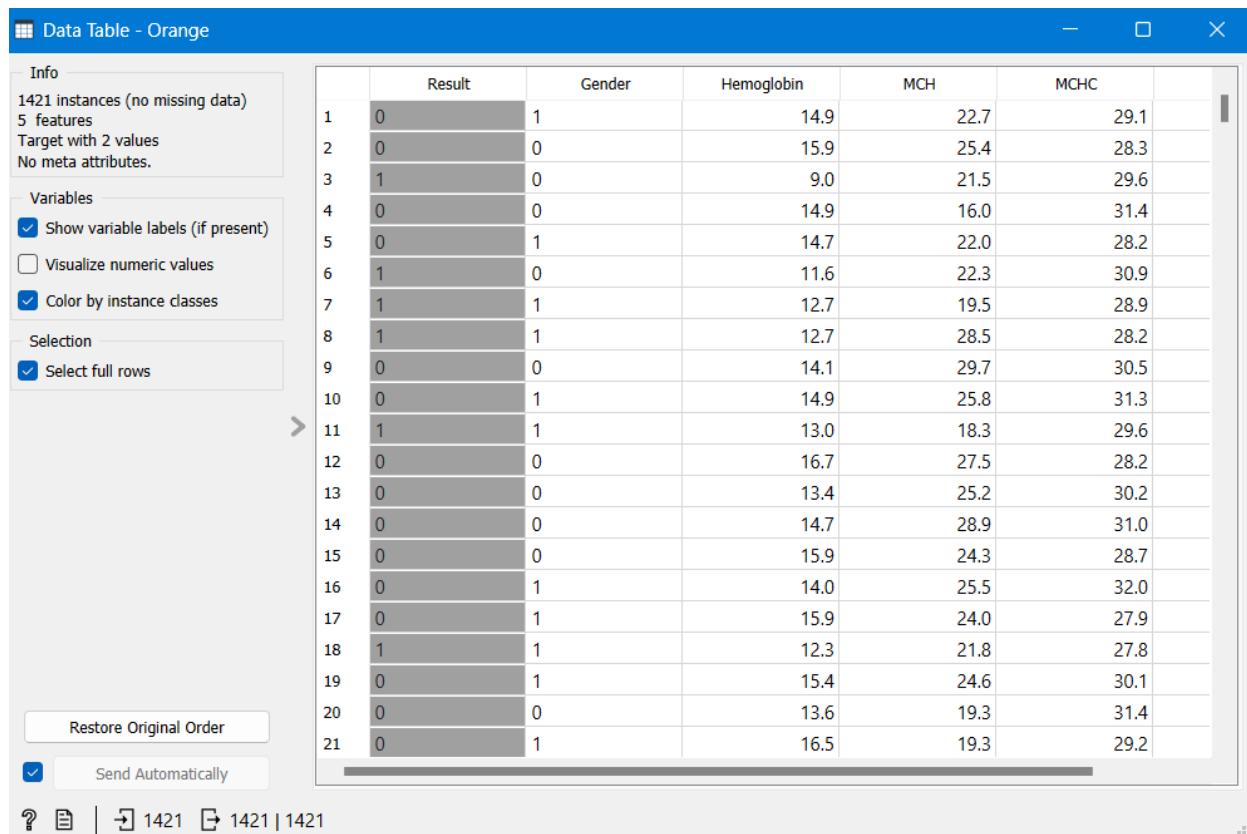
Šī datu kopā ir 2 klases, ar kurām var klasificēt objektus.

| Gendrs | Skaits |
|-----------|--------|
| Vīrieši | 681 |
| Sievietes | 740 |

1.2.tab. “Gender” kategorijas objektu skaits.

| Anemijas rezultāts | Skaitis |
|--------------------|---------|
| Negatīvs | 801 |
| Pozitīvs | 620 |

1.3.tab. "Result" kategorijas objektu skaits.



Data Table - Orange

Info
 1421 instances (no missing data)
 5 features
 Target with 2 values
 No meta attributes.

Variables
☒ Show variable labels (if present)
☐ Visualize numeric values
☒ Color by instance classes

Selection
☒ Select full rows

Restore Original Order

☒ Send Automatically

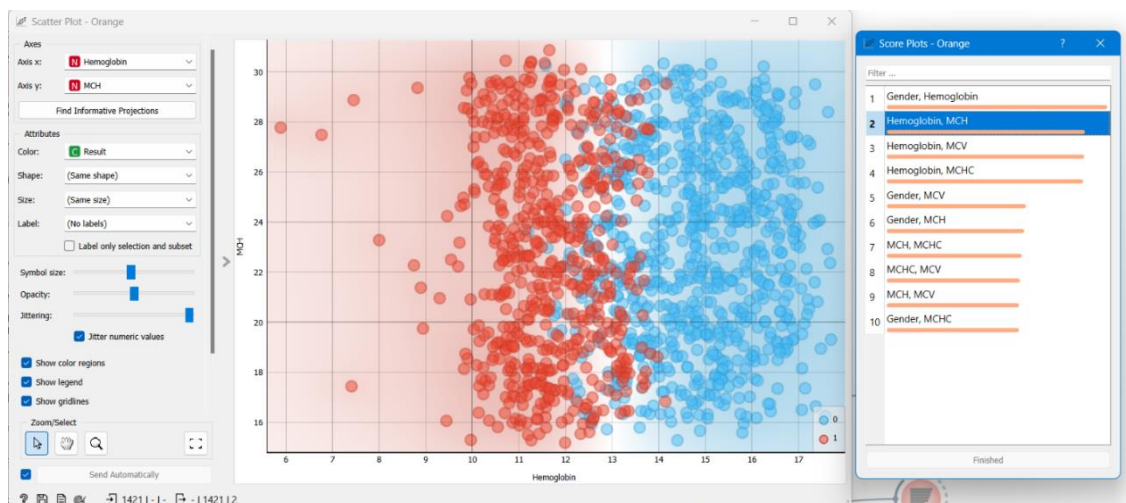
| | Result | Gender | Hemoglobin | MCH | MCHC |
|----|--------|--------|------------|------|------|
| 1 | 0 | 1 | 14.9 | 22.7 | 29.1 |
| 2 | 0 | 0 | 15.9 | 25.4 | 28.3 |
| 3 | 1 | 0 | 9.0 | 21.5 | 29.6 |
| 4 | 0 | 0 | 14.9 | 16.0 | 31.4 |
| 5 | 0 | 1 | 14.7 | 22.0 | 28.2 |
| 6 | 1 | 0 | 11.6 | 22.3 | 30.9 |
| 7 | 1 | 1 | 12.7 | 19.5 | 28.9 |
| 8 | 1 | 1 | 12.7 | 28.5 | 28.2 |
| 9 | 0 | 0 | 14.1 | 29.7 | 30.5 |
| 10 | 0 | 1 | 14.9 | 25.8 | 31.3 |
| 11 | 1 | 1 | 13.0 | 18.3 | 29.6 |
| 12 | 0 | 0 | 16.7 | 27.5 | 28.2 |
| 13 | 0 | 0 | 13.4 | 25.2 | 30.2 |
| 14 | 0 | 0 | 14.7 | 28.9 | 31.0 |
| 15 | 0 | 0 | 15.9 | 24.3 | 28.7 |
| 16 | 0 | 1 | 14.0 | 25.5 | 32.0 |
| 17 | 0 | 1 | 15.9 | 24.0 | 27.9 |
| 18 | 1 | 1 | 12.3 | 21.8 | 27.8 |
| 19 | 0 | 1 | 15.4 | 24.6 | 30.1 |
| 20 | 0 | 0 | 13.6 | 19.3 | 31.4 |
| 21 | 0 | 1 | 16.5 | 19.3 | 29.2 |

1421 | 1421

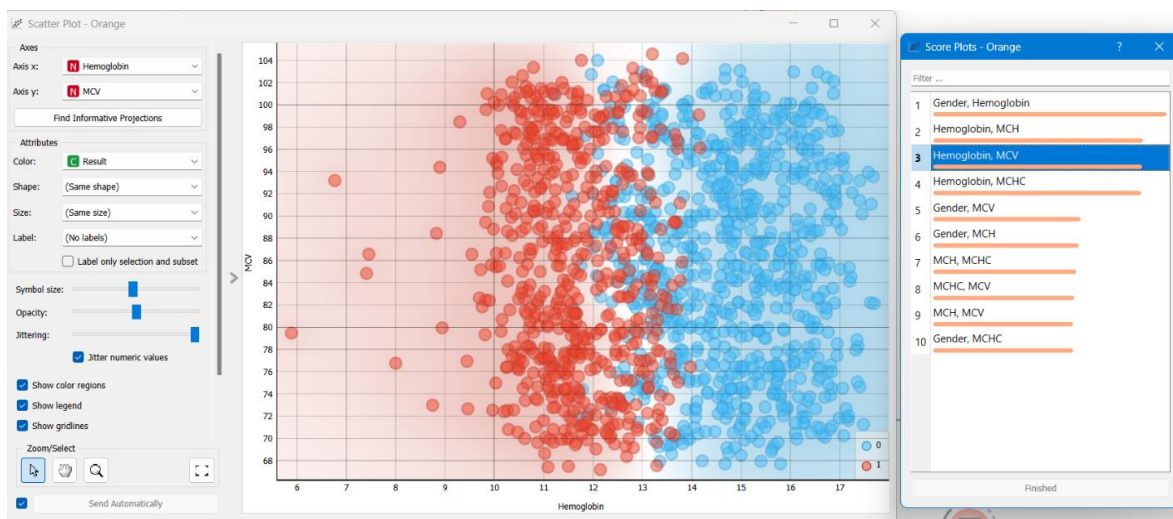
1.3.att. Datu faila struktūras fragments.

1.3. I daļas statistiskie rādītāji.

a) izkliedes diagrammas (scatter plot)

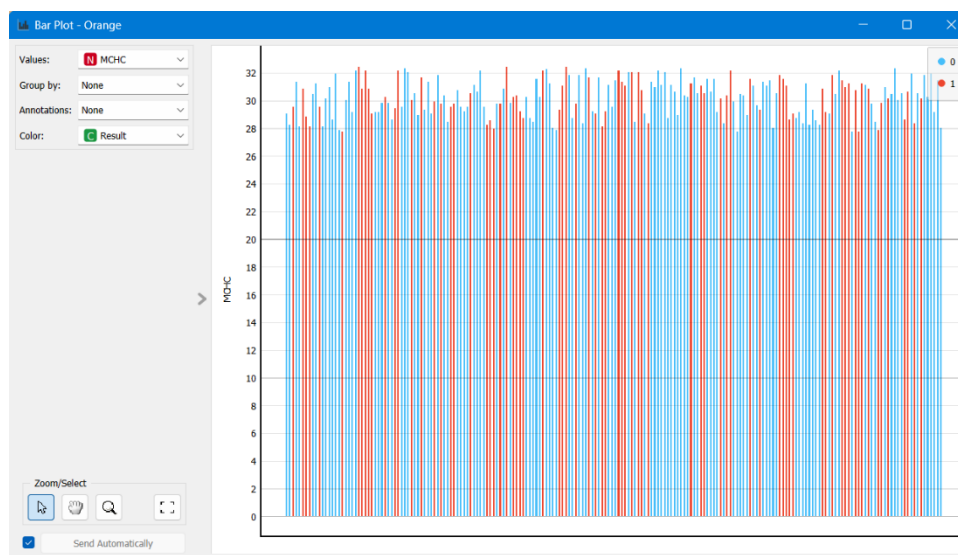


1.4.att. Izklīdes diagramma (“Hemoglobin/MCH”).

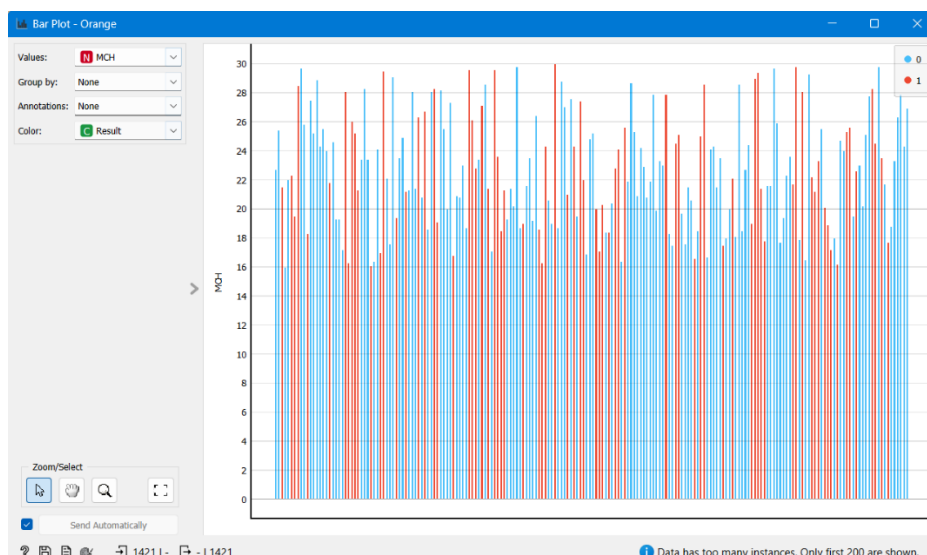


1.5.att. Izklīdes diagramma (“Hemoglobin/MCV”).

b) Klases atdalīšana ar histogrammām

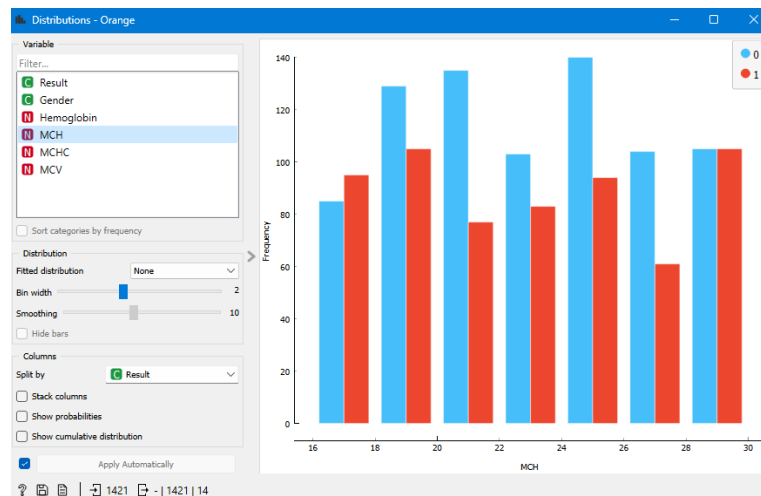


1.6.att. Histogramma, ar “MCHC” vērtību.

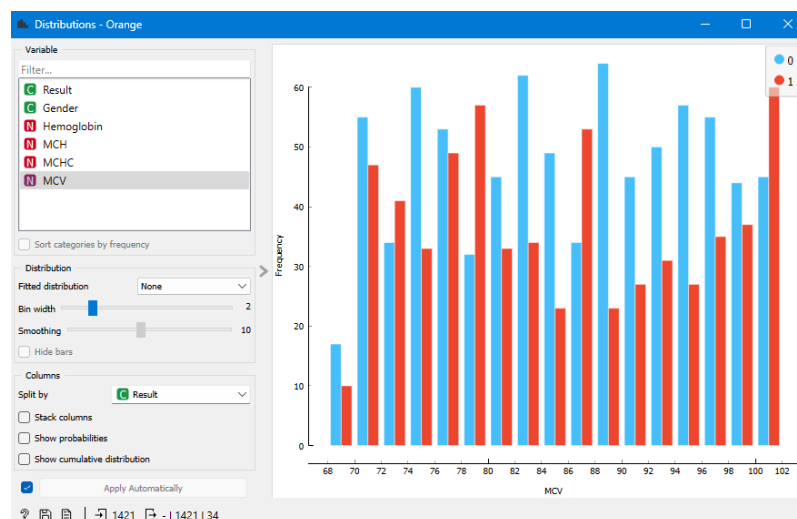


1.7.att. Histogramma, ar “MCP” vērtību.

c) Interesējošo pazīmju (atribūtu) sadalījums



1.8.att. Pazīmes “MCH” sadalījums.



1.9.att. Pazīmes “MCV” sadalījums.

d) Statistiskie rādītāji



1.10.att. Statistiskie rādītāji.

I daļa (jautājumi un atbildes)

1. Vai klases datu kopā ir līdzsvarotas, vai dominē viena (vai vairākas) klases?

Datu bāzē, kuru es izvēlējos, rezultāts ir gandrīz līdzsvarots, bet negatīvs rezultāts nedaudz dominē.

2. Vai datu atspoguļojums ļauj redzēt datu struktūru?

Izmantojot izklaides diagrammu es varu korekti redzēt klases un datus.

Izmantojot histogrammas es varu redzēt kādi dati dominē konkrētā klasē.

3. Cik datu grupējums ir iespējams identificēt, pētot datu vizuālo atspoguļojumu?

Ņemot vērā izklaides diagrammu es varu redzēt datus, un kā tie ir grupējās.

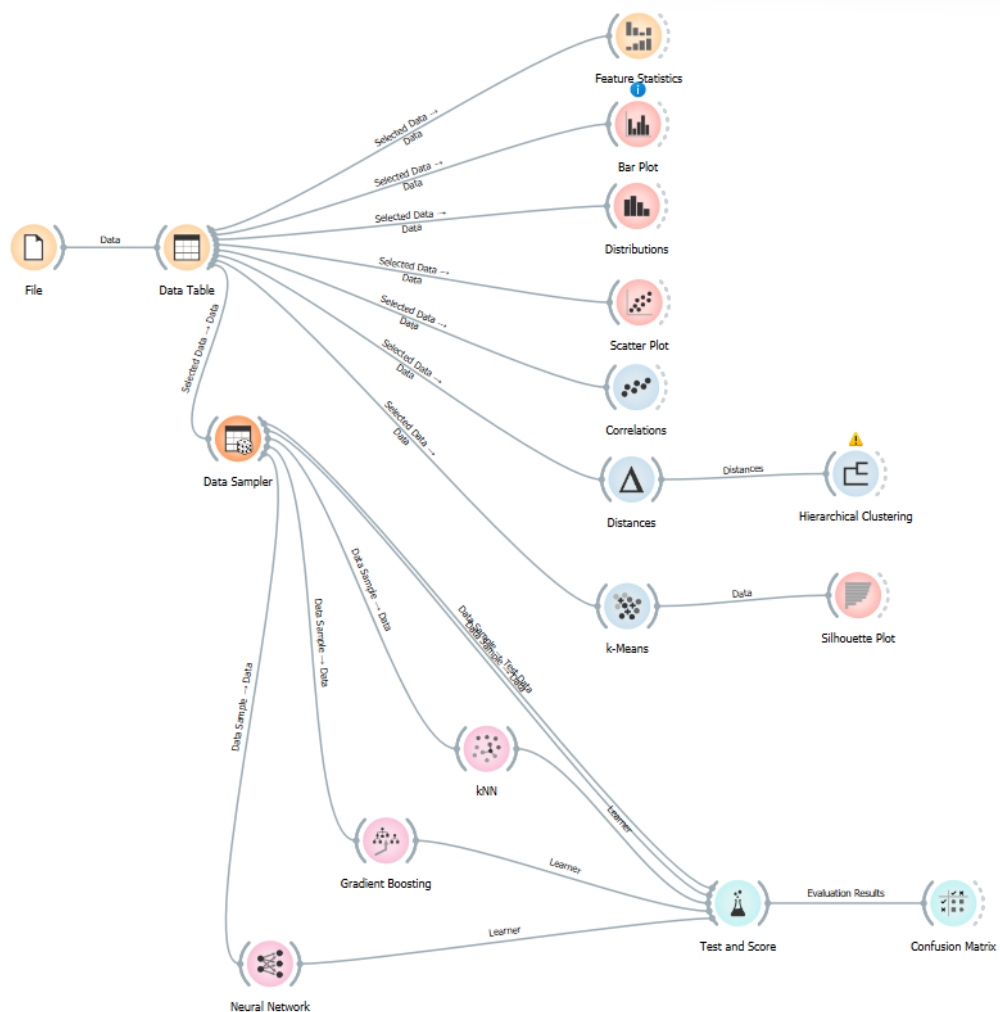
4. Vai identificētie datu grupējumi atrodas tuvu viens otram vai tāli viens no otra?

Jā, puse no datiem atrodas tuvi viens otram, bet ir vietas, kur tie ir tāluma viens no otra.

1.5. Rādītāju analīze

Pamatojoties uz datu kopas statistisko rādītāju, es varu secināt, ka:

- Visiem datiem modas un mediānas dati ir gandrīz vienādi
- (MCV) ir vislielākā vērtība un tā ir 101,6
- (Hemoglobin) ir vismazākā vērtība un tā ir 6,6



1.12.att. Orange darbs

| A1 : Gender,Hemoglobin,MCH,MCHC,MCV,Result | | | | | | | | | | |
|--|---------------------------------------|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | E | F | G | H | I | J |
| 1 | Gender,Hemoglobin,MCH,MCHC,MCV,Result | | | | | | | | | |
| 2 | 1,14.9,22.7,29.1,83.7,0 | | | | | | | | | |
| 3 | 0,15.9,25.4,28.3,72,0 | | | | | | | | | |
| 4 | 0,9,21.5,29.6,71.2,1 | | | | | | | | | |
| 5 | 0,14.9,16,31.4,87.5,0 | | | | | | | | | |
| 6 | 1,14.7,22,28.2,99.5,0 | | | | | | | | | |
| 7 | 0,11.6,22.3,30.9,74.5,1 | | | | | | | | | |
| 8 | 1,12.7,19.5,28.9,82.9,1 | | | | | | | | | |
| 9 | 1,12.7,28.5,28.2,92.3,1 | | | | | | | | | |
| 10 | 0,14.1,29.7,30.5,75.2,0 | | | | | | | | | |
| 11 | 1,14.9,25.8,31.3,82.9,0 | | | | | | | | | |
| 12 | 1,13,18.3,29.6,87.8,1 | | | | | | | | | |
| 13 | 0,16.7,27.5,28.2,93,0 | | | | | | | | | |
| 14 | 0,13.4,25.2,30.2,95.9,0 | | | | | | | | | |
| 15 | 0,14.7,28.9,31,69.8,0 | | | | | | | | | |
| 16 | 0,15.9,24.3,28.7,91.5,0 | | | | | | | | | |
| 17 | 1,14,25.5,32,81.6,0 | | | | | | | | | |
| 18 | 1,15.9,24,27.9,83.7,0 | | | | | | | | | |
| 19 | 1,12.3,21.8,27.8,77.9,1 | | | | | | | | | |
| 20 | 1,15.4,24.6,30.1,94.3,0 | | | | | | | | | |
| 21 | 0,13.6,19.3,31.4,78.9,0 | | | | | | | | | |
| 22 | 1,16.5,19.3,29.2,69.7,0 | | | | | | | | | |
| 23 | 1,16.2,17.2,32.2,78.4,0 | | | | | | | | | |
| 24 | 0,6.9,28.1,32.5,94.6,1 | | | | | | | | | |
| 25 | 1,11.8,16.3,30.9,78.7,1 | | | | | | | | | |
| 26 | 1,11,26,32.2,98.9,1 | | | | | | | | | |
| 27 | 1,11,25.2,30.9,83.2,1 | | | | | | | | | |

1.13.att. excel fails

II. NEPĀRRAUDZĪTĀ MAŠĪNMĀCĪŠANĀS

Tagad man ir nepieciešams izmantot neparraudzīto mašīnmācīšanu un es varu izmantot divus klasterizācijas algoritmus:

- 1) k-vidējo algoritms
- 2) hierarhiskā klasterizācija.

2.1. K-vidējo algoritms

Orange rīkā es izmantoju (K-Means) algoritmu kur ir tādi parametri kā:

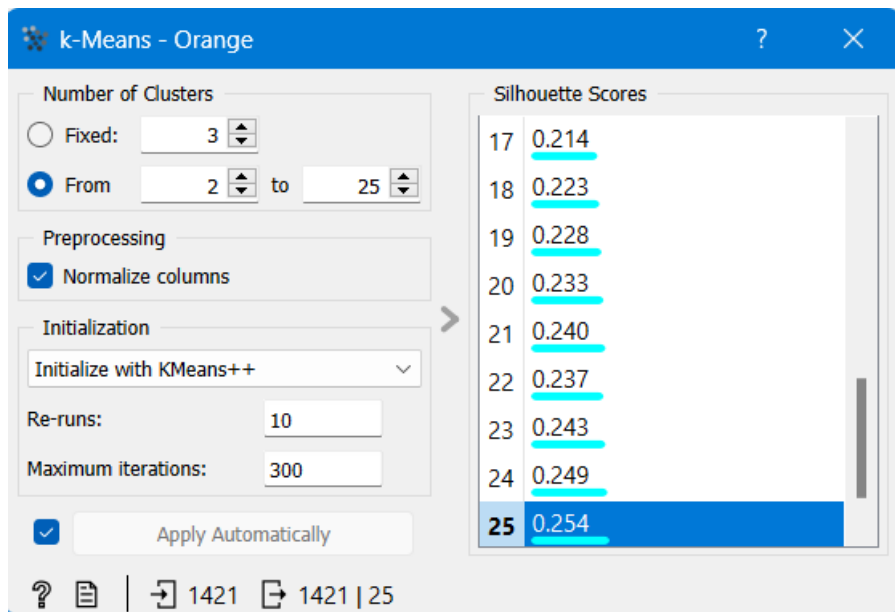
Number of Clusters:

- Fixed (vienīgs (fiksēts) klasteru skaits);
- From A to B (tas ir diapazons no un līdz, kur es pats varu izvēlēties klasteru skaitu)

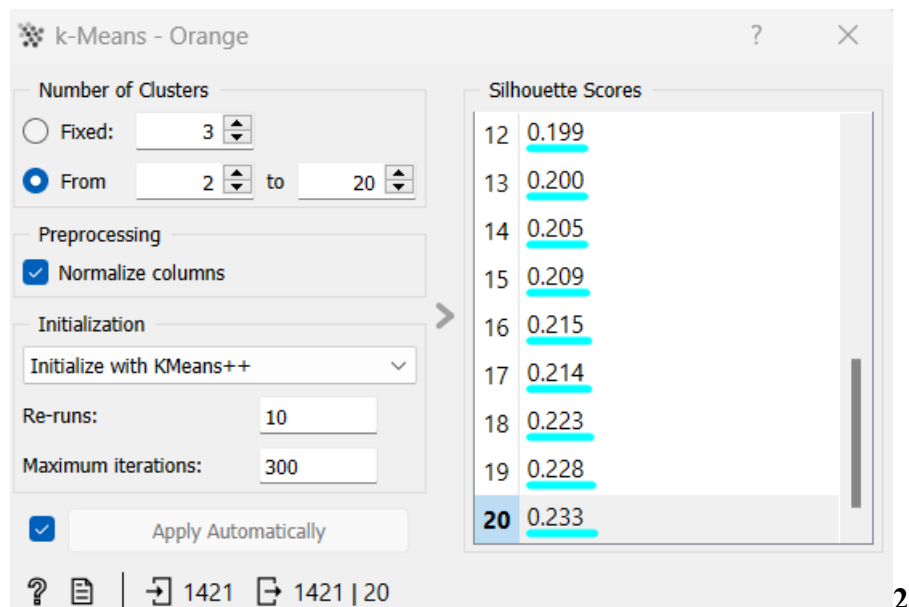
Preprocessing: ((normalizē columns) normalizē kolonnas vērtību vienam diapozonam)

Initialization:

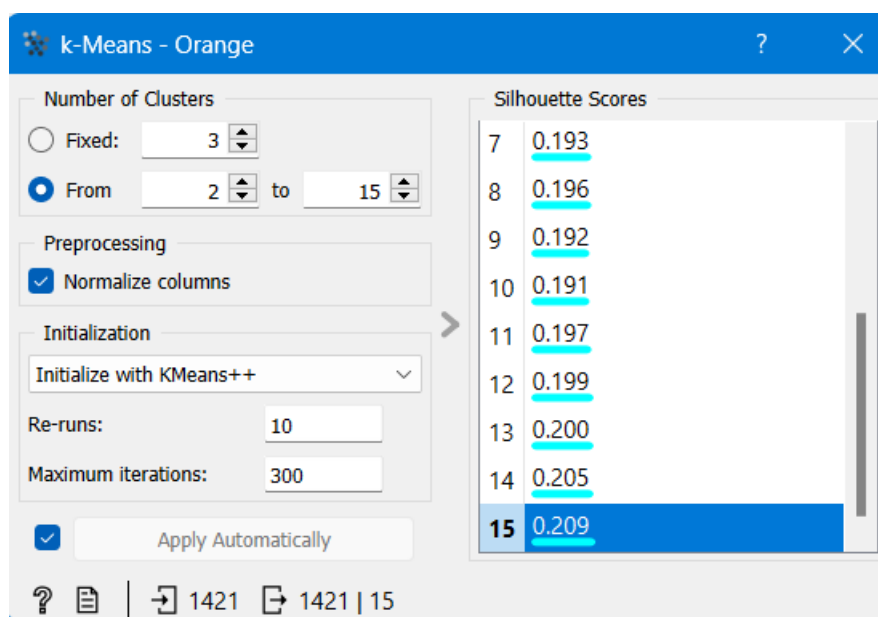
- Re-runs (atkarto procesu tik reizes, cik ir norādīts)
- Maximum iterations (iterāciju maksimālais skaits)



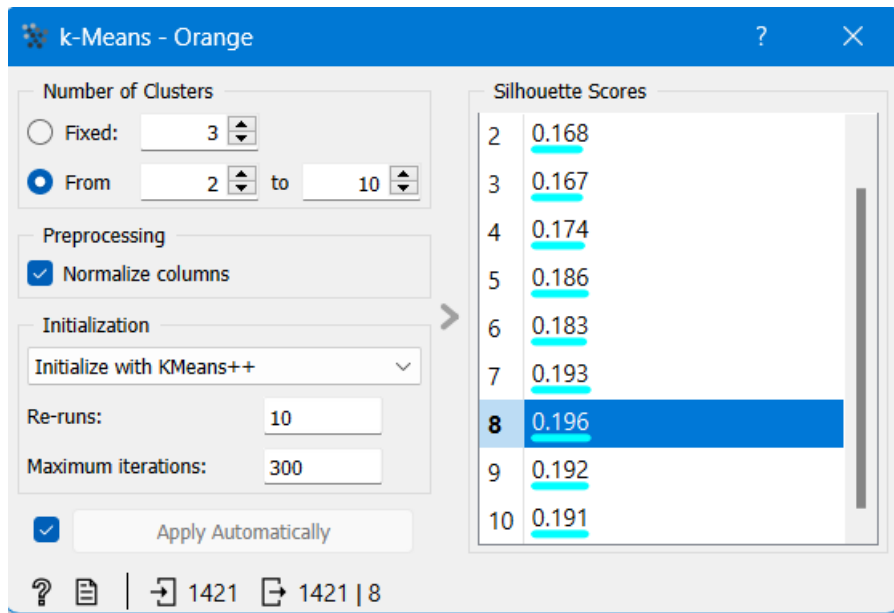
2.1.att. k-vidējo algoritma vislabākais klasteru skaits no 2 līdz 25.



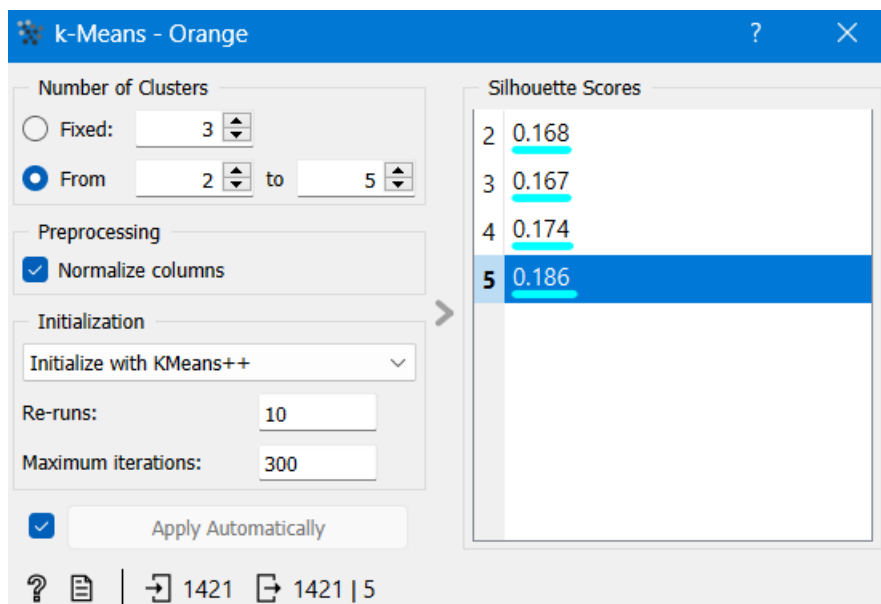
2.2.att. k-vidējo algoritma vislabākais klasteru skaits no 2 līdz 20.



2.3.att. k-vidējo algoritma vislabākais klasteru skaits no 2 līdz 15.



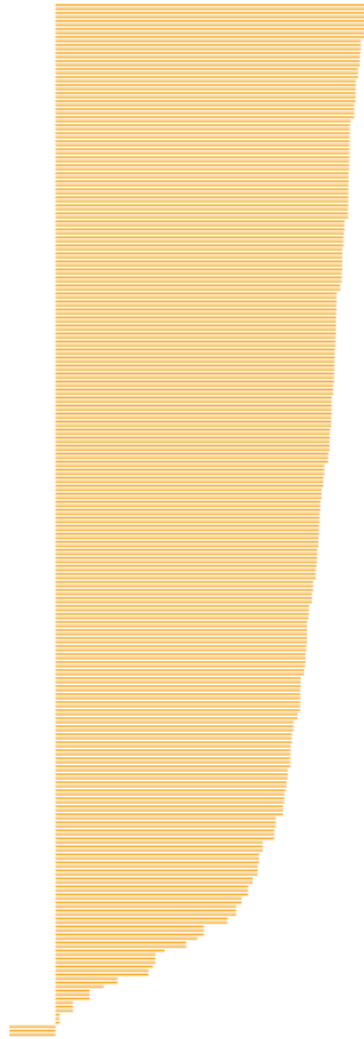
2.4.att. k-vidējo algoritma vislabākais klasteru skaits no 2 līdz 10.

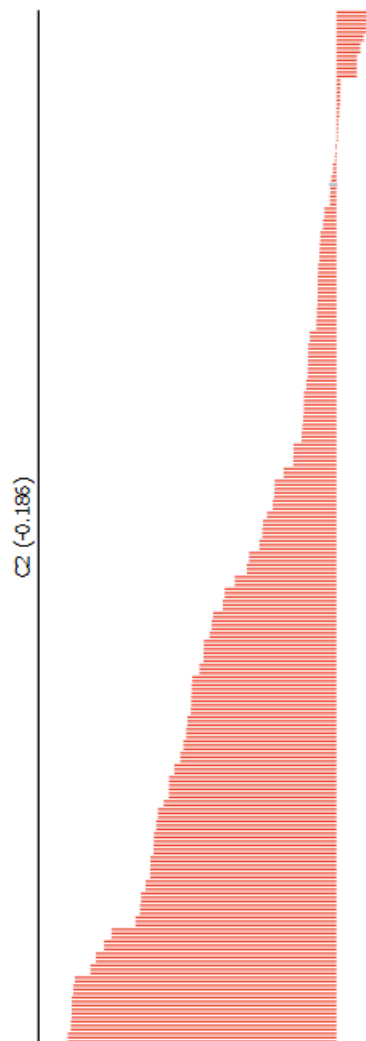


2.5.att. k-vidējo algoritma vislabākais klasteru skaits no 2 līdz 5.

Jā mēs apskatīsim algoritmu ar klasteru skaitu no 2 līdz 5, Orange rīks attēlos nākamo(diemžēl nevaru ielikt visu attēlu kpā, jo tas ir pārāk liels):

C4 (0.480)





Puse no klasteriem ir tāda, kā att. un otrā puse, kā attēla att.. Atkarība no šiem datiem, es varu secināt, ka klases nav labi atdalītas ar šo (k-vidējo) algoritmu.

2.2. hierarhiskā klasterizācija.

Es arī varu izmantot otro no piedāvātajiem algoritmiem kurš saucās par heirarhisko klasterizāciju.

Vņam ir tādi parametri, kā:

Linkage (saiste starp klasteriem):

- Single (aprēķina attālumu starp diviem elementiem no diviem klasteriem);

- Average (aprēķina vidējo attālumu starp diviem elementiem no diviem klasteriem);
- Weighted (izmanto metodi WPGMA);
- Complete (aprēķina attālumu starp diviem vistālākajiem elementiem no diviem klasteriem);
- Ward (aprēķina summas kļūdas pieaugumu, minimizē kopējo iekšējo variāciju).

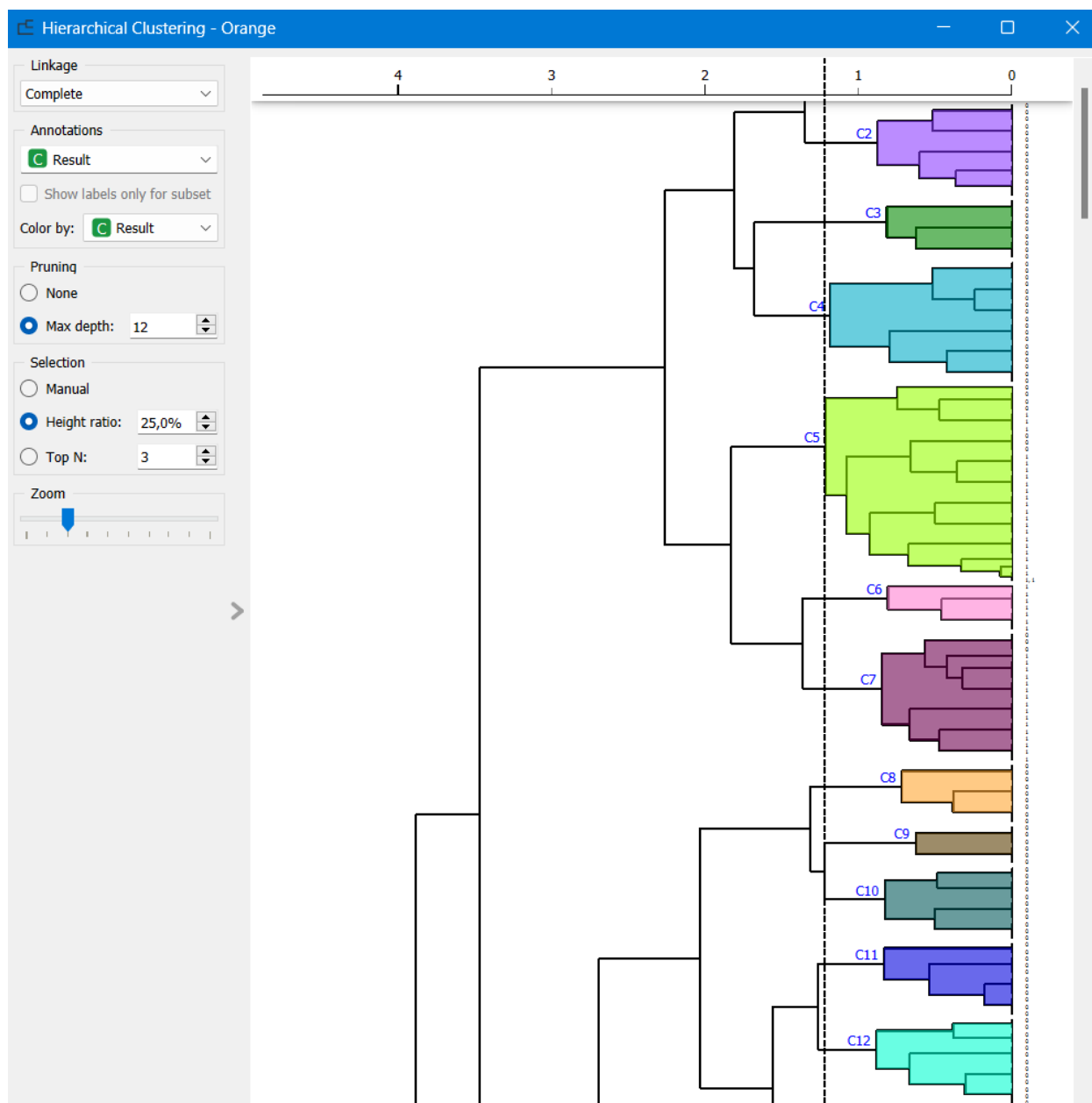
Annotations: (anotē datu objektus)

Pruning: (dendogrammas dziļums)

Selection: (izvēle)

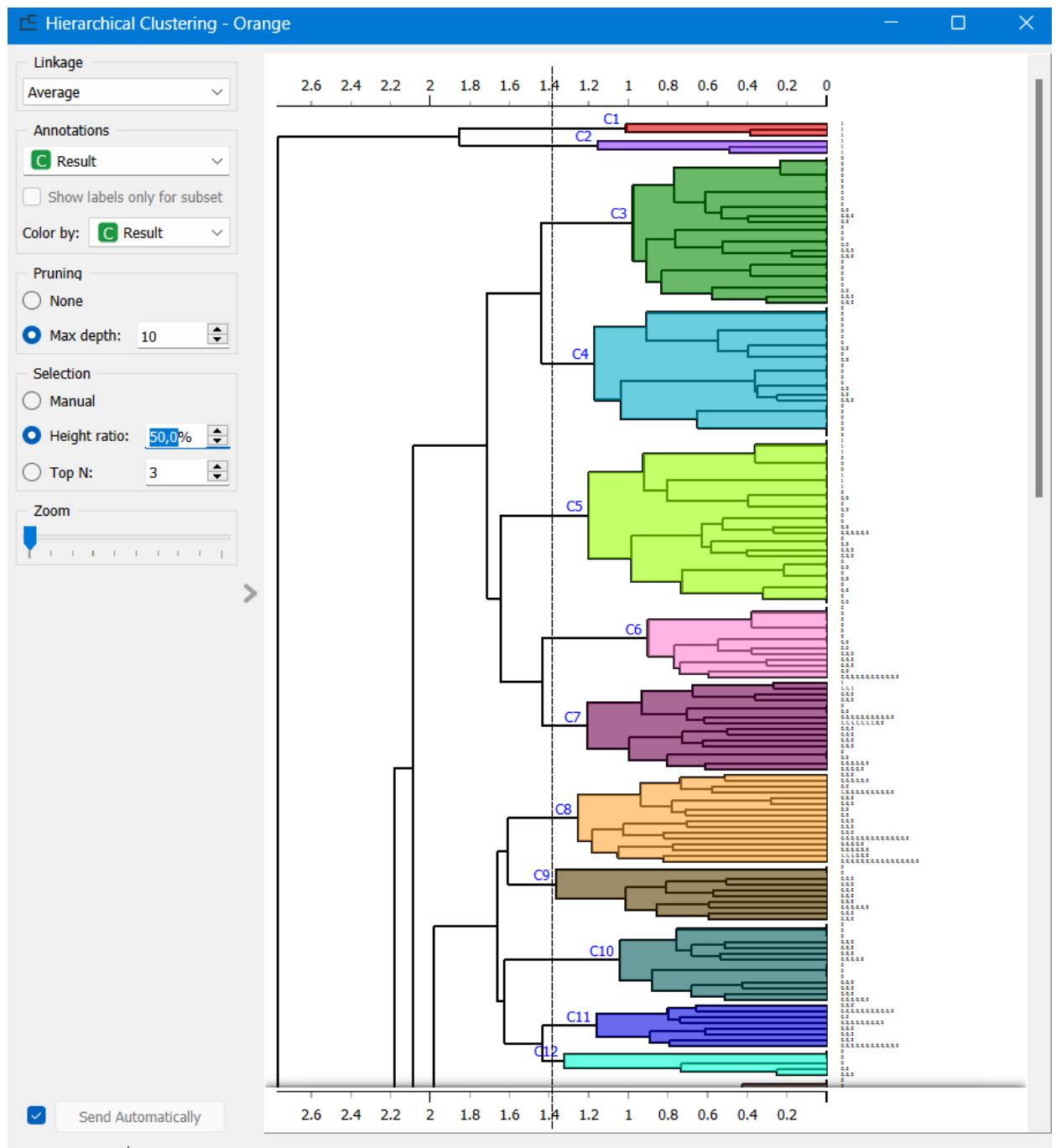
- Manual (ir iespēja pašam izvēlēties klasterus)
- Height ratio (atdalīšanas līnija)
- Top N (lielākais klasteru skaits)(3)

Pēc uzdevuma man ir nepieciešams izpildīt trīs eksperimenti ar Height ratio (atdalīšanas līnija)



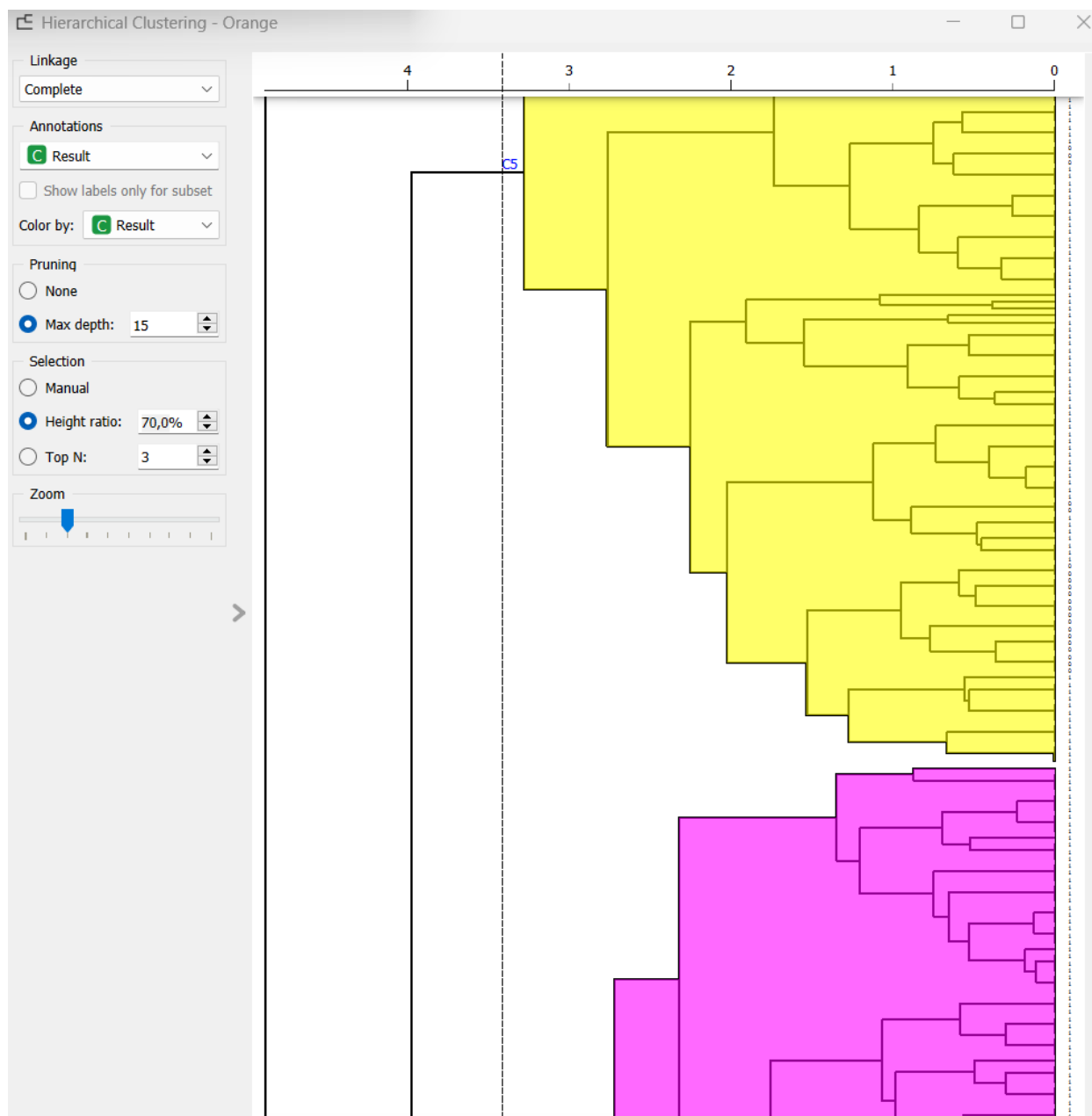
2.6.att. Hierarhiskā klasterizācija (Complete = saite starp klasteriem; 12 = maksimālais dziļums; 25% = atdalīšanas līnija).

Izmantojot parametrus (Complete = saite starp klasteriem; 12 = maksimālais dziļums; 25% = atdalīšanas līnija) mēs varam ieraudzīt, ka daudzi dati ir sadalīti atsevišķās daļās bet ļoti korekti



2.7.att. Hierarhiskā klasterizācija (Average = saite starp klasteriem; 10 = maksimālais dziļums; 50% = atdalīšanas līnija).

Daudzi dati (un negatīvi, un pozitīvi) ir sadalīti vienās klasterī, bet ir arī tādi klasteri, kur dati ir sadalīti korekti.



2.8.att Hierarhiskā klasterizācija (Compete = saite starp klasteriem; 15 = maksimālais dziļums; 70% = atdalīšanas līnija).

Šajā attēlā es varu secināt, ka klasteros ir daudz dažādās klases. Pateicoties šiem datiem es varu secināt, ka šī klasterizācijā nav laba, un es nevaru korekti atdalīt klases.

2.3. Nepārraudzītās mašīnmācīšanās secinājumi

Darboties ar nepārraudzītu mašīnmācīšanu es varu secināt, ka klases datu kopās gandrīz visur nav labi sadalīta. Bet gribu ieteikt, ka heirarhiskā klasterizācija ar šo uzdevumu nodarbojās labāk, nekā k-videjais algoritms. Protams tam arī ir savi trūkumi.

Man liekās, ka abi no šiem algoritmiem nav ideāli lai klasificēt šo datu kopu, bet par šām metodēm nevajag aizmirst, jo tās var piederēt kādās citās situācijās.

III. PĀRRAUDZĪTĀ MAŠĪNMĀCĪŠANĀS

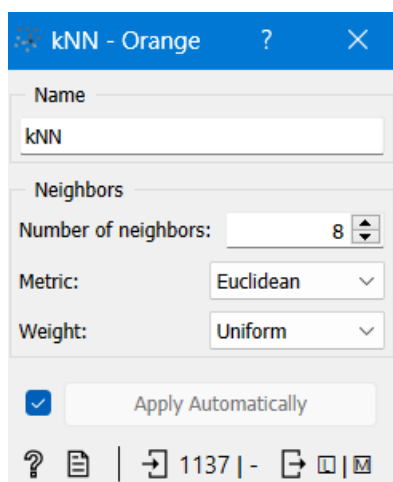
Šajā darba daļā man ir nepieciešams apstrādāt datu kopu izmantojot pārraudzīto mašīnmācīšanu izmantojot tādus algoritmus, kā

- 1) kNN algoritms
- 2) Neironu tīkls
- 3) Gradienta palielināšana

Darbojoties ar Orange funkciju (data sampler), es sadalīju datu kopu uz divām kopām, kur pirmā kopa (apmācīšanas) sastāv no 70 % un otrā (testēšanas) kopa satov no 30%

3.1. kNN algoritms

kNN algoritms – pārraudzītās mašīnmācīšanās algoritms, kuru var izmantot, lai izpildīt kā klasifikāciju, tā arī pret regresijas uzdevumiem. Šis algoritms bija izvēlēts tā, kā viņš ir ļoti viegls izmantošanai un to mēs apskatījam lekciju laikā. Viņš strādā tā: jā ir nepieciešams atrast kādus jaunus datus, algoritms meklē tuvāku datu kopu un dod jauniem datiem to kategoriju vai klasi(kas ir nepieciešams). (4)



3.1.att. kNN logs.

Algoritma hiperparametri:

Neighbors:

- Number of neighbors (raksta cik daudz kaimiņi būs paņemti);
- Metric (attālumu iegūšana un aprēķināšana)
 - Euclidean (attālums starp diviem punktiem)
 - Manhattan (atribūtu atšķirību summa)
 - Chebyshev (maksimālā atšķirību summa)

Mahalanobis (attālums starp punktu un sadalījumu)

- Weight (svars)
 - Uniform (vienāds svars visiem blakus punktiem)
 - Attālums (vistuvākie punkti ietekmē jaunu datu objektu vairāk). (4)

3.2. Gradienta palielināšana

Gradienta palielināšana – pārraudzītās mašīnmācīšanās algoritms, ko izmanto gan klasifikācijas, gan regresijas uzdevumiem. Šis algoritms tika izvēlēts, jo šis algoritms nodarbosies ar kokiem, kuri apvienos lēmumu kokus vienā lielā modelī, lai palielinātu tā precizitāti. (5)

Algoritma hiperparametri:

Metode:

Gradient Boosting (scikit-learn)
Extreme Gradient Boosting (xgboost)
Extreme Gradient Boosting Random Forest (xgboost)
Gradient Boosting (catboost)

Basic Properties:

Number of trees (koku daudzums, lai izveidotu modeli)
Learning rate (algoritma apmācīšanas ātrums)
Replicable training (vai apmācīšana būs atkartota)

Growth Control:

Limit depth of individual trees (koku dziļums)
Do not split subsets smaller than (apakškopu lielums) (5)

3.3. Apmācība un testēšana

1. eksperiments:

- kNN algoritms (8 kaimiņi, metrika: Euclidian, svars: Uniform)
- Neironu tīkls (neironi paslēptos slāņos: 15, 30; aktivācija: Logistic; risina: Adam; mācīšanas ātrums: 0.05; maksimālu iterāciju skaits: 800)
- Gradients (Koku daudzums: 50; Algoritma apmācīšanas ātrums: 0, 100; koku dziļums: 3).

| Model | AUC | CA | F1 | Precision | Recall |
|-------------------|-------|-------|-------|-----------|--------|
| kNN | 0.949 | 0.874 | 0.874 | 0.875 | 0.874 |
| Neural Network | 1.000 | 0.994 | 0.994 | 0.994 | 0.994 |
| Gradient Boosting | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

2.eksperiments:

- kNN algoritms (5 kaimiņi; metrika: Chebyshevs; svars: Distance)
- Neironu tīkls (neironi paslēptos slāņos: 40, 80; aktivācija: ReLu; risina: L-BFGS-B; mācīšanas ātrums: 0.5; maksimālu iterāciju skaits: 1000)
- Gradients (Koku daudzums: 100; Algoritma apmācīšanas ātrums: 0, 200; koku dziūms: 7).

| Evaluation results for target (None, show average over classes) ▼ | | | | | |
|---|-------|-------|-------|-----------|--------|
| Model | AUC | CA | F1 | Precision | Recall |
| kNN | 0.970 | 0.931 | 0.931 | 0.931 | 0.931 |
| Neural Network | 1.000 | 0.993 | 0.993 | 0.993 | 0.993 |
| Gradient Boosting | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

3. eksperiments:

- kNN algoritms (3 kaimiņi; metrika: Mahalanobis; svars: Uniform)
- Neironu tīkls (neironi paslēptos slāņos: 50; aktivācija: tanh; risina: SGD; mācīšanas ātrums: 1; maksimālu iterāciju skaits: 1200)
- Gradients (Koku daudzums:200; Algoritma apmācīšanas ātrums: 0, 400; koku dziūms: 10).

| Evaluation results for target (None, show average over classes) ▼ | | | | | |
|---|-------|-------|-------|-----------|--------|
| Model | AUC | CA | F1 | Precision | Recall |
| kNN | 0.466 | 0.502 | 0.479 | 0.480 | 0.502 |
| Neural Network | 1.000 | 0.984 | 0.984 | 0.984 | 0.984 |
| Gradient Boosting | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

3.4. Modeļu izveidošana

Otrajā eksperimentā medele dod vislabāko precizitāti.

kNN: 5 kaimiņi ir labākais, un tas dod vislabāko precizitāti. Izmantojot metriku: Chebyshevs un savru: Distance mēs dabūjam vislielāko precizitāti.

Neironu tīkli: labākais paslēptais slāņu skaits ir 40,80. Svarīgs bija arī tas, ka es izmantoju aktivācija: ReLu un šo risināja: L-BFGS-B.

Gradianta palielināšana: manuprāt, skatoties uz rezultātu, man liekas, ka es izpildīju kaut ko nepareizi, ja tas nekā nemainās atkarībā no eksperimenta.

| | | Predicted | | Σ |
|----------|---|-----------|-----|----------|
| | | 0 | 1 | |
| Actual | 0 | 383 | 171 | 554 |
| | 1 | 325 | 116 | 441 |
| Σ | | 708 | 287 | 995 |

3.2.att. kNN algoritms

| | | Predicted | | Σ |
|----------|---|-----------|-----|----------|
| | | 0 | 1 | |
| Actual | 0 | 538 | 16 | 554 |
| | 1 | 0 | 441 | 441 |
| Σ | | 538 | 457 | 995 |

3.3.att. Neironu tīkli

| | | Predicted | | Σ |
|--------|----------|-----------|-----|----------|
| | | 0 | 1 | |
| Actual | 0 | 554 | 0 | 554 |
| | 1 | 0 | 441 | 441 |
| | Σ | 554 | 441 | 995 |

3.4.att. Gradianta palielināšana

SECINĀJUMI

Darba gaitā es pirmo reizi strādāju ar Orange rīku. Tas bija vienlaicīgi ļoti interesanti un nepateiktu ka tas bija ļoti viegls. Izamntojot Orange rīku es uzzināju vairāk par to, ka apstrādāt un analizēt datu kopu. Es uzzināju vairāk par algoritmiem, kurus es izmantoju.

Bija ļoti daudz kritēriju un darbs sanāca liels, bet es nekādā gadījumā nevaru teikt, ka esmu izšķērdējis savu laiku. Vel es daudz strādāju ar daudziem algoritmiem un pateicoties Orange rīkam es detalizētāk uzzināju par katru no algoritmiem.

Es gribu pateikt paldies par videomateriāliem, ja nezinot neko par Orange rīku, bez šiem video, pats es patērētu daudzāk laika, tikai lai saprast, ka ar šo dārboties.

Summējot visu, darbs bija interesants, un ja būs, kādas kļūdas es gribētu uzzināt, kur tās ir.

IZMANTOTIE AVOTI UN LITERATŪRA

1. <https://www.kaggle.com/datasets/biswaranjanrao/anemia-dataset>
2. <https://orangedatamining.com/download/#windows>
3. <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/unsupervised/hierarchicalclustering.html>
4. <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/knn.html>
5. <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/gradientboosting.html>
6. <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/neuralnetwork.html>
7. <https://www.youtube.com/watch?v=bmwH3EcTBEM>
8. <https://www.youtube.com/watch?v=ojxvlQSYLr0>
9. <https://www.youtube.com/watch?v=UiGH4v3VKPc>