Rīgas Tehniskā universitāte

Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte

Atskaite par otro praktisko darbu

Studiju kurss “Mākslīgā intelekta pamati”

Komandas numurs: 25

Darba izpildītāji:

Aleksandrs Vaiculevičs (221RDB189)

Artūrs Melmanis (221RDB183)

Aleksandrs Politika (221RDB141)

Violeta Krajeva (211DDB028)

Dmitrijs Firsovs (211RDB310)

Katerina Siņicka (221RDB179)

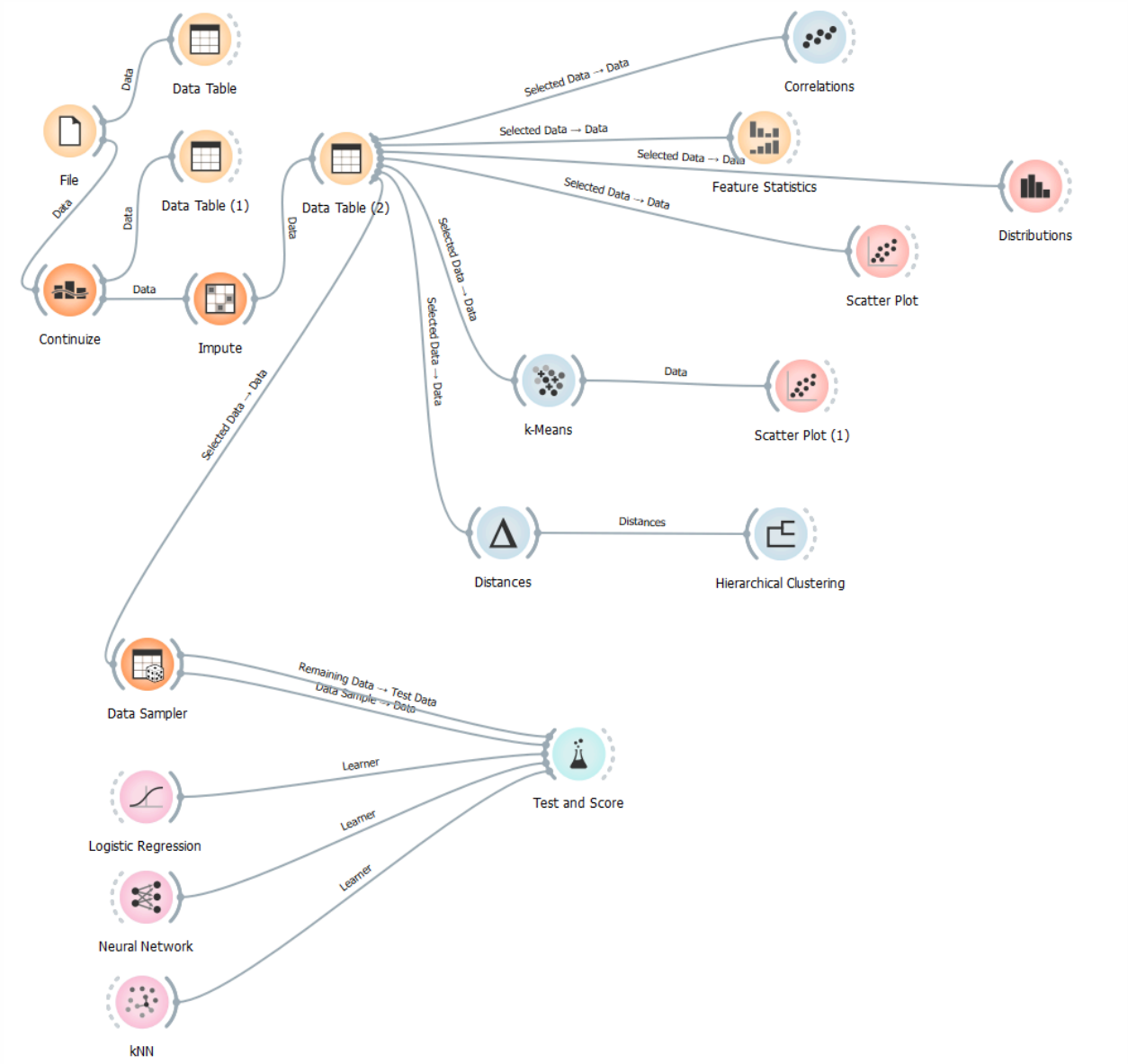
Mācībspēks: Alla Anohina-Naumeca

Saite uz projektu: https://github.com/ArtursMelmanis/MI_25grupa

Saite uz datu kopu: https://www.kaggle.com/datasets/rabieelkharoua/predict-survival-of-patients-with-heart-failure

2023./2024.studiju gads

Orange rīka darbplūsma



I daļa

Datu kopas apraksts

<u>Datu kopas nosaukums:</u>

- *"Predict survival of patients with heart failure"*

<u>Datu kopas avots:</u>

- *Kaggle*

Datu kopas izveidotājs un/vai īpašnieks:

- *Davide Chicco (Krembil Research Institute, Canada)*
- *Giuseppe Jurman (Fondazione Bruno Kessler, Italy)*

Datu kopas problēmsfēras apraksts:

- *Šajā datu kopā savākti 299 pacientu medicīniskie ieraksti ar sirds mazspēju, kas novēroti konkrētā laikā periodā, kur katram pacientam ir 13 klīniskās pazīmes.*

Datu kopas licencēšanas nosacījumi:

- *https://creativecommons.org/licenses/by/4.0/*
- *Tas nozīmē ka mēs varam brīvi koplietot un pielāgot datus.*

Informācija par datu kopas savākšanas veidu vai procedūru:

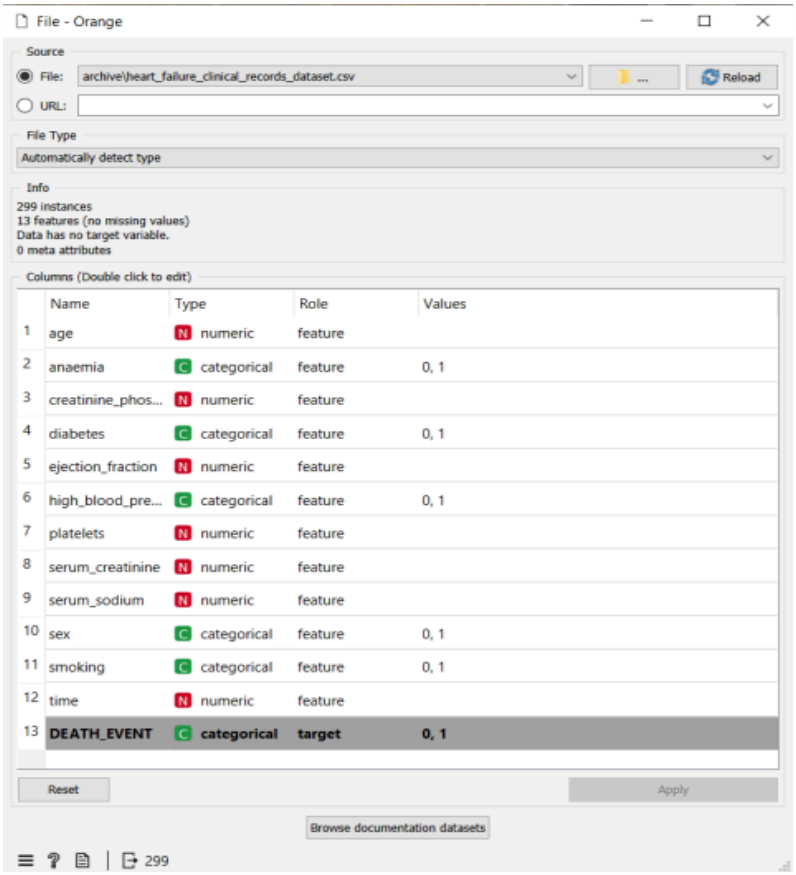
- *Bija savākti medicīniskie ieraksti par 299 sirds mazspējas pacientiem, Faisalabadas Kardioloģijas institūtā un Sabiedroto slimnīcā Faisalabadā, Pendžabā, Pakistānā, laikā no 2015. gada aprīļa līdz decembrim.*

Datu kopas satura apraksts

Datu objektu skaits datu kopā:

- *299 pacientu medicīniskie ieraksti.*

Datu kopas pazīmju (atribūtu) atspoguļojums kopā ar to lomām Orange rīkā:



Klašu skaits datu kopā: 1

Klašu apraksts:

Mūsu gadījumā tabulā ir viena mērķa kolonna, mēs to apskatīsīm, un tieši to var uzskatīt par klasi. Šajā klasē ir divu veidu objekti - 1 un 0, tas nozīmē, vai pacients pētījuma laikā izdzīvoja vai nē. 1 nozīmē, ka viņš neizdzīvoja, 0 nozīmē, ka viņš izdzīvoja.

Datu objektu skaits, kas pieder katrai klasei: 299

Klases iezīme	Datu objektu skaits
1	96
0	203

Pazīmju apraksts: (pēc sākumā tabulas)

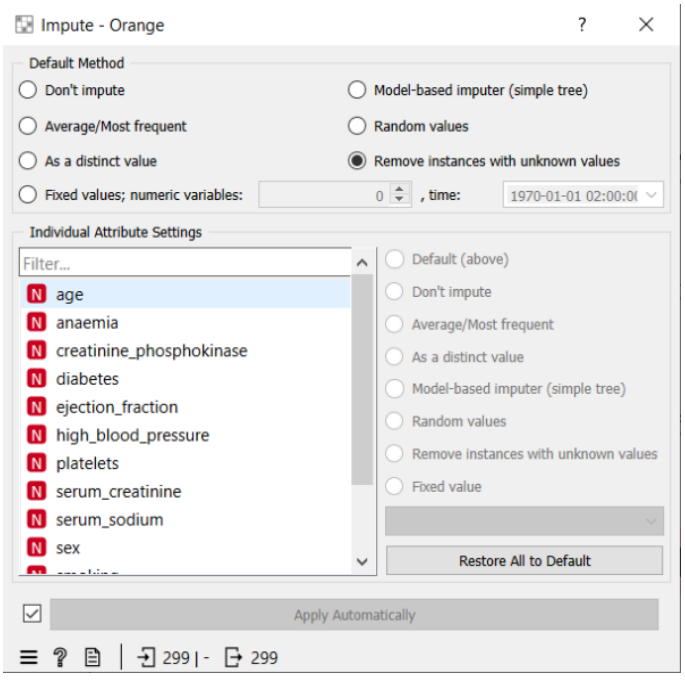
Pazīmes apzīmējums/nosaukums	Pazīmes skaidrojums	Vērtību tips	Vērtību diapazons
age	Pacienta vecums.	numeric	40-95
anaemia	Sarkano asins šūnu vai hemoglobīna līmeņa pazemināšanās.	categorical	0-1
creatinine_phosphokinase	CPK enzīma līmenis asinīs.	numeric	23-7861
diabetes	Ja pacientam ir cukura diabēts.	categorical	0-1
ejection_fraction	Asins procentuālā daļa, kas atstāj sirdi katrā kontrakcijā.	numeric	14-80
high_blood_pressure	Ja pacientam ir hipertensija.	categorical	0-1
platelets	Trombocīti asinīs.	numeric	25100-850000
serum_creatinine	Seruma kreatinīna līmenis asinīs.	numeric	0.50-9.40
serum_sodium	Nātrija līmenis serumā asinīs.	numeric	113-148

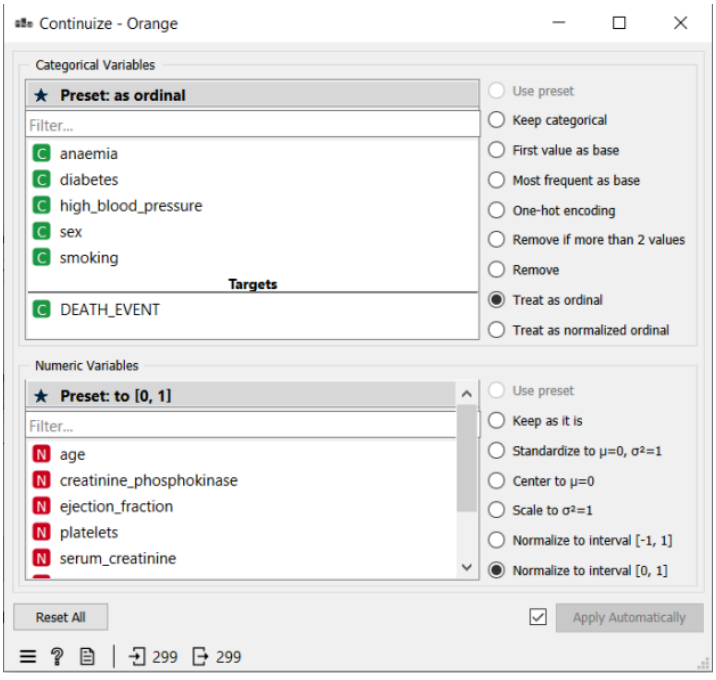
sex	Sieviete vai vīrietis.	categorical	0-1
smoking	Ja pacients smēķē vai nē.	categorical	0-1
time	Novērošanas periods.	numeric	4-285

Datu faila struktūra:

	DEATH_EVENT	age	anaemia	satinine_phosphokina	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	time
1	1	75.000	0	582	0	20	1	265000.00	1.90	130	1	0	
2	1	55.000	0	7861	0	38	0	263358.03	1.10	136	1	0	
3	1	65.000	0	146	0	20	0	162000.00	1.30	129	1	1	
4	1	50.000	1	111	0	20	0	210000.00	1.90	137	1	0	
5	1	65.000	1	160	1	20	0	327000.00	2.70	116	0	0	
6	1	90.000	1	47	0	40	1	204000.00	2.10	132	1	1	
7	1	75.000	1	246	0	15	0	127000.00	1.20	137	1	0	
8	1	60.000	1	315	1	60	0	454000.00	1.10	131	1	1	
9	1	65.000	0	157	0	65	0	263358.03	1.50	138	0	0	
10	1	80.000	1	123	0	35	1	388000.00	9.40	133	1	1	
11	1	75.000	1	81	0	38	1	368000.00	4.00	131	1	1	
12	1	62.000	0	231	0	25	1	253000.00	0.90	140	1	1	
13	1	45.000	1	981	0	30	0	136000.00	1.10	137	1	0	
14	1	50.000	1	168	0	38	1	276000.00	1.10	137	1	0	
15	0	49.000	1	80	0	30	1	427000.00	1.00	138	0	0	
16	1	82.000	1	379	0	50	0	47000.00	1.30	136	1	0	
17	1	87.000	1	149	0	38	0	262000.00	0.90	140	1	0	
18	1	45.000	0	582	0	14	0	166000.00	0.80	127	1	0	
19	1	70.000	1	125	0	25	1	237000.00	1.00	140	0	0	
20	1	48.000	1	582	1	55	0	87000.00	1.90	121	0	0	
21	0	65.000	1	52	0	25	1	276000.00	1.30	137	0	0	
22	1	65.000	1	128	1	30	1	297000.00	1.60	136	0	0	
23	1	68.000	1	220	0	35	1	289000.00	0.90	140	1	1	
24	0	53.000	0	63	1	60	0	368000.00	0.80	135	1	0	
25	1	75.000	0	582	1	30	1	263358.03	1.83	134	0	0	
26	1	80.000	0	148	1	38	0	149000.00	1.90	144	1	1	
27	1	95.000	1	112	0	40	1	196000.00	1.00	138	0	0	
28	1	70.000	0	122	1	45	1	284000.00	1.30	136	1	1	
29	1	58.000	1	60	0	38	0	153000.00	5.80	134	1	0	
30	1	82.000	0	70	1	30	0	200000.00	1.20	132	1	1	
31	1	94.000	0	582	1	38	1	263358.03	1.83	134	1	0	
32	1	85.000	0	23	0	45	0	360000.00	3.00	132	1	0	
33	1	50.000	1	249	1	35	1	319000.00	1.00	128	0	0	
34	0	50.000	1	159	1	30	0	302000.00	1.20	138	0	0	
35	1	65.000	0	94	1	50	1	188000.00	1.00	140	1	0	
36	1	69.000	0	582	1	35	0	228000.00	3.50	134	1	0	

*Informācija par trūkstošajām vai izlecošajām vērtībām:*Mūsu gadījumā nevienam atribūtam nav tukšu datu, un tas arī bija pārbaudīts ar “Impute” funkciju, bet ir izcelošie dati, tātad ir nepiciešams izmantot “Continuize” lai normalizēt datus:



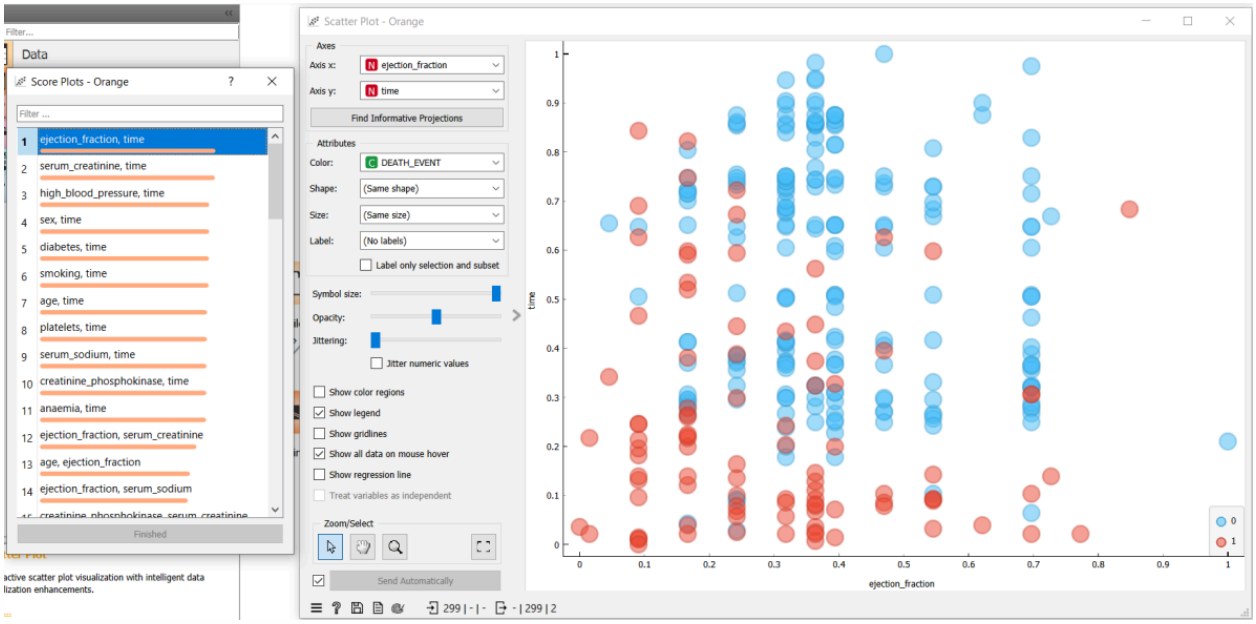


Šeit var novērot izmaiņas tabulā pēc normalizācijas:

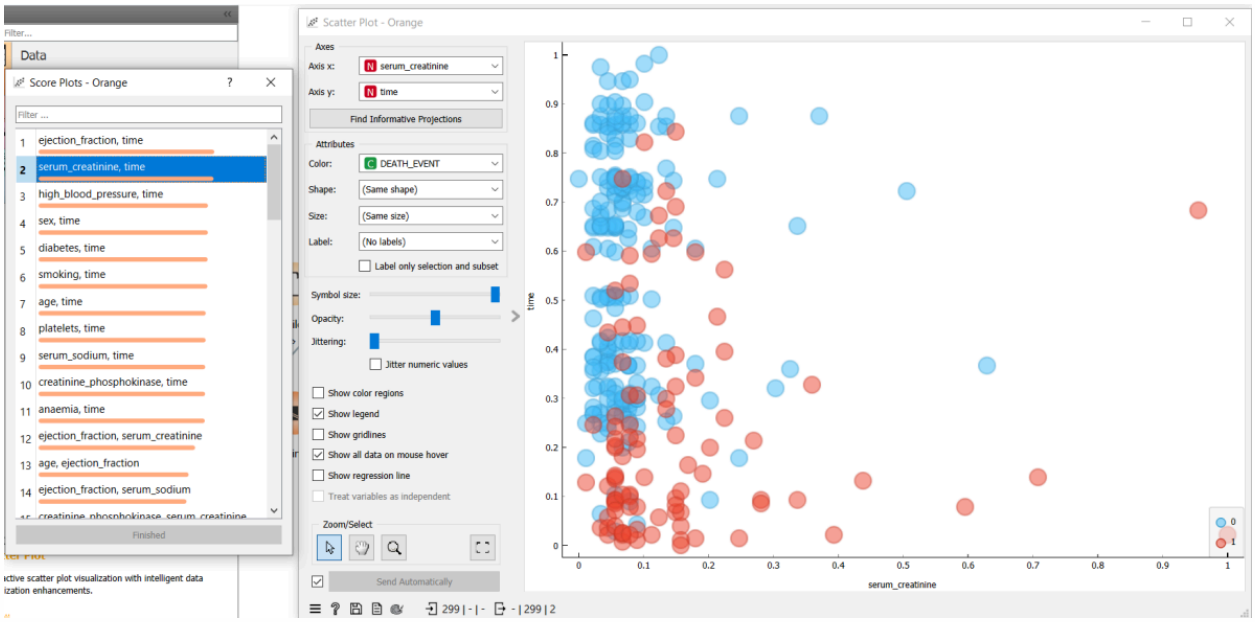
	DEATH_EVENT	age	anaemia	atinine_phosphokina	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	time
1	1	0.636364	0	0.0713192	0	0.0909091	1	0.290823	0.157303	0.485714	1	0	
2	1	0.272727	0	1	0	0.363636	0	0.288833	0.0674157	0.657143	1	0	0.00711
3	1	0.454545	0	0.0156928	0	0.0909091	0	0.16596	0.0898876	0.457143	1	1	0.0106
4	1	0.181818	1	0.0112274	0	0.0909091	0	0.224148	0.157303	0.685714	1	0	0.0106
5	1	0.454545	1	0.0174789	1	0.0909091	0	0.365984	0.247191	0.0857143	0	0	0.0142
6	1	0.909091	1	0.00306201	0	0.393939	1	0.216875	0.179775	0.542857	1	1	0.0142
7	1	0.636364	1	0.0284511	0	0.0151515	0	0.12353	0.0786517	0.685714	1	0	0.0213
8	1	0.363636	1	0.0372544	1	0.69697	0	0.519942	0.0674157	0.514286	1	1	0.0213
9	1	0.454545	0	0.0170962	0	0.772727	0	0.288833	0.11236	0.714286	0	0	0.0213
10	1	0.727273	1	0.0127584	0	0.318182	1	0.439932	1	0.571429	1	1	0.0213
11	1	0.636364	1	0.00739985	0	0.363636	1	0.415687	0.393258	0.514286	1	1	0.0213
12	1	0.4	0	0.0265374	0	0.166667	1	0.276276	0.0449438	0.771429	1	1	0.0213
13	1	0.0909091	1	0.122225	0	0.242424	0	0.134441	0.0674157	0.685714	1	0	0.024
14	1	0.181818	1	0.0184996	0	0.363636	1	0.304158	0.0674157	0.685714	1	0	0.024
15	0	0.163636	1	0.00727226	0	0.242424	1	0.487211	0.0561798	0.714286	0	0	0.0284
16	1	0.763636	1	0.0454197	0	0.545455	0	0.0265487	0.0898876	0.657143	1	0	0.0320
17	1	0.854545	1	0.0160755	0	0.363636	0	0.287186	0.0449438	0.771429	1	0	0.0355
18	1	0.0909091	0	0.0713192	0	0	0	0.170809	0.0337079	0.4	1	0	0.0355
19	1	0.545455	1	0.0130135	0	0.166667	1	0.25688	0.0561798	0.771429	0	0	0.0391
20	1	0.145455	1	0.0713192	1	0.621212	0	0.0750394	0.157303	0.228571	0	0	0.0391
21	0	0.454545	1	0.00369992	0	0.166667	1	0.304158	0.0898876	0.685714	0	0	0.0427
22	1	0.454545	1	0.0133963	1	0.242424	1	0.329616	0.123596	0.657143	0	0	0.0569
23	1	0.509091	1	0.025134	0	0.318182	1	0.319918	0.0449438	0.771429	1	1	0.0569
24	0	0.236364	0	0.00510334	1	0.69697	0	0.415687	0.0337079	0.628571	1	0	0.0640
25	1	0.636364	0	0.0713192	1	0.242424	1	0.288833	0.149438	0.6	0	0	0.0676
26	1	0.727273	0	0.0159479	1	0.363636	0	0.1502	0.157303	0.885714	1	1	0.0676
27	1	1	1	0.0113549	0	0.393939	1	0.207177	0.0561798	0.714286	0	0	0.0711
28	1	0.545455	0	0.0126308	1	0.469697	1	0.313856	0.0898876	0.657143	1	1	0.0782
29	1	0.327273	1	0.00472059	0	0.363636	0	0.155049	0.595506	0.6	1	0	0.0782
30	1	0.763636	0	0.00599643	1	0.242424	0	0.212026	0.0786517	0.542857	1	1	0.0782
31	1	0.981818	0	0.0713192	1	0.363636	1	0.288833	0.149438	0.6	1	0	0.0818
32	1	0.818182	0	0	0	0.469697	0	0.405989	0.280899	0.542857	1	0	0.0854
33	1	0.181818	1	0.0288339	1	0.318182	1	0.356286	0.0561798	0.428571	0	0	0.0854
34	0	0.181818	1	0.0173514	1	0.242424	0	0.335677	0.0786517	0.714286	0	0	0.088
35	1	0.454545	0	0.00905843	1	0.545455	1	0.197478	0.0561798	0.771429	1	0	0.088
36	1	0.527273	0	0.0713192	1	0.318182	0	0.245969	0.337079	0.6	1	0	0.0925

Datu kopas vizuālais un statistiskais atspoguļojums

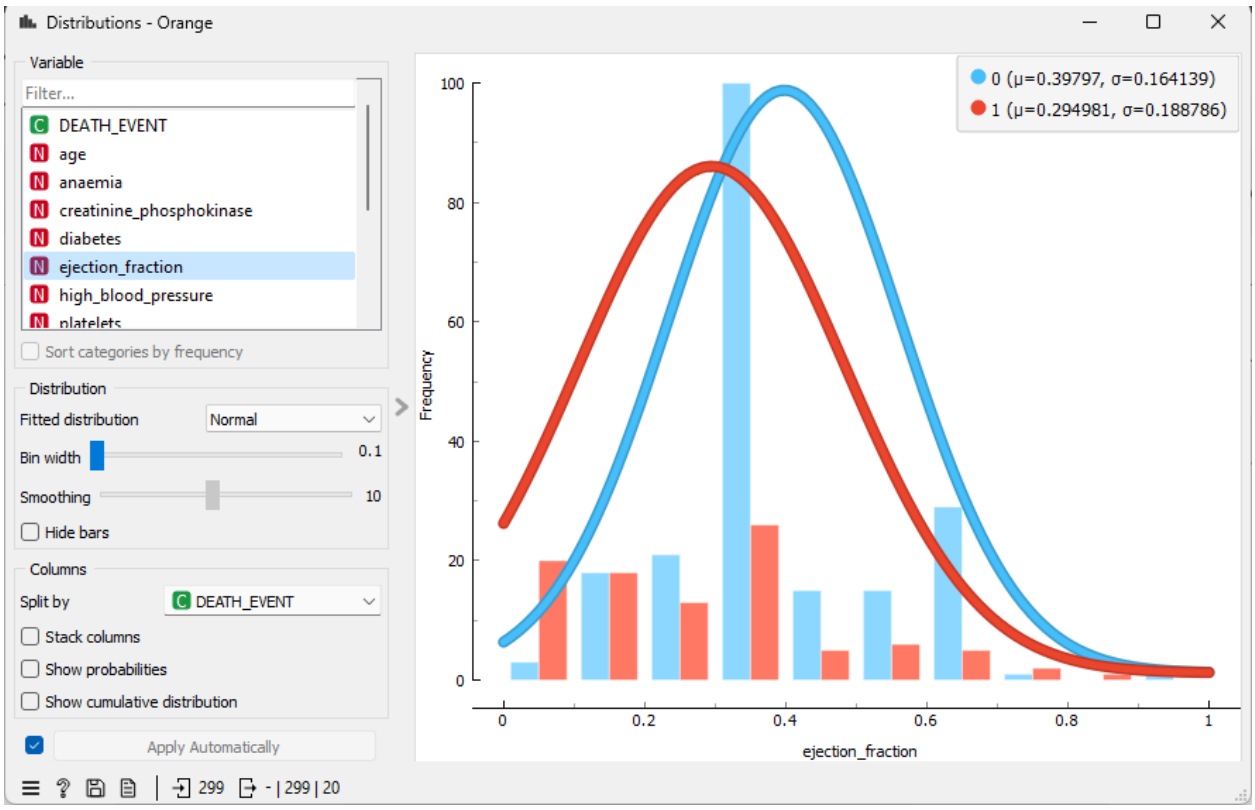
1) Izkliedes diagrammas ekrānuzņēmums:



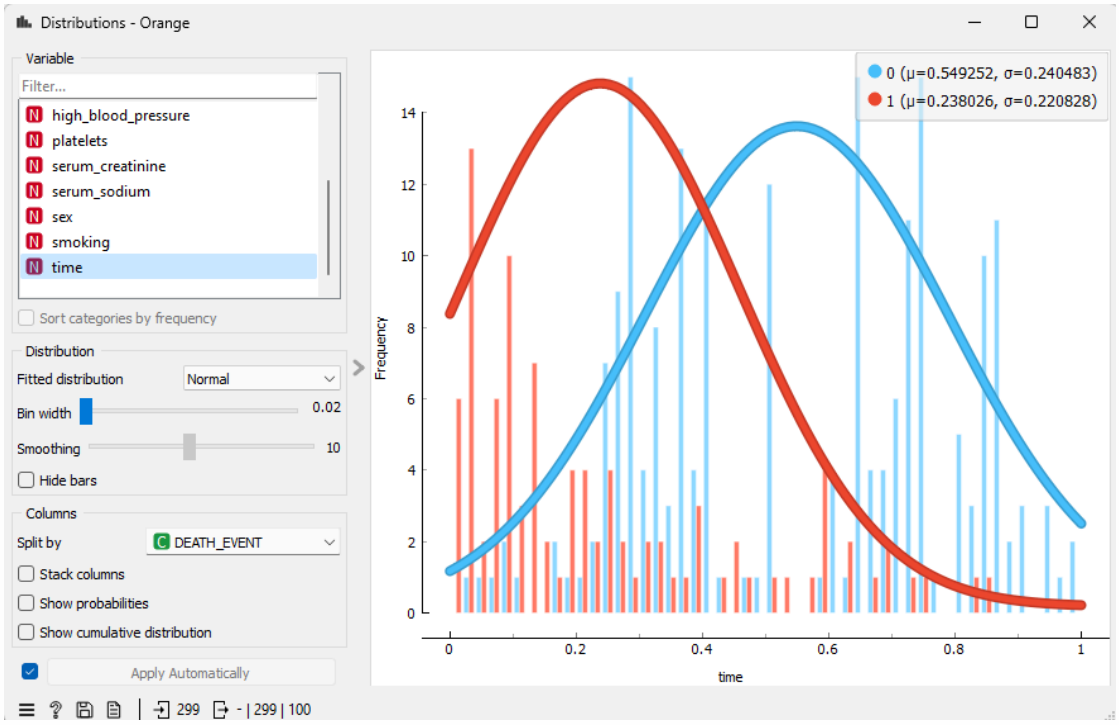
2) Izkliedes diagrammas ekrānuzņēmums:



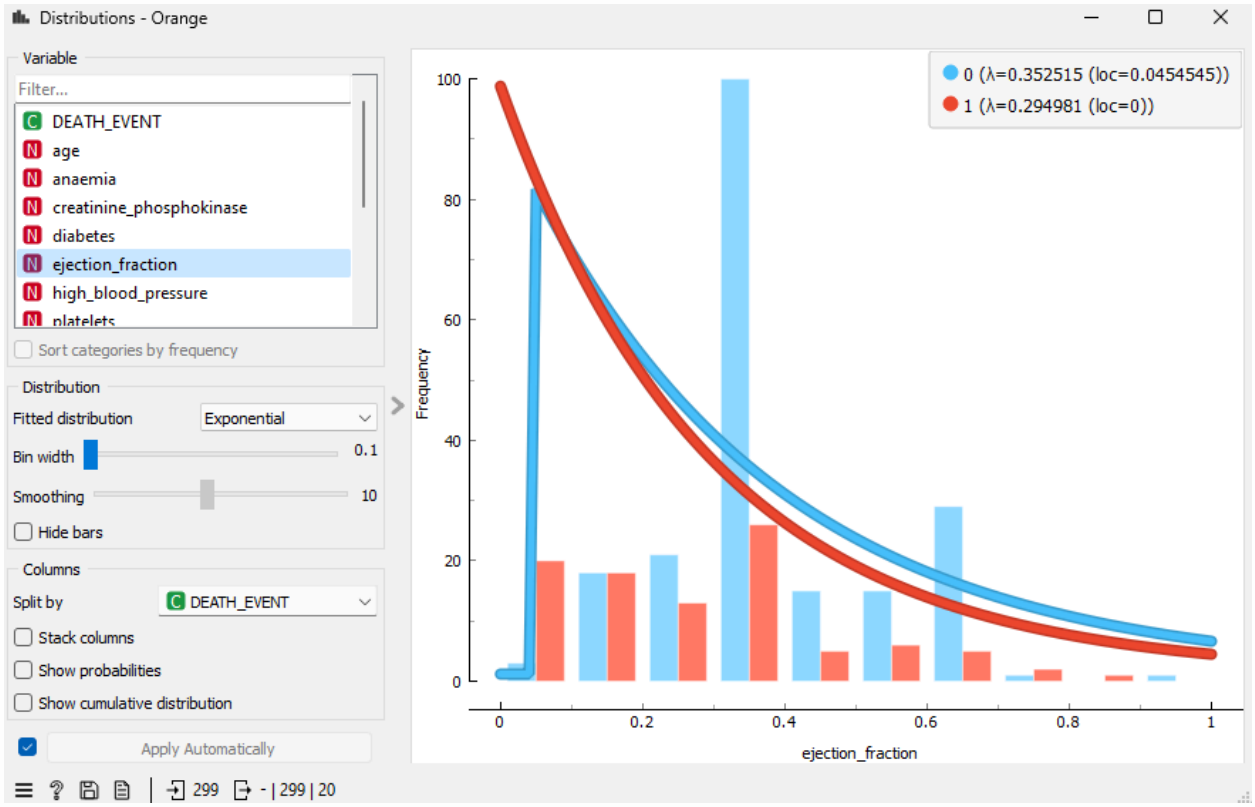
1) Histogrammas ekrānuzņēmums:



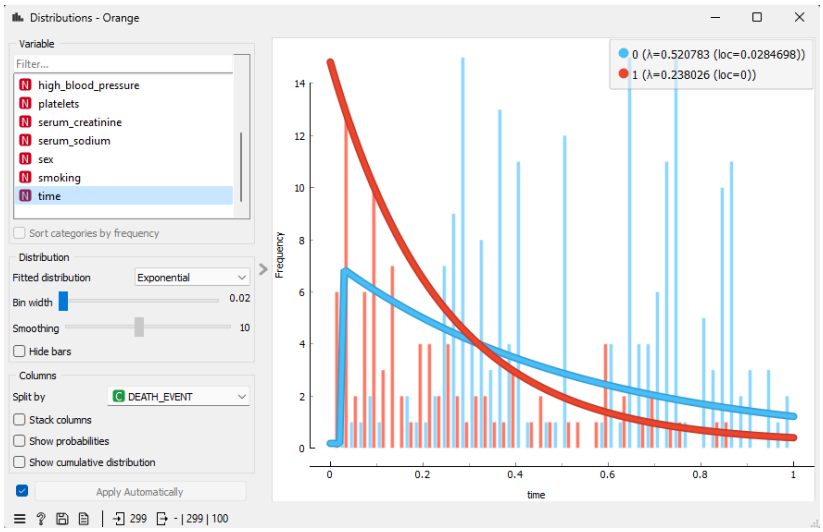
2) Histogrammas ekrānuzņēmums:



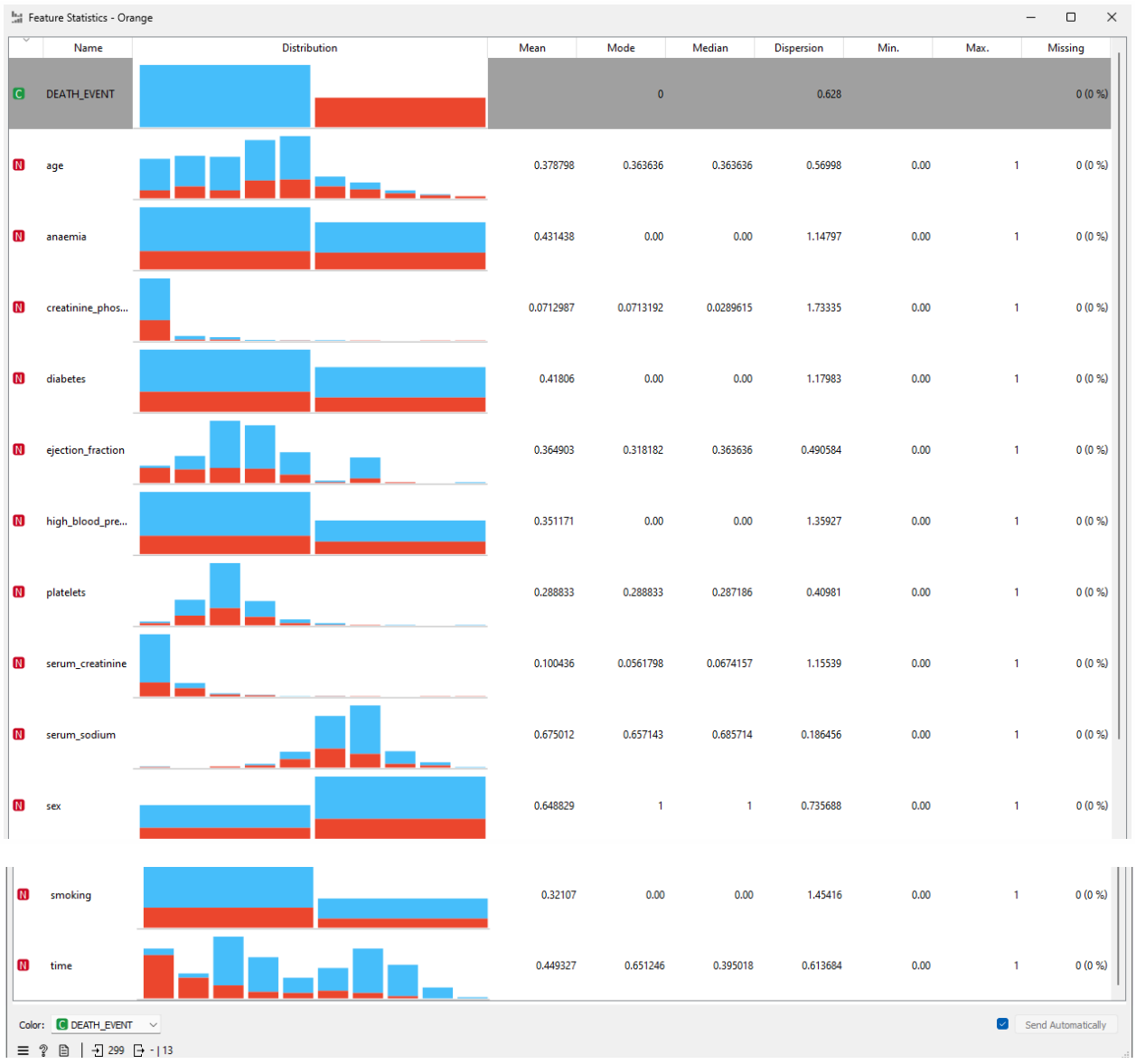
1) Ekrānuzņēmums ar pazimju sadalījumiem:



2) Ekrānuzņēmums ar pazimju sadalījumiem:



1) Ekrānuzņēmums ar statistiskajiem rādītājiem:



Atbildes uz jautājumiem

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

Mūsu gadījumā klases datu kopā nav līdzsvarota, jo vienas datu kopās dati ir divas reizēs lielāki par otru. Var redzēt, ka klases datu kopā “1” ir tikai 96 ieraksti un kopā “0” ir 203 ieraksti.

Vai datu vizuālais atspoguļojums ļauj redzēt datu struktūru?

Jā, var redzēt datu struktūru.

Cik datu grupējumus ir iespējams identificēt, pētot datu vizuālo atspoguļojumu?

Var paskatīties uz izkliedes diagrammas ekrānuzņēmumiem, var redzēt, ka tabulā datus var grupēt ar vismaz trīm parametriem, tie ir ejection_fraction un time, ka arī labākājs sadalījūms būs ar serum_creatininine un time, bet var arī kombinēt pilnīgi dažādus parametrus, kam nebūs tik labs sadalījums.

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

Ja vēlreiz paskatīsimies uz abiem Izkliedes diagrammas ekrānuzņēmumiem, tad var redzēt cik tuvu vai tālu viens no otra atrodas punkti. Tur būs redzams, ka punkti var atrasties gan tālu, gan tuvu viens otram, vienlaikus veidojot arī labāko iespējamo sadalījumu.

Secinājumi, kas izriet no statistisko rādītāju analīzes

No statistisko rādītāju analīzes mēs varām analizēt kopumā 13 diagrammas, no kurām viena ir “DEATH_EVENT”. Katrai diagrammai mēs varam aprēķināt mean, mode, median un dispersion. Un arī min. un max. parametri, ko mēs iestatām kā diapazonu no 0 līdz 1.

Analizējot iegūtos datus, var teikt, ka lielākais mean ir serum_sodium (0,675012), bet mazākais ir creatinine_phosphokinase (0,0712987). Lielākais mode ir paredzēts sex (1), savukārt trūkst diabetes, high_blood_pressure un smoking mode (0). Median aptuveni atkārto vērtības mode. Vislielākā dispersion ir creatinine_phosphokinase (1,73335), bet vismazākā - platelets (0,40981).

Zems mean nozīmē milzīgu plaisu starp datiem, to pārsvaru (0,0712987, kur viena kolonna ļoti atšķirās). Vērtība no 0,3 līdz 0,7 ir labāks datu sadalījums. To pierāda arī dispersion, kas ir lielākā kategorijā creatinine_phosphokinase (1,73335).

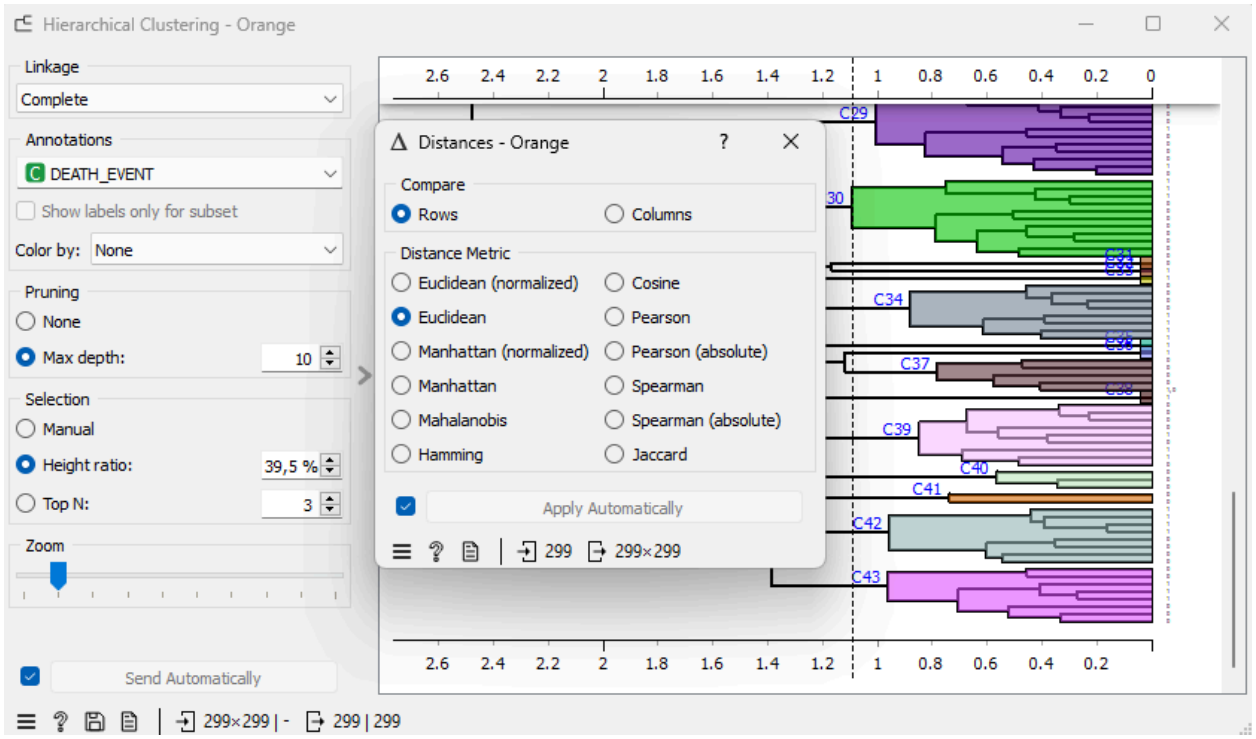
II daļa

Hierarhiskā klasterēšana

Orange rīkā pieejamo hiperparametru apraksts:

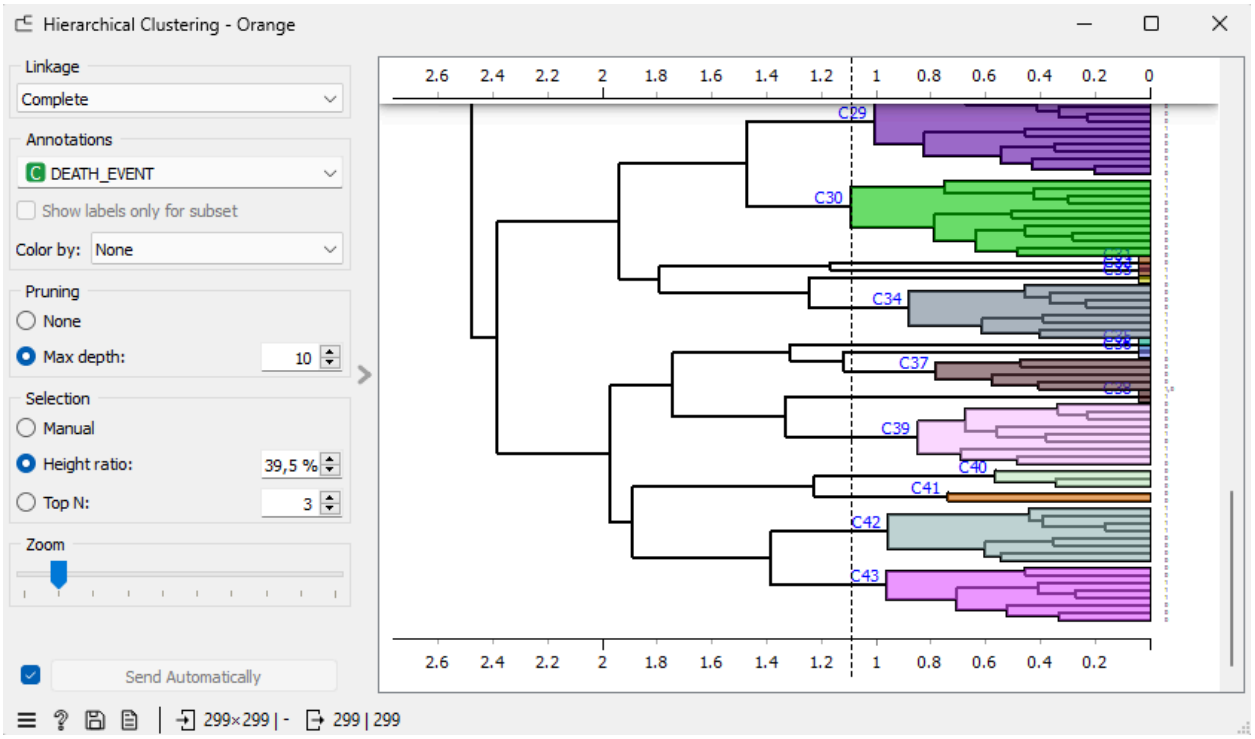
Hiperparametrs	Apraksts
Linkage - **Complete** , Single , Average , Weighted , Ward	Klastēru grupēšanas metode, kas aprēķinā attālumu starp klasteriem vai to sastāvdaļām. Izveleta metode - Complete - klastersu attālums tiek definēts kā maksimālais attālums starp to individuālajiem komponentiem.
Annotations - DEATH_EVENT	Tas nodrošina grafisko interfeisu, lai viegli varētu veikt datu analīzi, veikt klasterošanu, veidot modeļus un attēlot rezultātus vizuāli.
Pruning - **Max Depth(10)** , None	"Pruning" jeb "sēklēšana" mašīnmācīšanās kontekstā norāda uz procesu, kurā tiek samazināts koku modela kompleksitāte, novēršot pārāk lielu koku pārēršanos un tādējādi mazinot pārmācīšanos."Max Depth(10)" norāda uz maksimālo koka dziļumu vai līmeni, līdz kuram tiks veikta sēklēšana. Šis parametrs ierobežo kokam pieejamo dziļumu.
Selection - **Height Ratio** , Manual , Top N	Dažādi algoritmu izvēles kritēriji, ko varētu izmantot koku modela veidošanā vai sēklēšanā.Mūsu kritērijs norāda, ka algoritms izvēlas zarus vai mezglus koka struktūrā, pamatojoties uz to augstuma attiecību. Tas varētu nozīmēt, ka tiek atlasīti zari vai mezgli, kuru attiecība starp

	augstumu un platību ir noteikta intervālā vai ir lielāka par noteiktu slieksni.
Zoom	Tuvināšana vai attālināšana.
Distance Metric - Euclidean	Norāda uz izvēlēto attāluma mēru datu analīzē, konkrēti - Eiklīda attālumu.



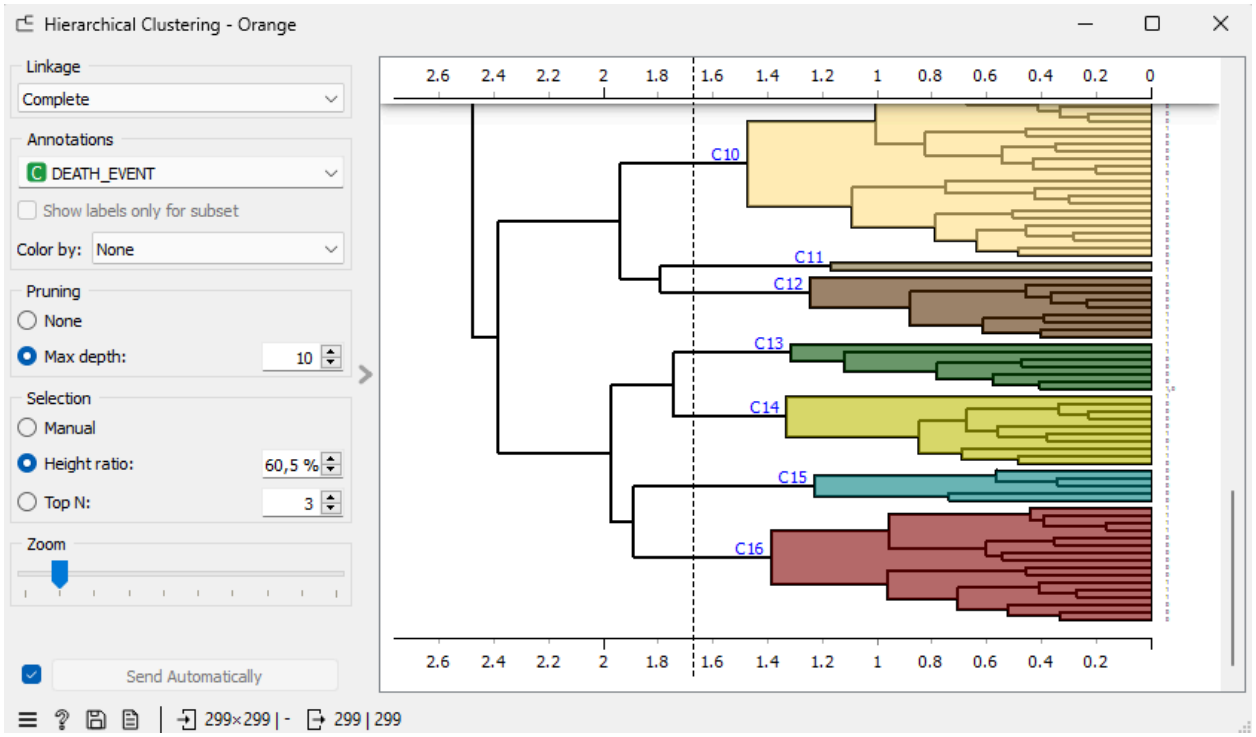
Eksperimentu apraksts

1)Heigh ratio 39.5%. Klasteru skaits - 43. C1 -11 data objekti. C2 - 6 data objekti. C3 - 10 data objekti. C4 - 3 data objekti. C5 - 12 data objekti. C6 - 12 data objekti. C7 - 5 data objekti. C8 - 1 data objekti. C9 - 10 data objekti. C10 - 13 data objekti. C11 - 1 data objekti. C12 - 9 data objekti. C13 - 15 data objekti. C14 - 6 data objekti. C15 - 1 data objekti. C15 - 13 data objekti. C16 - 3 data objekti. C17 - 14 data objekti. C18 - 17 data objekti. C19 - 4 data objekti. C20 - 1 data objekti. C21 - 7 data objekti. C22 - 12 data objekti. C23 - 4 data objekti. C24 - 16 data objekti. C25 - 1 data objekti. C26 - 7 data objekti. C27 - 12 data objekti. C28 - 12 data objekti. C29 - 11 data objekti. C30 - 1 data objekti. C31 - 1 data objekti. C32 - 1 data objekti. C33 - 8 data objekti. C34 - 1 data objekti. C35 - 1 data objekti. C36 - 6 data objekti. C37 -1 data objekti. C38 - 9 data objekti. C39 - 3 data objekti. C40 - 3 data objekti. C41 - 2 data objekti. C42 - 8 data objekti. C43 - 8 data objekti.

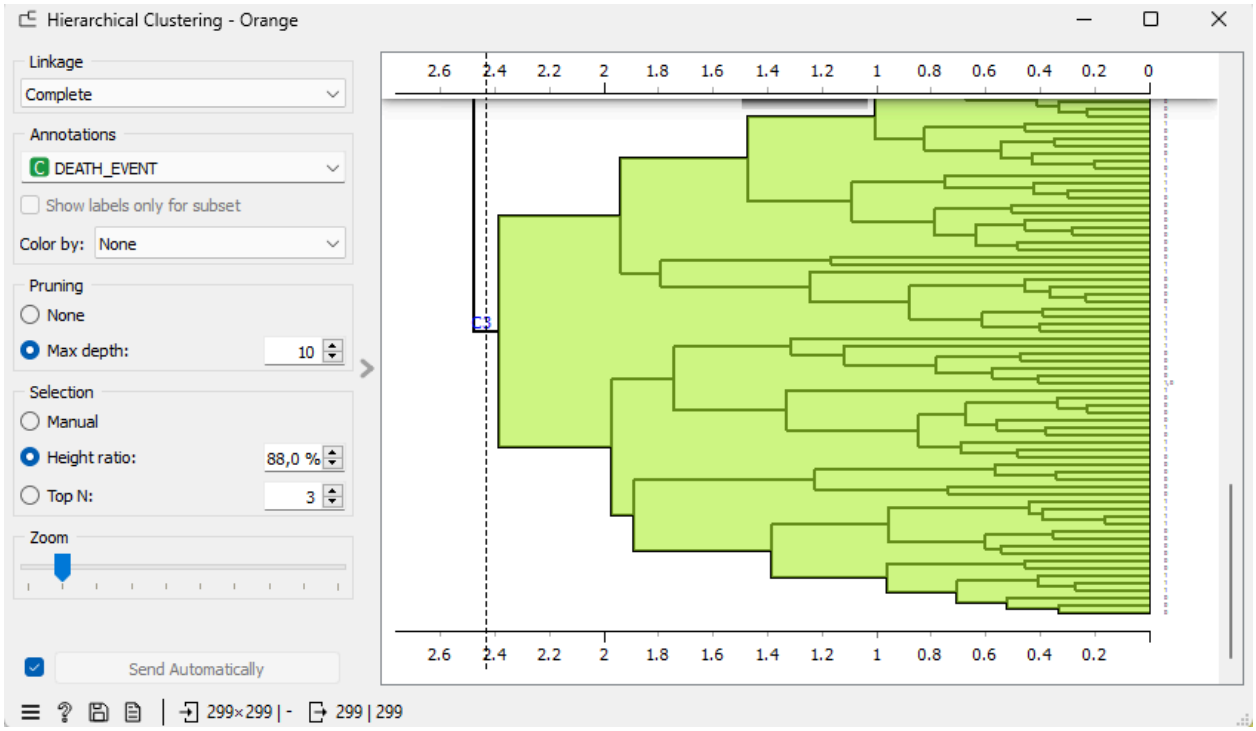


2)Heigh ratio 60.5%. Klasteru skaits - 16. C1 - 17 data objekti. C2 - 25 data objekti. C3 - 28 data objekti. C4 - 23 data objekti. C5 - 21 data objekti. C6 - 31 data objekti. C7 - 29 data objekti. C8 - 32 data objekti. C9 - 20 data objekti. C10 - 23 data objekti. C11 - 2 data objekti. C12 - 9 data

objekti. C13 - 8 data objekti. C14 - 10 data objekti. C15 - 5 data objekti. C16 - 16 data objekti.



3)Heigh ratio 88%. Klasteru skaits - 3. C1 - 93 data objekti. C2 - 133 data objekti. C3 - 73 data objekti.



Secinājumi no eksperimentiem:

Eksperimenta Nr. 1 (Heigh ratio 39.5%, Klasteru skaits - 43):

- Šajā eksperimentā, izmantojot 39.5% augstuma attiecību, tika atrasti 43 klasteri.
- Klastera izmēri mainās no vienas līdz 17 datu objektiem.
- Lai gan klasteru skaits ir liels un tajos ir dažāda izmēra datu objekti, šķiet, ka datu objekti ir sadalīti diezgan vienmērīgi.
- Var secināt, ka šajā eksperimentā ir izveidoti daudzi nelieli klasteri, kas var norādīt uz jutīgu klastera sadalījumu.

Eksperimenta Nr. 2 (Heigh ratio 60.5%, Klasteru skaits - 16):

- Ar 60.5% augstuma attiecību tika atrasti 16 klasteri.
- Klaustera izmēri svārstās no 2 līdz 32 datu objektiem.
- Šajā eksperimentā klasteru skaits ir mazāks nekā pirmajā, un tie ir arī lielāki.
- Izmantojot lielāku augstuma attiecību, klastera sadalījums šķiet jutīgāks un varētu norādīt uz skaidrākiem klastera grupējumiem.

Eksperimenta Nr. 3 (Heigh ratio 88%, Klasteru skaits - 3):

- Ar augstuma attiecību 88% tika atrasti tikai 3 klasteri.
- Šajos klasteros ir ievērojami lielāki datu objektu skaits, kas svārstās no 73 līdz 133.
- Šis eksperiments rāda, ka, izmantojot ļoti augstu augstuma attiecību, klastera sadalījums kļūst ļoti pēctecīgs un var rezultēt ar mazāku, bet lielāku grupējumu, kas ietver lielāku datu objektu skaitu.

Visi eksperimenti demonstrē, ka augstuma attiecība (height ratio) ietekmē klastera veidošanos un tā rezultātus.Ar augstāku augstuma attiecību ir tendence veidot mazāku klastera skaitu ar lielāku datu objektu skaitu katrā klasterī, bet tas var būtiski atšķirties atkarībā no datu struktūras un rakstura.Eksperimenti varētu norādīt uz nepieciešamību izmantot dažādas augstuma attiecības, lai atrastu optimālo klastera sadalījumu, kas atspoguļo datu struktūras raksturīgās īpašības.

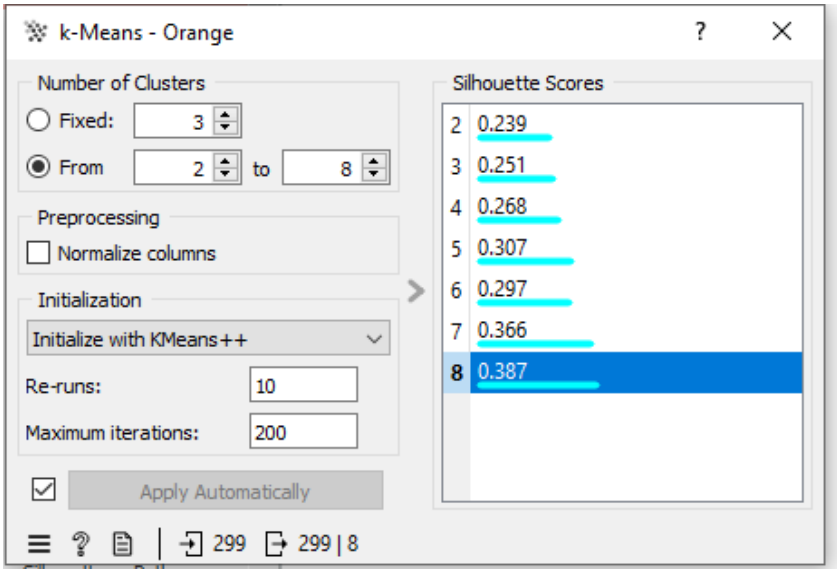
K-vidējo algoritms

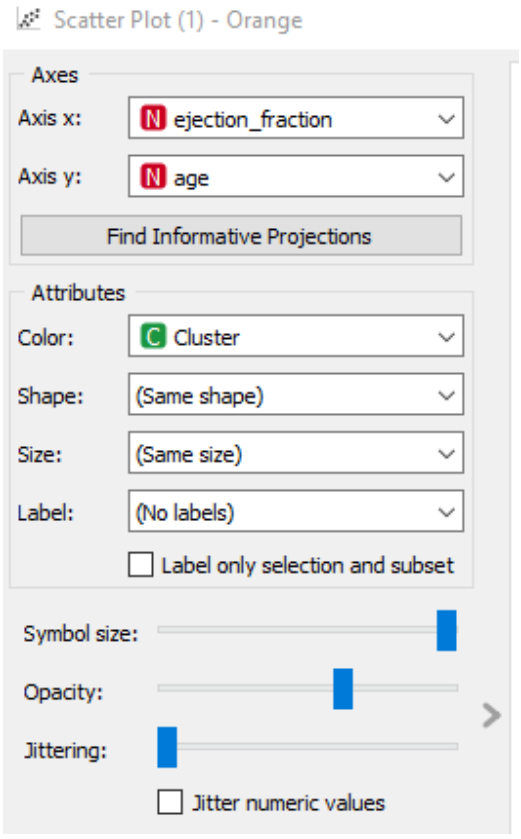
Orange rīkā pieejamo hiperparametru apraksts:

Hiperparametrs	Apraksts
Fixed	Algoritmu klasteri datus noteiktam klasteru skaitam.
From X To Y	Atlasītais klasteru diapazons, izmantojot silueta punktu skaitu
Preprocessing	Piedāvā metodi, kas var nepārtraukti kategorizēt mainīgos (ar vienu iezīmi uz vienu vērtību) un trūkstošās vērtības aprēķina ar vidējām vērtībām
Normalize columns	Ja opcija ir atlasīta, tad vidējais centrēts uz 0 un standartnovirze mērogota uz 1
Initialization method	Veids, kā algoritms sāk klasterēšanu
Initialize with KMeans++	Pirmais centrs tiek izvēlēts nejauši, nākamie tiek izvēlēti no atlikušajiem punktiem ar varbūtību, kas proporcionāla kvadrātveida attālumam no tuvākā centra
Random initialization (Gadījuminicializācija)	klasteri sākumā tiek piešķirti nejauši un pēc tam atjaunināti ar turpmākiem atkārtojumiem
Re-runs (Atkārtotas palaišanas)	Cik reizes algoritms tiek palaists no nejaušām sākotnējām pozīcijām

Maximum iterations (Maksimālais atkārtoto atkārtojumu skaits)	Maksimālais iterāciju skaits katrā algoritma izpildes reizē
Silhouette Scores	Ir mērs tam, cik līdzīgs objekts ir savam klasterim salīdzinājumā ar citiem klasteriem, un tam ir izšķiroša nozīme silueta laukuma izveidē.
Scatter Plot	Vizualizācija ar izpētes analīzi un inteliģentiem datu vizualizācijas uzlabojumiem.
Find Informative Projections	Šis līdzeklis novērtē atribūtu pārus pēc vidējās klasifikācijas precizitātes un atgriež labāko punktu pāri ar vienlaicīgu vizualizācijas atjauninājumu.
Attributes	Iestatiet etiķeti, formu un lielumu, lai atšķirtu punktus.

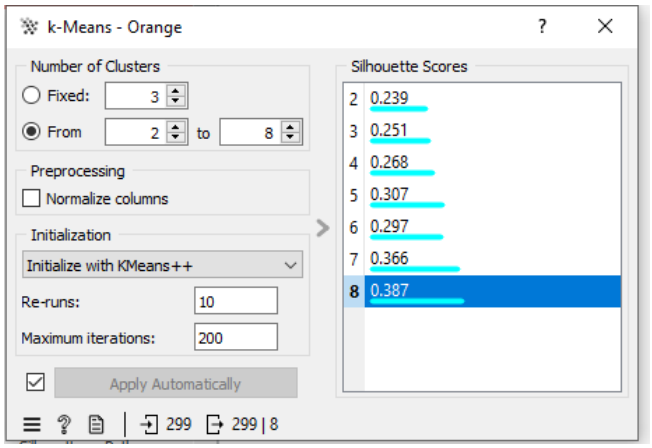
Ekrānuzņēmums ar algoritmam uzstādītajām hiperparametru vērtībām(k-Means un Scatter Plot)



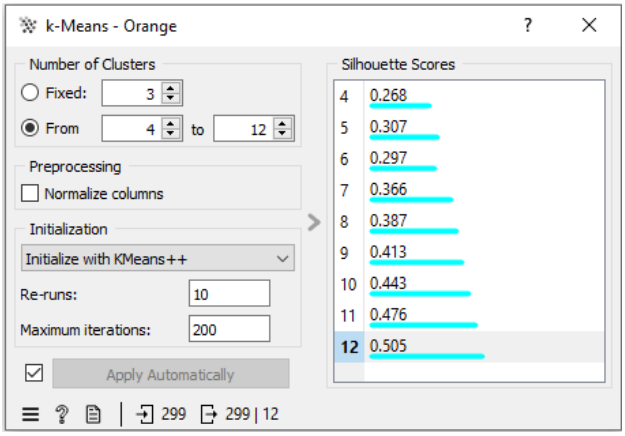


Eksperimentu apraksts

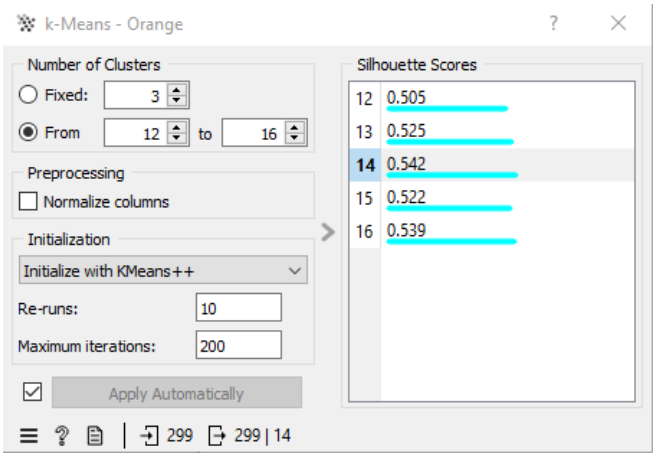
1) Pirmajā eksperimentā, atlasīts attālums starp 2 un 8 siluetiem. Var redzēt, ka 8 siluets ir tuvākpar to, lai nokļūt par klasteri ar silueta koeficienta vērtībai 0.387



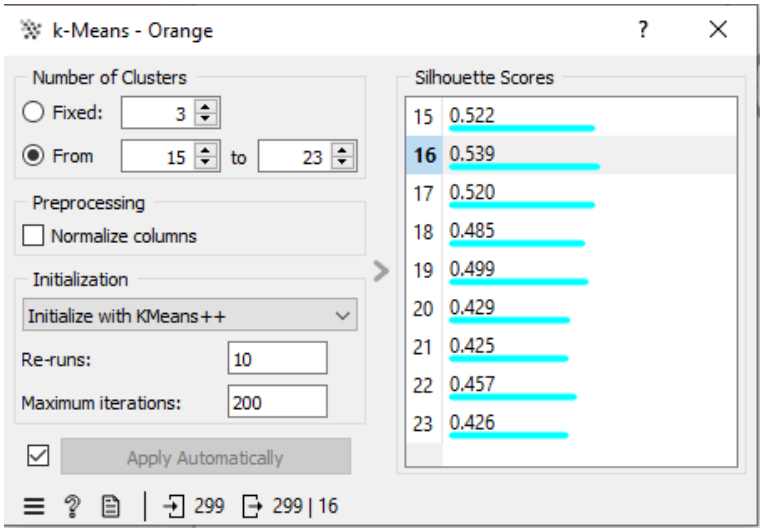
2) Otrajā eksperimentā, atlasīts attālums starp 4 un 12 siluetiem. Var redzēt, ka 12 siluets ir tuvākpar to, lai nokļūt par klasteri ar silueta koeficienta vērtībai 0.505



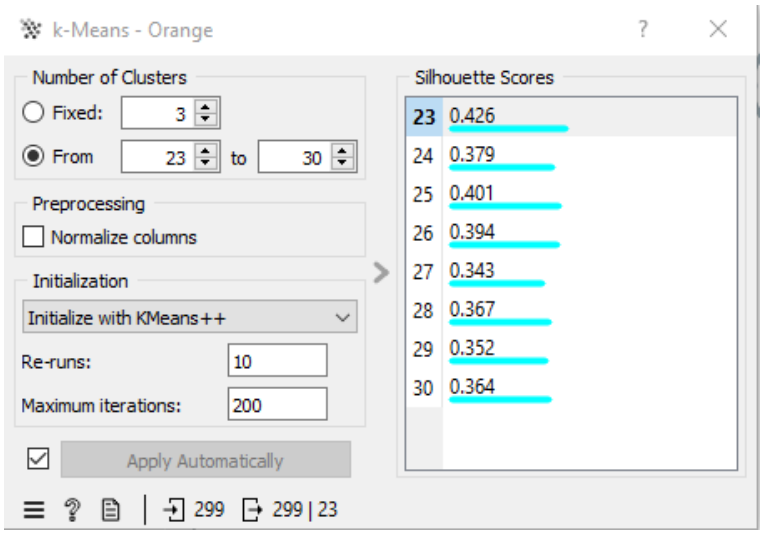
3) Trēšajā eksperimentā, atlasīts attālums starp 12 un 16 siluetiem. Var redzēt, ka 14 siluets ir tuvākpar to, lai nokļūt par klasteri ar silueta koeficienta vērtībai 0.542



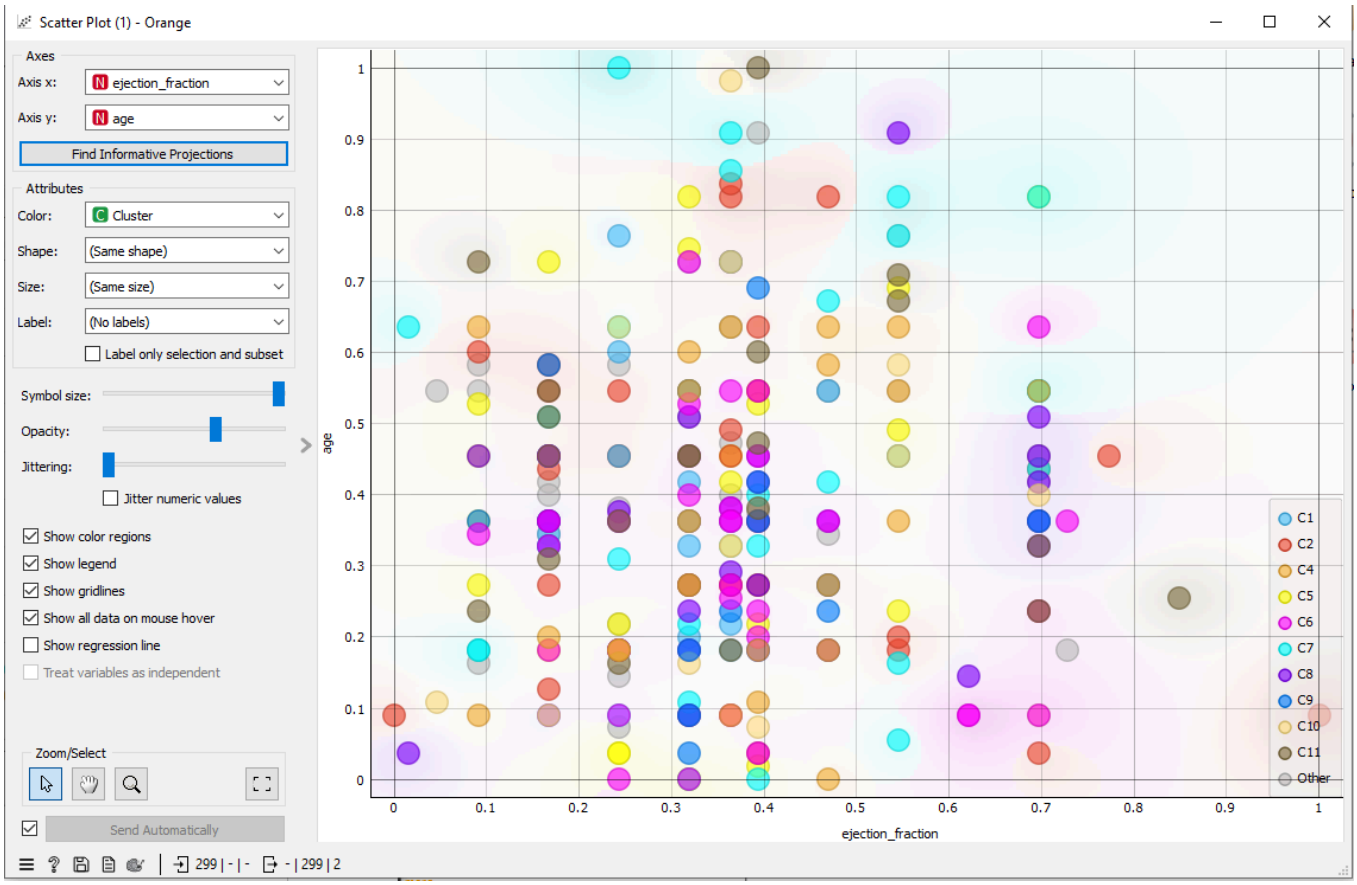
4) Ceturtājā eksperimentā, atlasīts attālums starp 15 un 23 siluetiem. Var redzēt, ka 16 siluets ir tuvākpar to, lai nokļūt par klasteri ar silueta koeficienta vērtībai 0.539



5) Piektājā eksperimentā, atlasīts attālums starp 23 un 30 siluetiem. Var redzēt, ka 16 siluets ir tuvākpar to, lai nokļūt par klasteri ar silueta koeficienta vērtībai 0.426



Izkliedes diagrammas ekrānuzņēmums ar trešo eksperimentu, kur ir labākā vērtība no visiem siluetu koeficientiem, kas ir 0,542.



Secinājumi no eksperimentiem:

Atlasītais klasteru diapazons ir tikai līdz 30, un eksperimenta laikā šis limits tika sasniegts. Var novērot, ka gandrīz visi silueta koeficienti bija diapazonā no [0,239 līdz 0,542], kas no vienības viens(tas ir augstvērtīga, kas norāda, ka kopas ir labi atdalītas un saliedētas), nav ļoti tālu un klastera atdalījums nebūs ideāls, bet nebūs ļoti slikti. Tas var būt saistīts ar datu struktūras sarežģītību, jo ļoti daudz dati ir svarīgi algoritmam, no 13 funkcijām loma "skip" tika mainīta tikai pie 2 funkcijām, un rezultātā palika 11(es noliku "skip" modifikāciju uz ”Time” un “Sex”, jo šie dati nav ļoti svarīgi), kas varētu ietekmēt eksperimenta rezultātus. Ja pie atlikušajām funkcijām pievienotu lomu "skip", tad silueta koeficients kļūst ļoti mazs, kas arī nav labi.

Diagrammai tika ņemti dati no trešā eksperimenta, kur tika sasniegta maksimālā silueta koeficienta vērtība 0,542. Y ass ir cilvēku vecums, un X ass ir viņu EF(“ejection fraction”) vērtība, kur norma tiek uzskatīta par 55 līdz 70 procentiem un to sasniedz mazāk nekā puse. Vairāk nekā puse vērtību atrodas starp 40 un 30 procentiem, kas liecina, ka sirds mazspēja pasliktinās.

Noslēguma secinājumi

Abi algoritmi - k-vidējais un hierarhiskā klasterēšana parāda spēju veidot klasterus, bet to efektivitāte ir atkarīga no sākotnēji izvēlētajiem parametriem un datu struktūras.K-vidējais algoritms parāda tendenci izveidot klasterus ar vienmērīgi sadalītiem datu objektiem, bet tas var būt jutīgs pret parametru izmaiņām un sākotnēji izvēlēto klaustera skaitu.hierarhiskā klasterēšana ir mazāk jutīga pret sākotnējiem parametriem un parasti izveido klasterus ar atšķirīgu izmēru un formu, kas var būt noderīgi, ja dati ir sarežģītāki un struktūrā ir daudzveidīgums.Abi algoritmi var būt noderīgi datu analīzē, bet ir svarīgi uzmanīgi izvērtēt to rezultātus un pielāgot parametrus atbilstoši konkrētajiem datiem un analīzes mērķiem.

III daļa

Izvēlēto algoritmu apraksts

Pirmā algoritma nosaukums:

Loģistiskā regresija

Pirmā algoritma apraksts:

Logistiskā regresija tiek paredzēta diskreta rezultāta varbūtība, ņemot vērā ievades mainīgo. Visbiežākā logistiskā regresija modelē bināru rezultātu, kas var pieņemt divas vērtības, piemēram, true/false, yes/no, utt. Ir arī iespējams modelēt situacijas, kur ir vairākas diskretas iznākumu vērtības. Loģistiskā regresija biežāk izmanto, kad nevajag risināt kaut ko grutāko, ta izmanto tikai , kad datu kopa ir daudz ipašībās ar bināro vertības, nepārtrauktos mainīgos. Musu gadijuma mes to ņemam, jo mes varam tos izmantot ar musu datu baze bez jebkadam grutībam. Eksperimentu vide mes uzzinas, ka ši metode ir visprecīzākais no visiem musu gadijuma.

Otrā algoritma nosaukums:

KNN algoritms

Otrā algoritma apraksts:

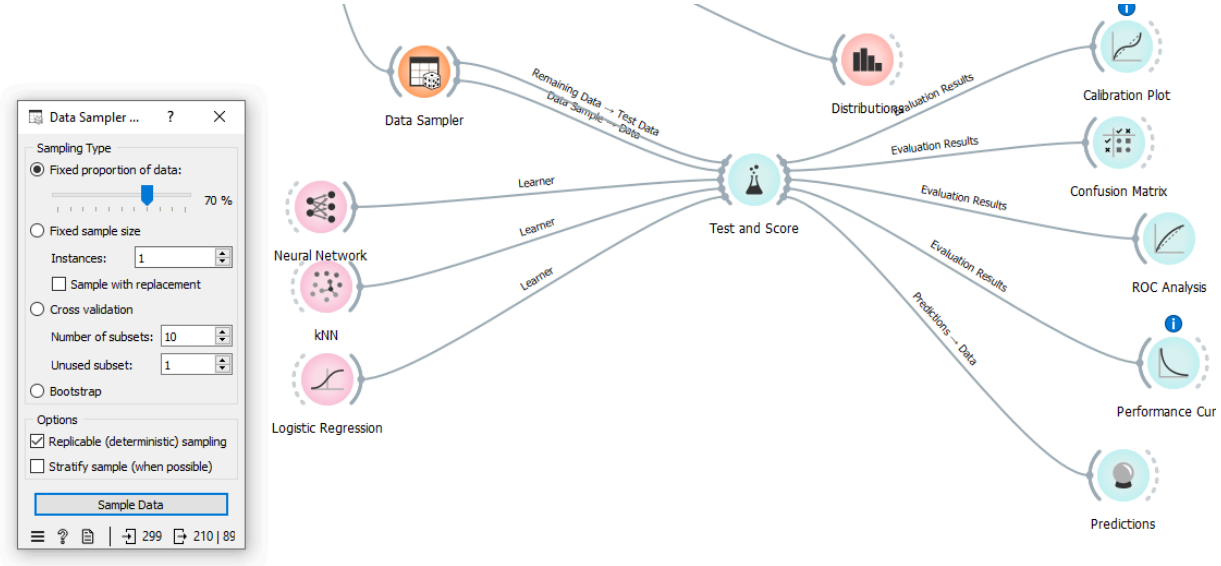
Klasifikators kNN ir bezparametriskais uzraudzītās mācīšanās algoritms, kas balstās uz tuvuma izmantošanu, lai noteiktu atsevišķu datu punktu grupējumu vai prognozētu to klasifikāciju. Visbiežak to izmantots regresijas un klasifikācijas uzdevumiem. Lielākoties to izmanto kā klasifikācijas algoritmu, ņemot vērā faktu, ka līdzīgi punkti būs tuvu viens otram. Algoritmam ir hiperparametri, piemēram, kaimiņu skaits, mērījuma metrika: Euclidean, Manhattan, Maximal, Mahalanobis, un svars: Uniform or Distance. Izvēles motivacija šīm algoritma ir viņā labāka pazistamība un ta bija apskatits lekcija laika, ka ari ortusa vide.

Hiperparametru apraksts

Hiperparameters	Apraksts un vērtības
Mākslīgo neironu tīkli	
Neurons in hidden layers Neironu skaits sleptajos slanos	i-tais elements, kas apzīmē neironu skaitu j-tajā slēptajā slānī. Piem. neironu tīklu ar 3 slāņiem var definēt kā 100, 100, 100, tas nozime ka katra slani ir 100 neironi.
Activation	Aktivizacijas izvele Identity,Logistic,tahn,ReLu
Solver	Optimizatora izvele, L-BFGS-B,SDG,Adam
Regularization, a Regularizacija, alfa	Koeficients kā soda termiņš funkcijai , šim parametram pieejamās vērtības ir no 0,0000 līdz 1000
Maximal number of iterations Maksimālais iterāciju skaits	Maksimālais atkārtojumu skaits
kNN algoritms	

Number of neighbors	Kaimiņu skaits, ko algoritms apskatīs, veicot izvēli, datu tips Integer
Metric	Mērīšanās sistēmās , ar kuru palīdzību mes apskatisim kNN algorimu, 4 tipi : Euclidean, Manhattan, Chebyshev, Mahalanobis
Weight	Svara tips - Uniform, By Distances
Loģistiskā regresija	
Regularization type	Regularizacijas tipa izvele, Lasso (L1), Ridge(L2), None
Strength	Iestatiet regularizacijas stiprumu (noklusējums ir C=1).

Informācija par testa un apmācības datu kopām



Datu objektu skaits apmācības datu kopā:

210

Datu objektu % proporcija apmācības datu kopā:

70%

Klases iezīme	Datu objektu skaits apmācības datu kopā	Datu objektu % proporcija apmācības datu kopā
0	138	66%
1	72	34%

Datu objektu skaits testa datu kopā:

89

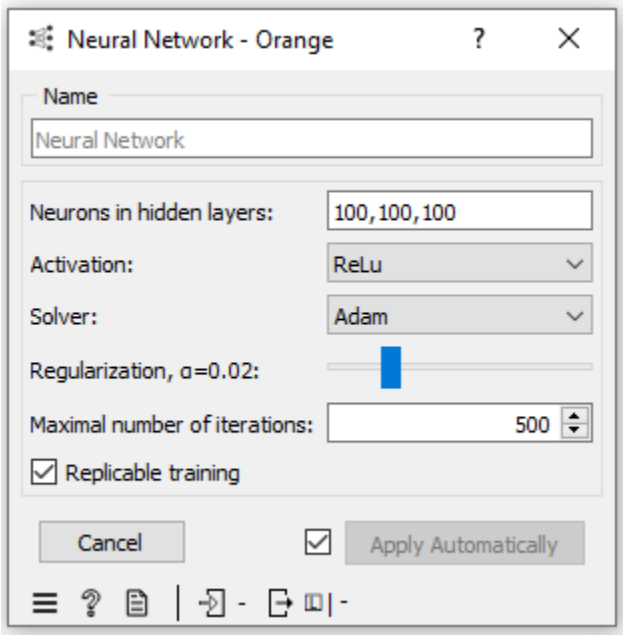
Datu objektu % proporcija testa datu kopā:

30%

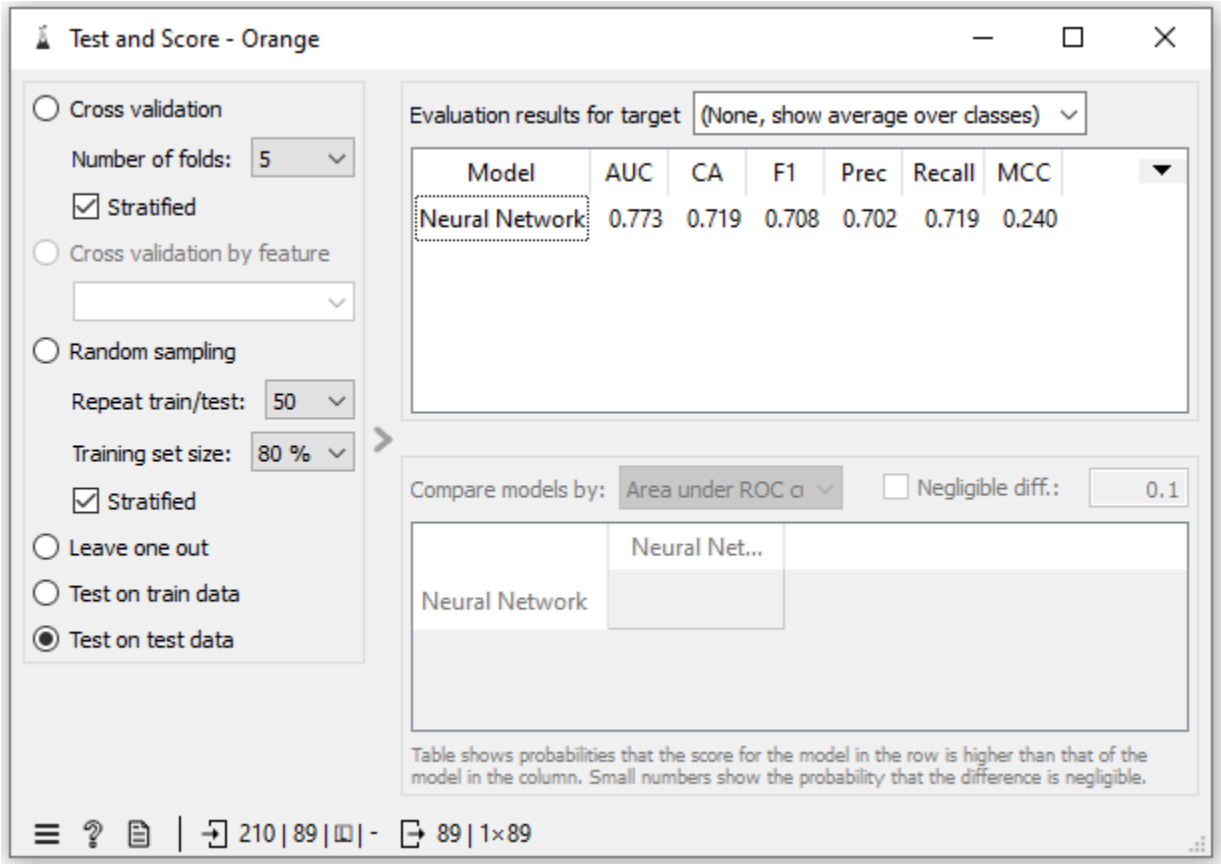
Klases iezīme	Datu objektu skaits testa datu kopā	Datu objektu % proporcija testa datu kopā
0	65	73%
1	24	27%

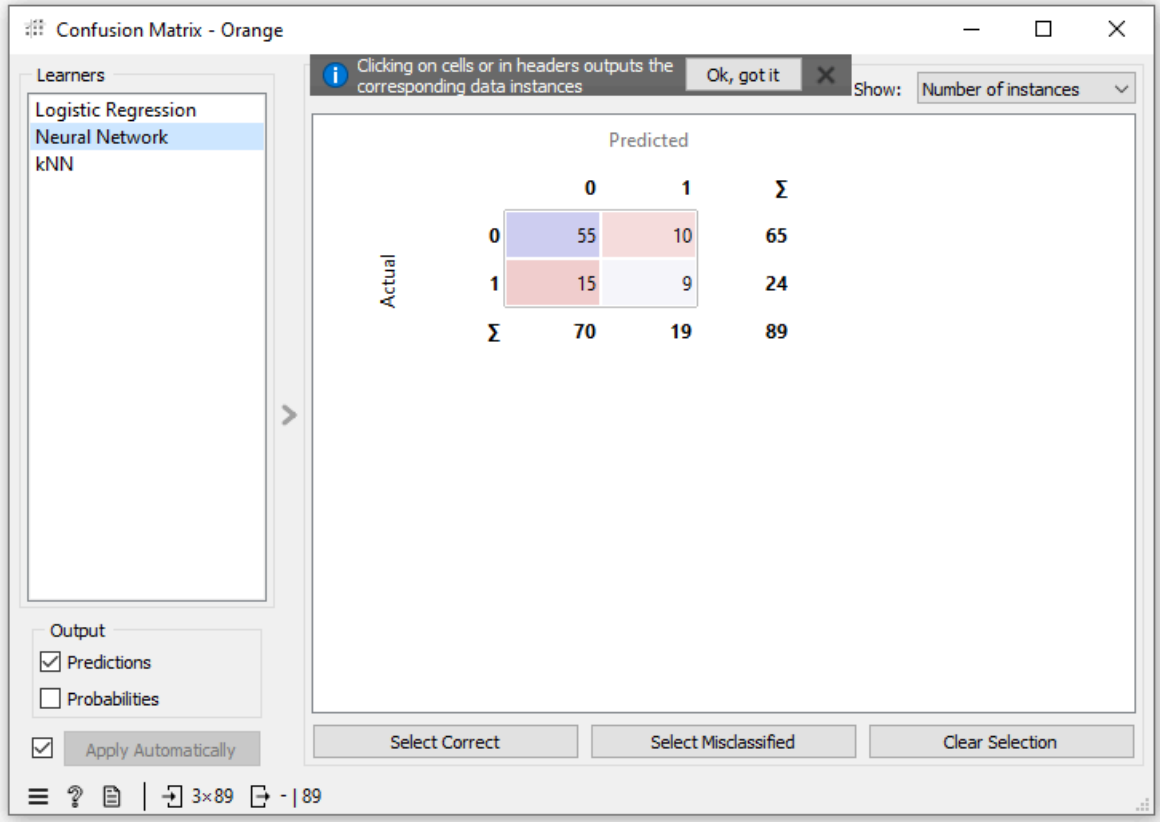
Eksperimenti ar mākslīgo neironu tīklu

Eksperiments	Hiperparametru vērtības
1.eksperiments	Neurons in hidden layers = 100,100,100 Regularization, a = 0.02 Maximal number of iterations = 500
2.eksperiments	Neurons in hidden layers = 100,200,100 Regularization, a = 0.0001 Maximal number of iterations = 500
3.eksperiments	Neurons in hidden layers = 100,200,100 Regularization, a = 0.0001 Maximal number of iterations = 1000/20

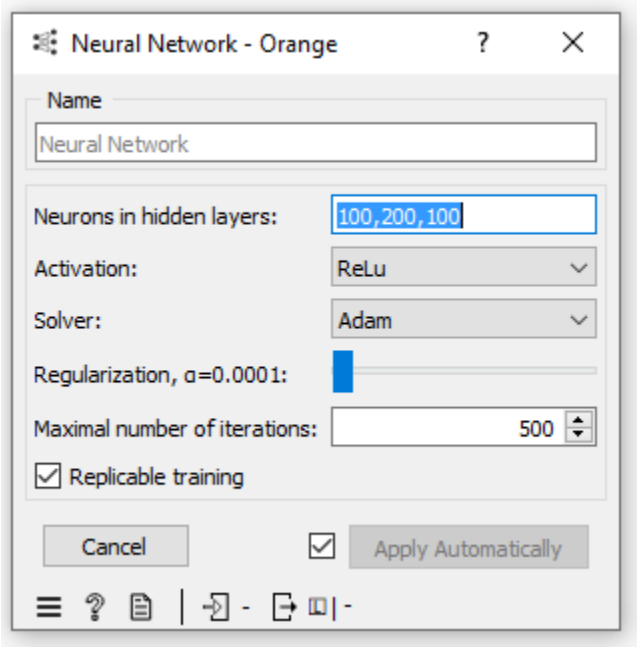


ekrānuzņēmums 1.eksperimenta hiperparametru vērtībām

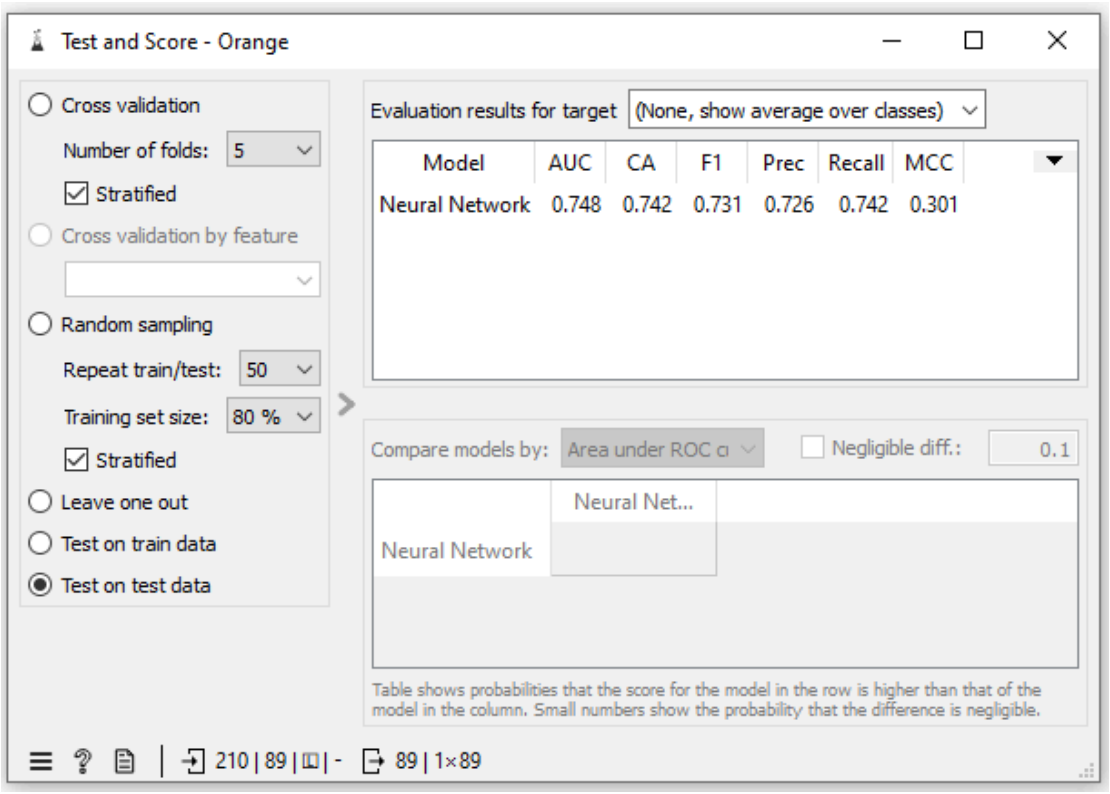


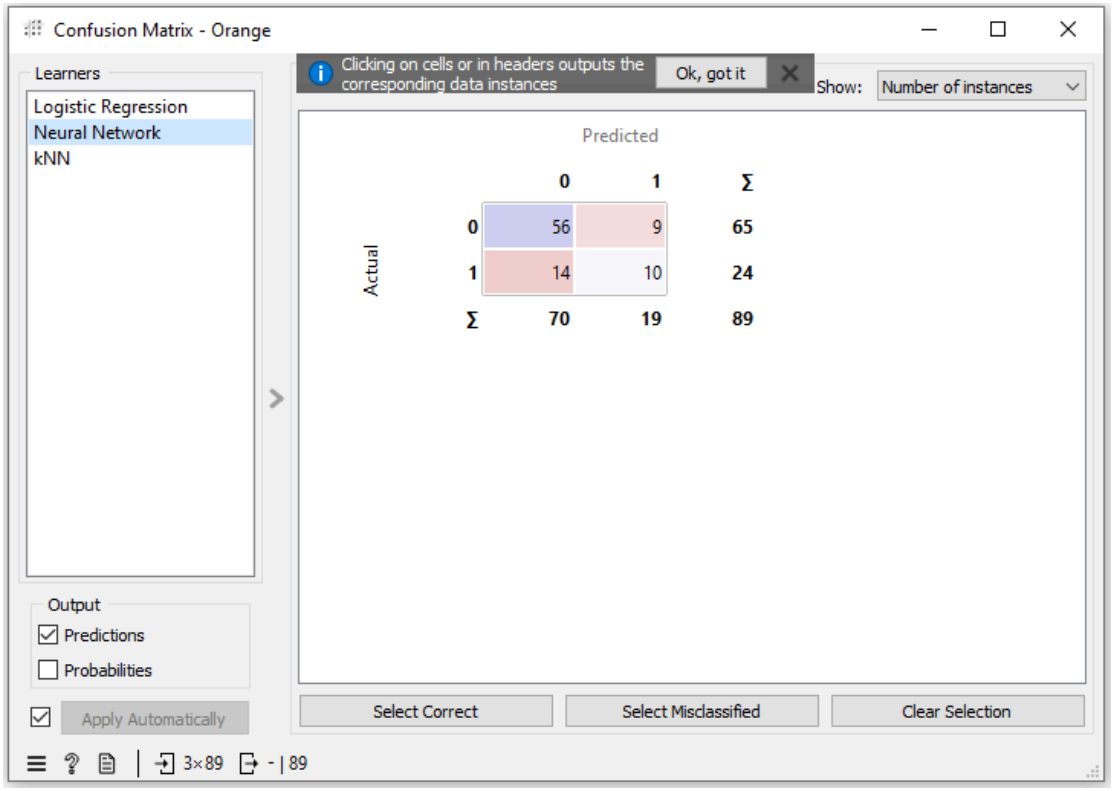


ekrānuzņēmumi 1.esperimenta veiktspējas metrikām

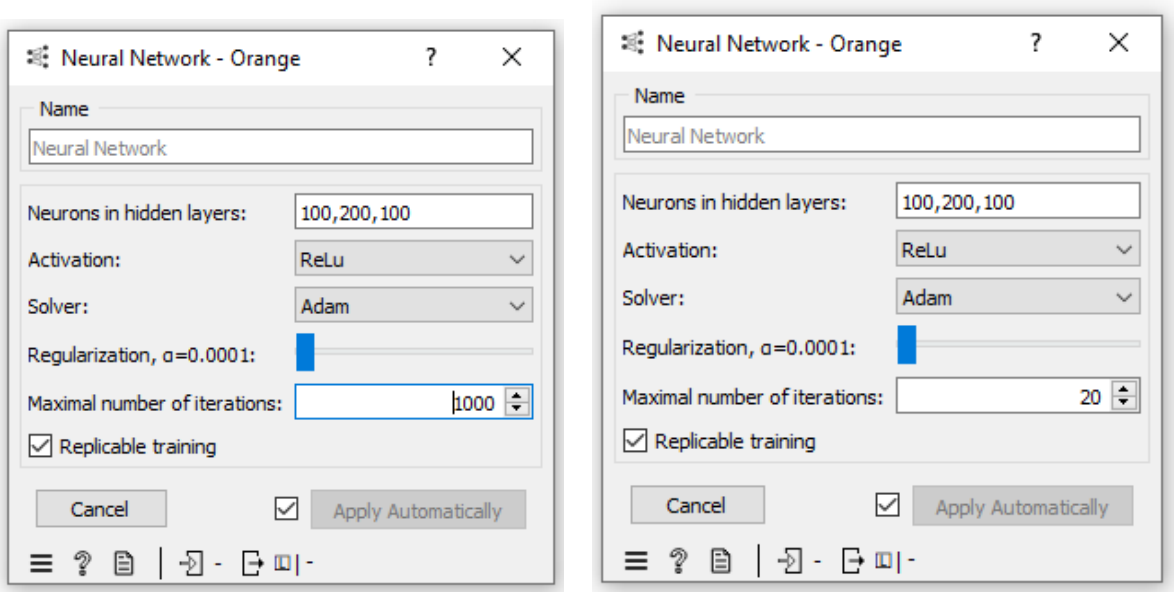


ekrānuzņēmums 2.eksperimenta hiperparametru vērtībām

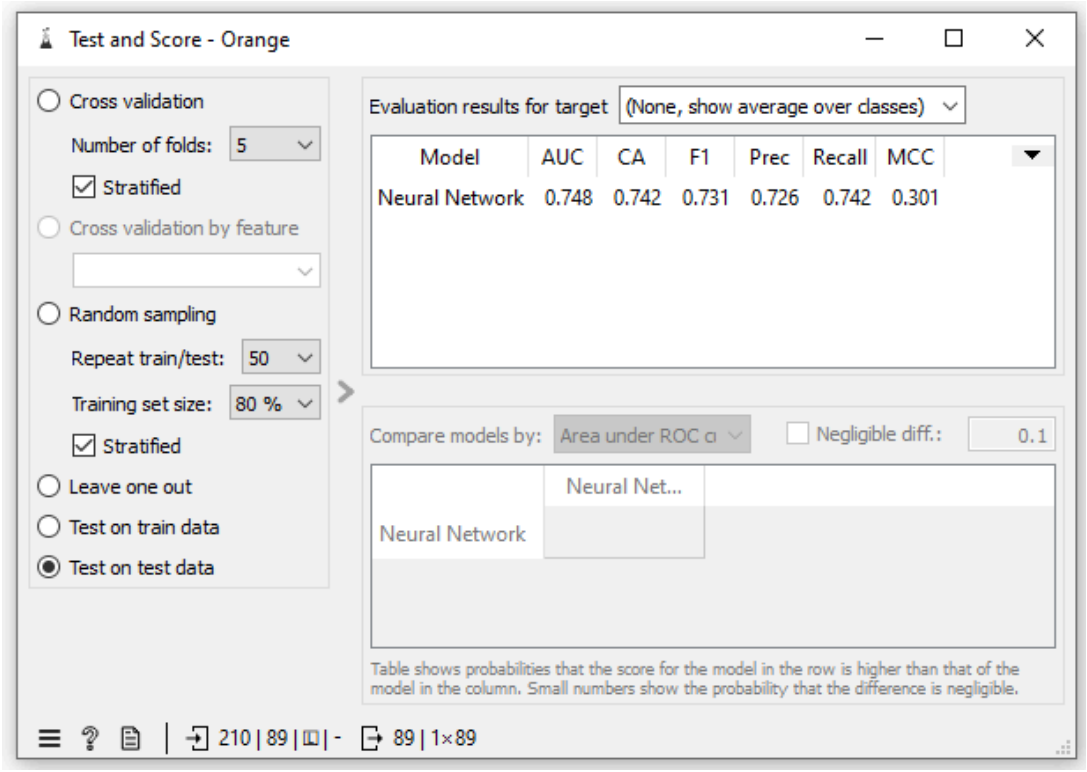


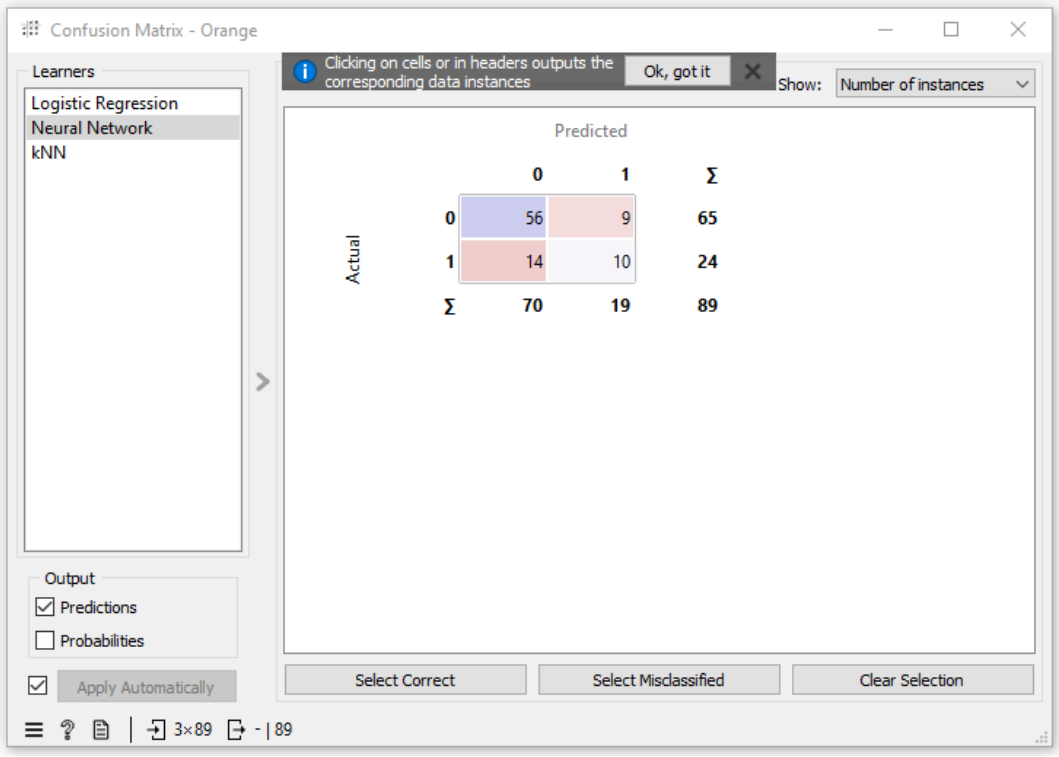


ekrānuzņēmums 2.esperimenta veiktspējas metrikām

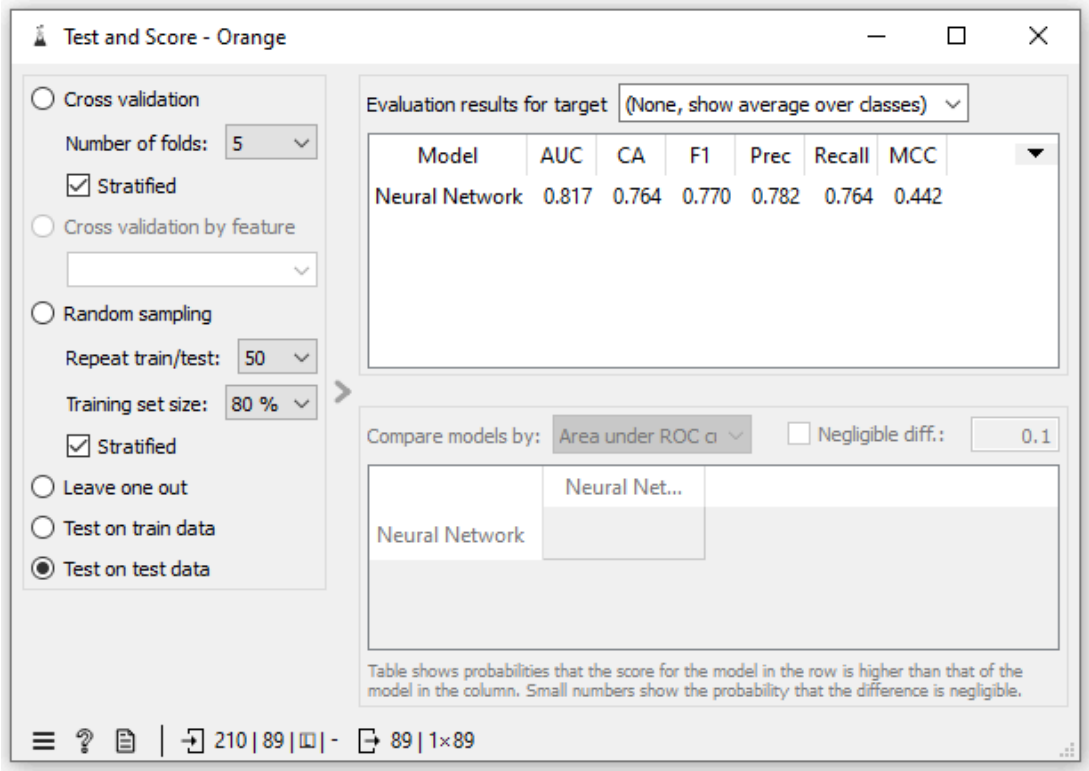


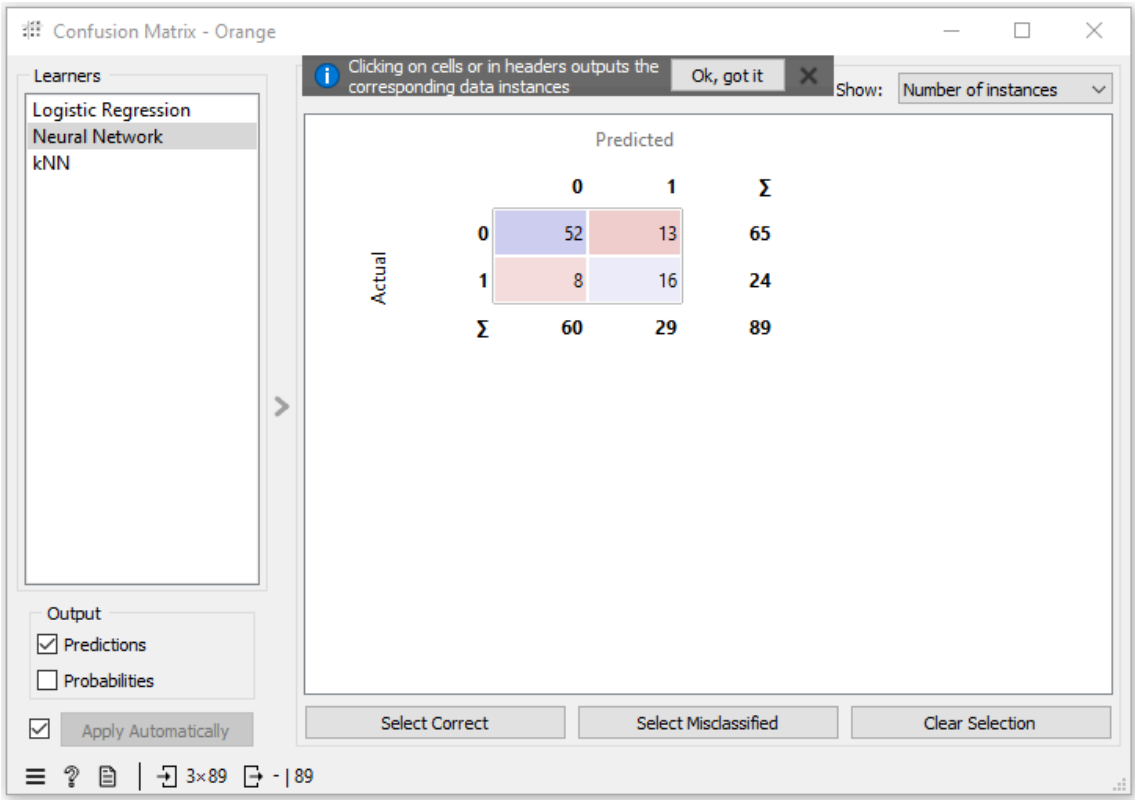
ekrānuzņēmumi 3.eksperimenta hiperparametru vērtībām





ekrānuzņēmumi 3.esperimenta veiktspējas metrikām pie max. iter. =1000





ekrānuzņēmumi 3.esperimenta veiktspējas metrikām pie max. iter. =20

Secinājumi no eksperimentiem:

Eksperimentējot ar šo algoritmu, redzēt, ka tā precizitāte ļoti, ļoti lielā mērā ir atkarīga no neironu skaita slāņos, kas ir pamanāms, to pirmā(ekrānuzņēmumi 1.esperimenta veiktspējas metrikām) un otrā eksperimenta(ekrānuzņēmumi 2.esperimenta spējas metrikām) var saprast, ka rezultāts kļūst precīzāks, ja palielināsim regularizācijas vērtību un neironu skaitu slēptajos slāņos. Pastāv iespēja, ka sākotnējo vērtību dēļ, kuras izvēlējāmies 100,100,100, slēptajos slāņos ir pārāk daudz neironu, tāpēc varam izdarīt daļēji nepareizu secinājumu, ka neironu palielināšana slēptajos slāņos noteikti uzlabos rezultātu. , taču šī ideja mūsu eksperimentos nav atspoguļota, tāpēc nolēmām to ievietot secinājumos.

Mūsu gadījumā mēs varam redzēt, ka visprecīzāku rezultatu ieguvām rezultātu ar vērtībām 100 200 100, izmantojot regulējumu 0,0001 un ar iterāciju skaitu 20(ekrānuzņēmums 3.esperimenta veiktspējas metrikām), kas var šķist pretrunā, jo ar lielāku iterāciju skaitu mūsu algoritms kļūst mazāk precīzs vai pastav tads pats, salidzinot rezultatu 3. Eksperimenta 1. Dalai un 2. Eksperimentu varam secināt ka pec kada momenta iteraciju skaits sak pasliktinat algoritmu un vel pec tam nekas nedara, lai gan tam vajadzētu būt otrādi.

Izdarot vispārīgus secinājumus par šo algoritmu, mēs varam teikt, ka tas ir diezgan precīzs, taču tas prasa diezgan lielus resursus, ja veic lielu skaitu iterāciju vai veido daudzus

slāņus vai neironus slāņos. Var teikt, ka mūsu eksperimentos šis algoritms darbojās labi, ar vidējo precizitāti aptuveni 75 procenti.

Testēšanai izvēlētais modelis:

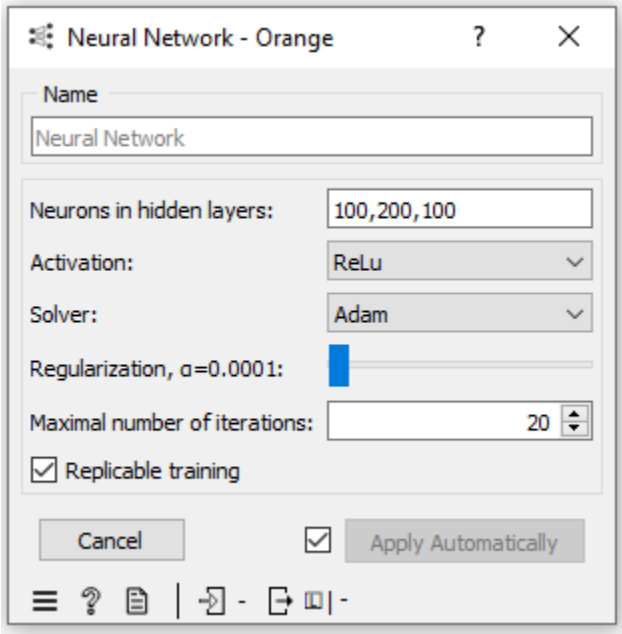
Neurons in hidden layers: 100,200,100

Activation : ReLu

Solver : Adam

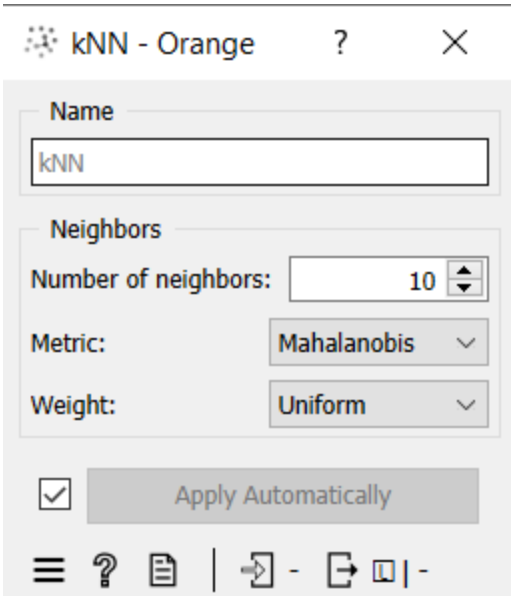
Regularization: 0.0001

Maximal number of iterations: 1000

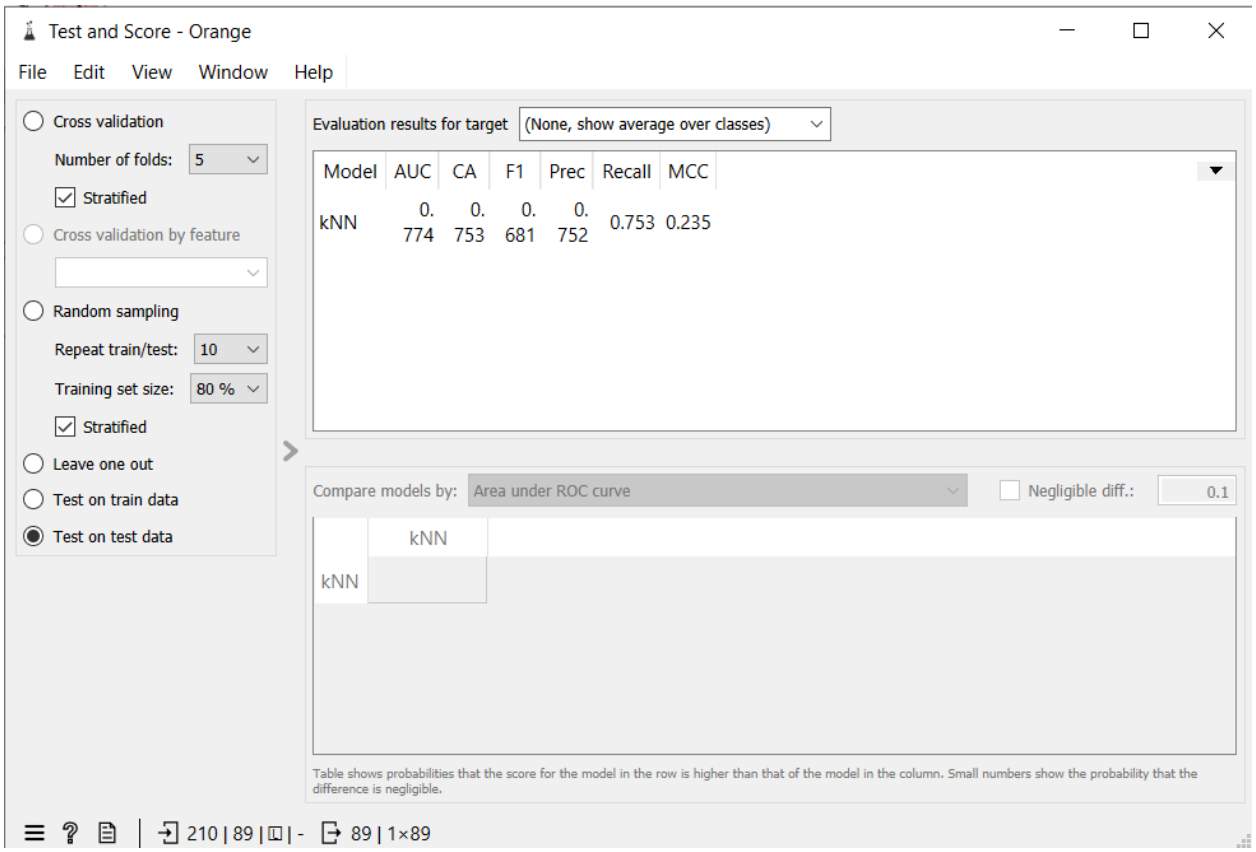


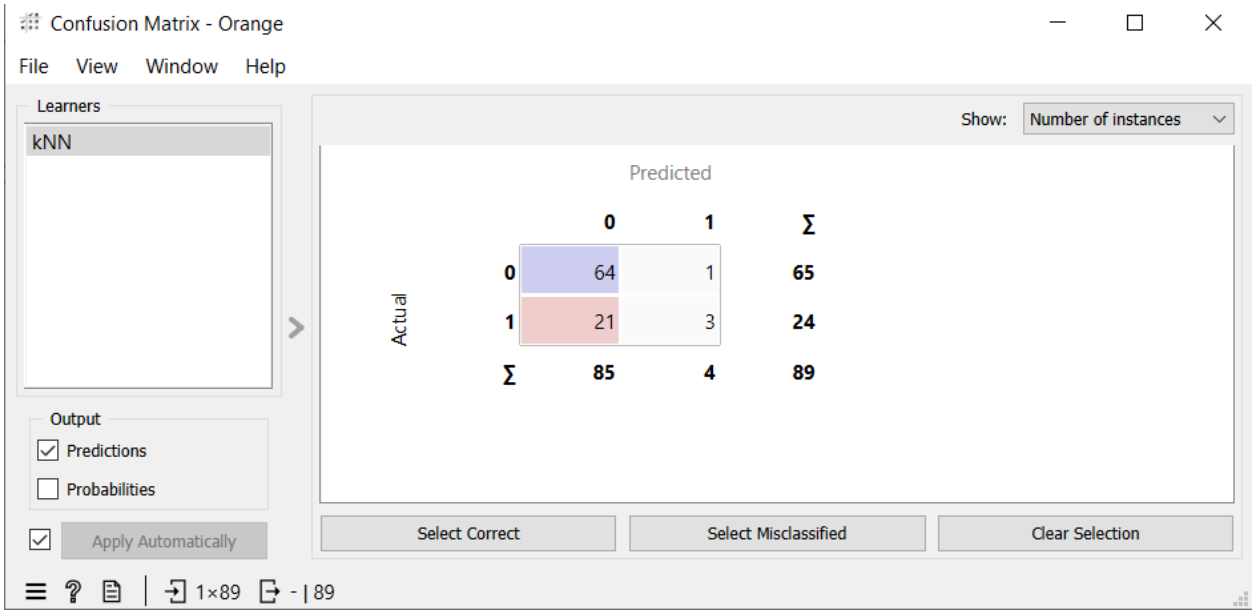
Eksperimenti ar KNN

Eksperiments	Hiperparametru vērtības
1.eksperiments	Number of neighbors = 10 Metric = Mahalanobis Weight = Uniform
2.eksperiments	Number of neighbors = 5 Metric = Mahalanobis Weight = Uniform
3.eksperiments	Number of neighbors = 15 Metric = Mahalanobis Weight = Uniform

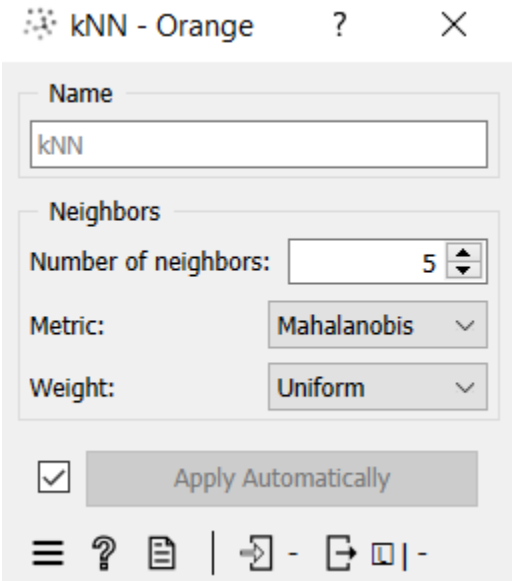


ekrānuzņēmums 1.eksperimenta hiperparametru vērtībām

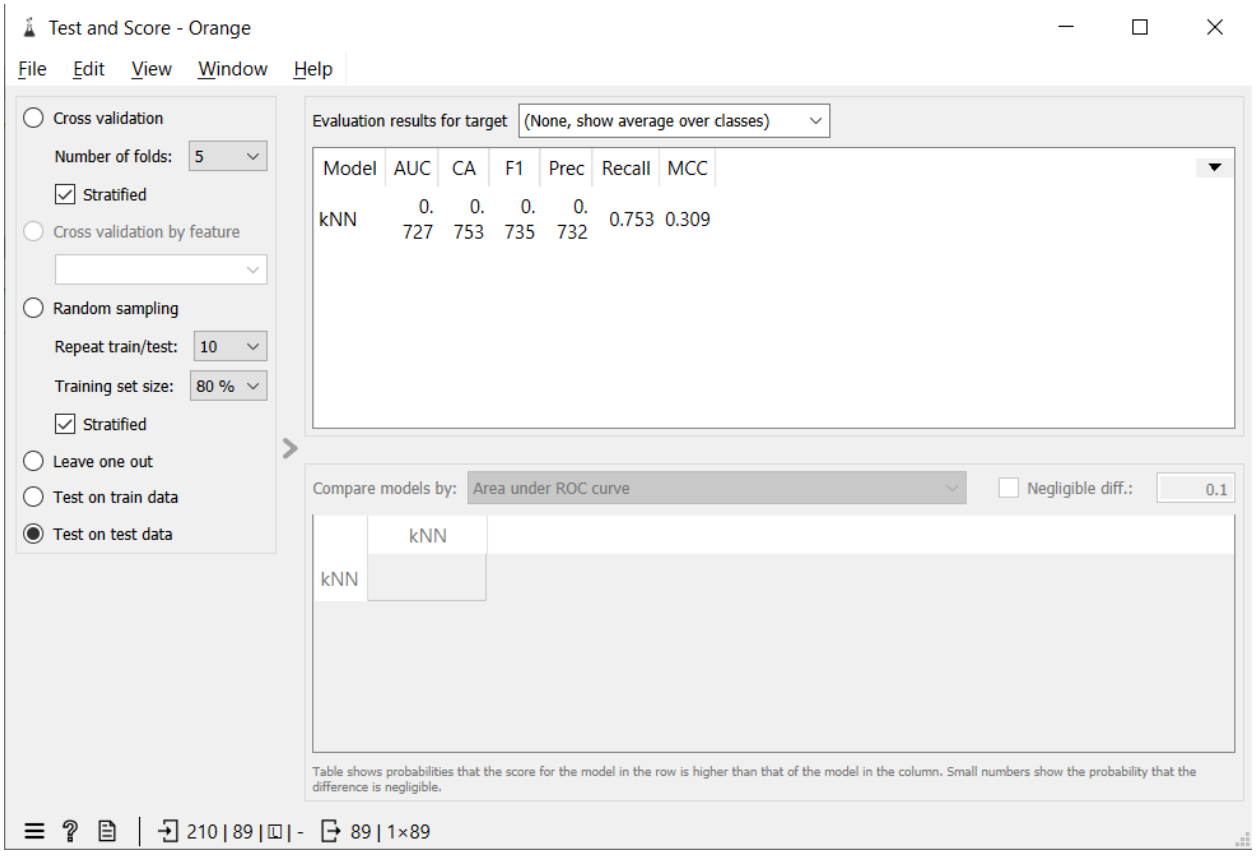


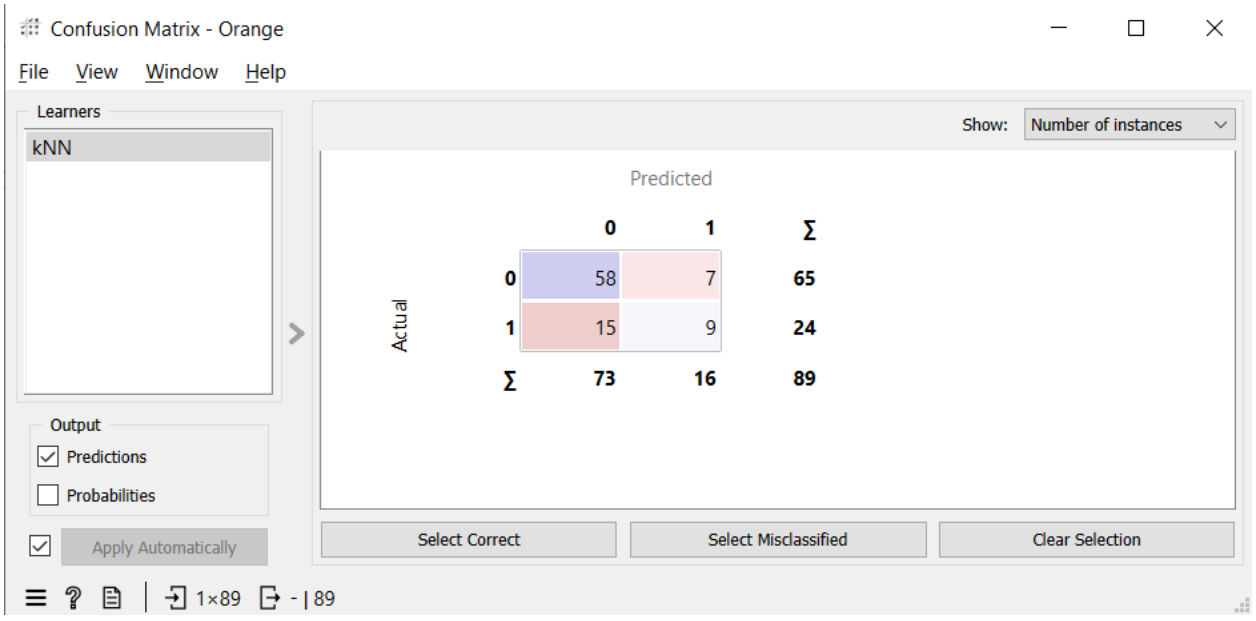


ekrānuzņēmums 1.esperimenta veiktspējas metrikām

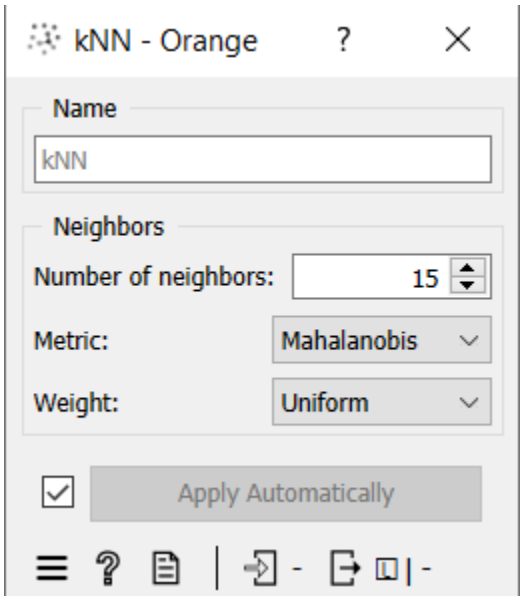


ekrānuzņēmums 2.eksperimenta hiperparametru vērtībām

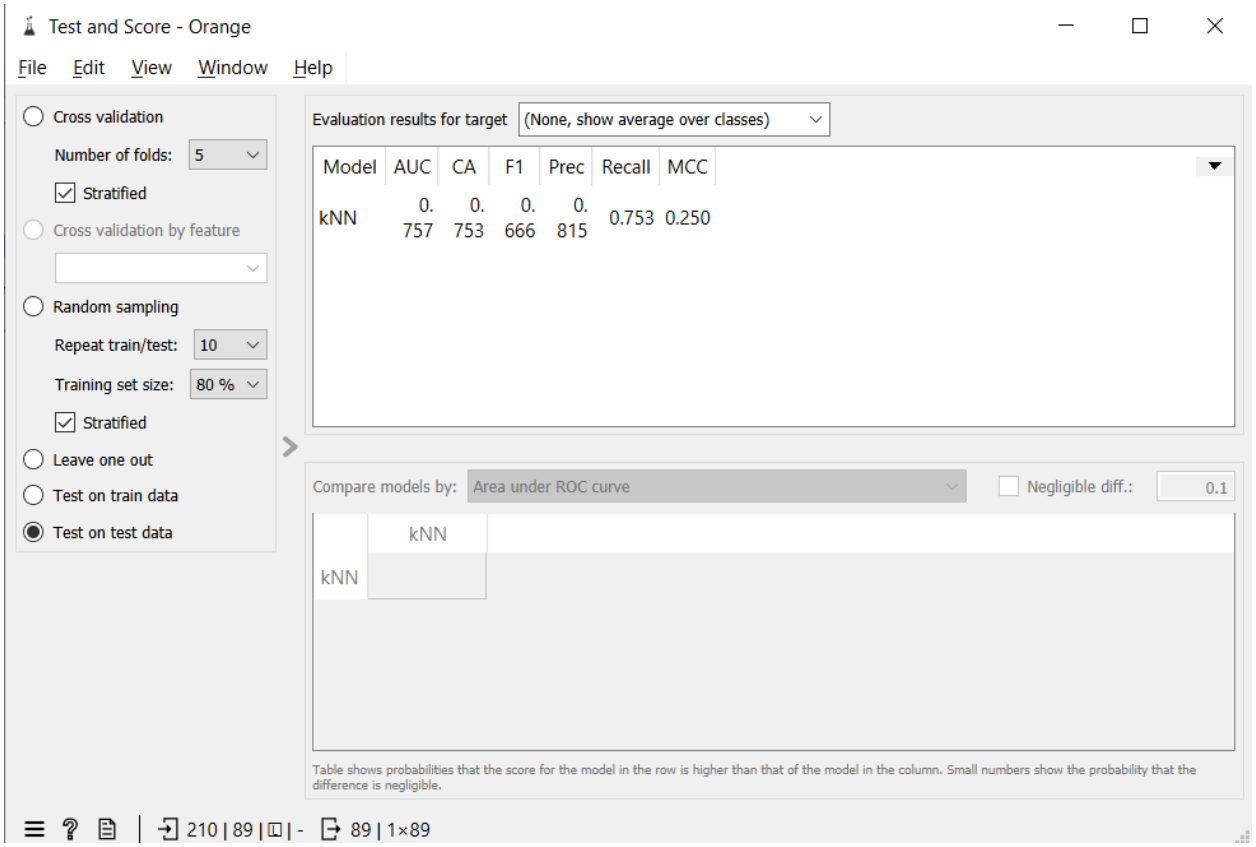


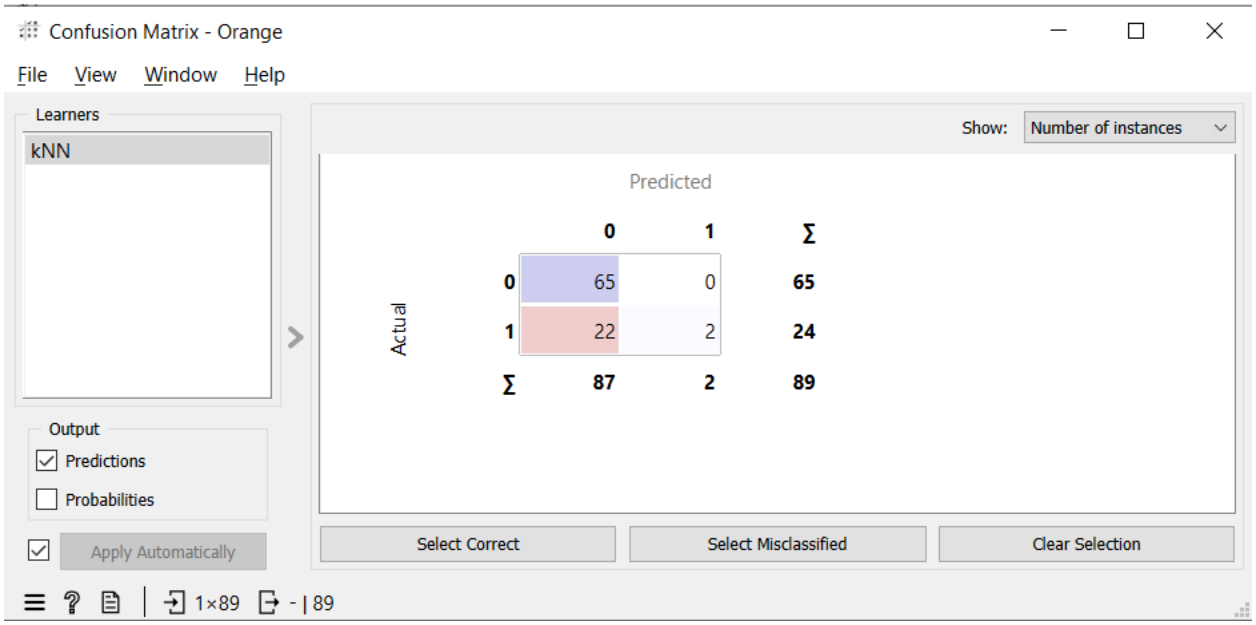


ekrānuzņēmums 2.esperimenta veiktspējas metrikām



ekrānuzņēmums 3.eksperimenta hiperparametru vērtībām



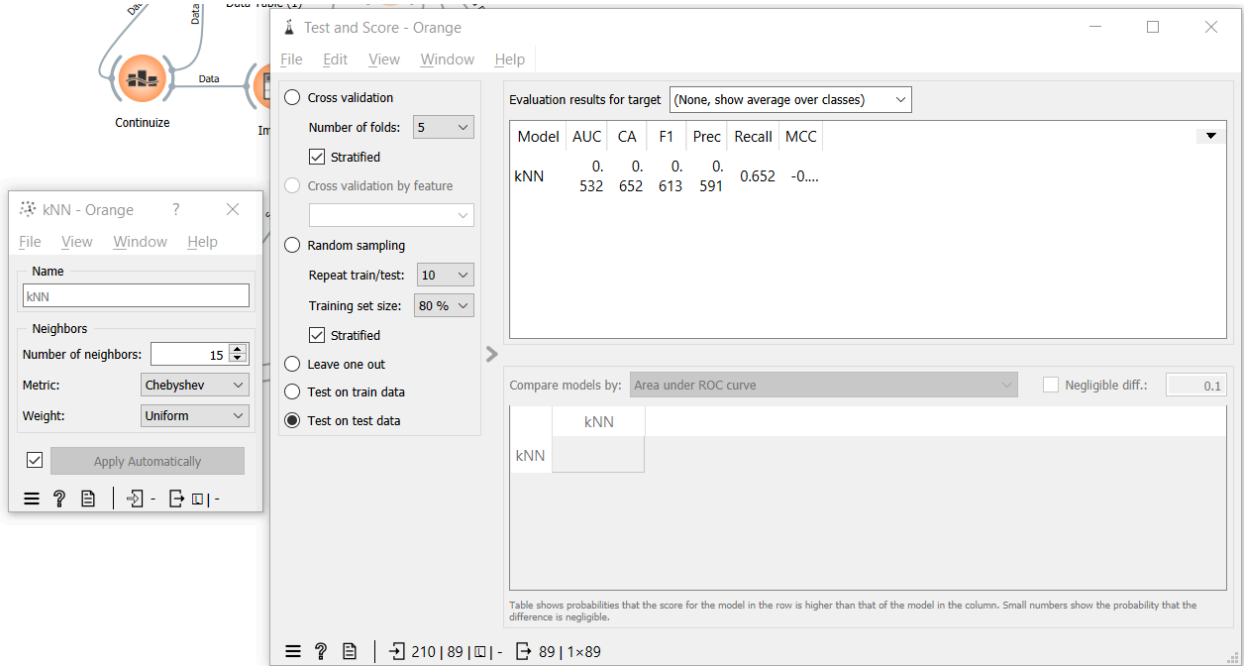


ekrānuzņēmums 3.esperimenta veiktspējas metrikām

Secinājumi no eksperimentiem:

Rezultata ar šo algoritmu, mes varam redzēt, ka precizitāte ļoti vertigā kaimiņšu skaits un mērinašanas sistema Mahalanobis ir vislabakais variants no visiem ar kaimiņu skaitu no 14, jo ar viņu rezultāts ir visprecizākais, bet ar Chebushev sistemu precizitāte ļoti maza un ta ir sliktākais no visiem iespējamajiem, ka arī kaimiņu skaits mazaks par 13 dot mazaku precizitāte.

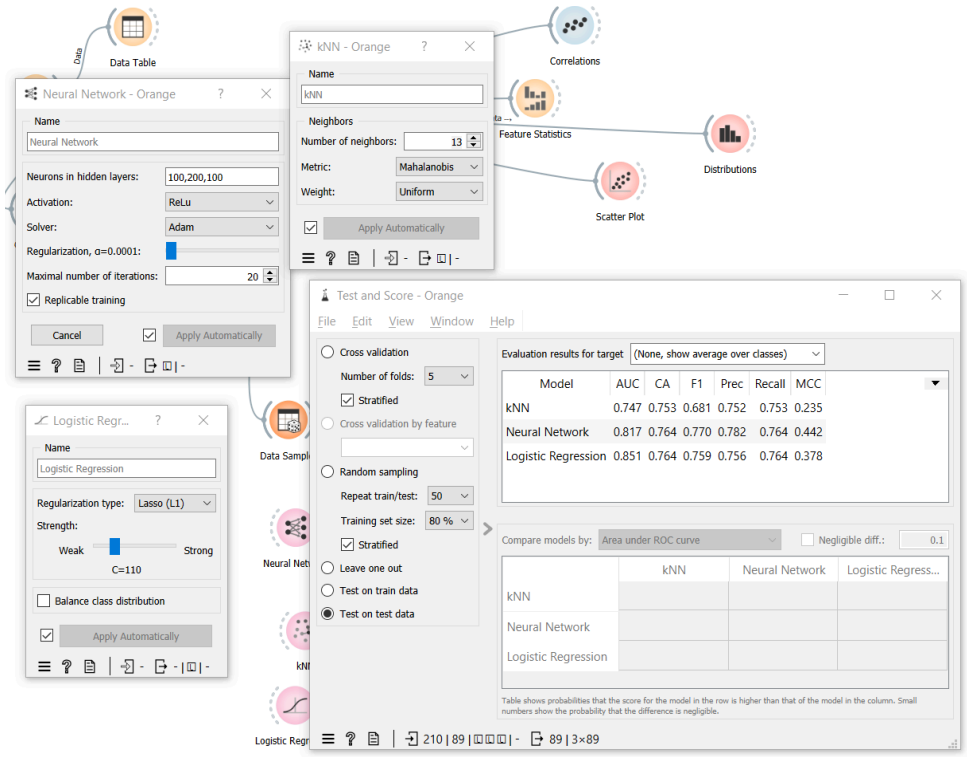
Eksperimenta rezultātā izrādījās, ka visprecīzākā mērīšanas sistēma mūsu situacija ir Mahalanobis, jo ar citiem mērišānas sistēmas “Precision” parameters ir māzaks, ka arī vismazaka precizitāte ar Chebushev sistemu. 3 eksperimenta mes varam redzet, kada “Precision” ar Mahalanobis un salidzinat to ar zemak noradito Chebushev sistemu attelu.

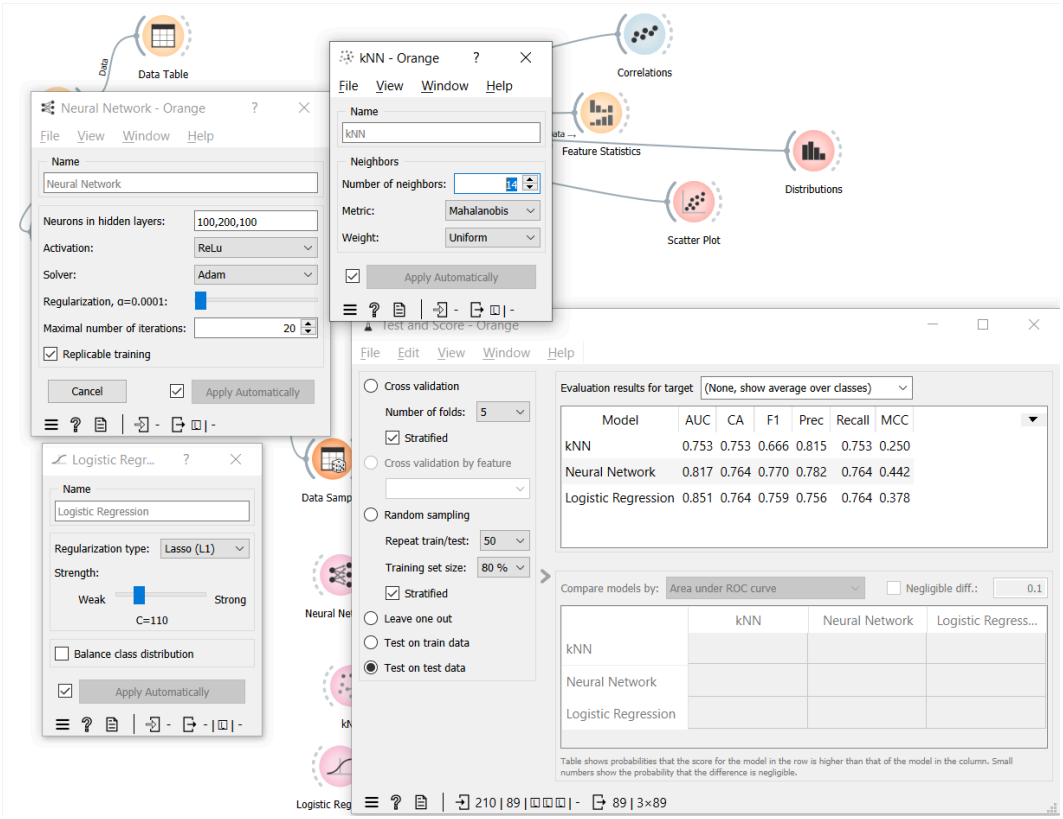


Bet, jā mes izmantosim tikai 5 kaimiņu , tad prezicitāte būs vismazaka no visiem metodiem.

Mes varam secināt, ka kaimiņšu skaits ļoti lielā mērā ietekmē Knn algoritma precizitātes koeficientu.Tad vislabakais variants būs, ja kaimiņu skaits ir lielāks vai vienāds (>=) 14 un mērīšanās sistēmā būs - Mahalanobis un svara koeficients būs - Uniform.

Ja kaimiņu skaits ir mazaks vai vienāds par 13, precizitāte nav visaugstāka no visiem iespejamiem rezultatiem, to mēs varam redzēt no attelem, kas atrodas zemāk.


Data Table
Correlations
Feature Statistics
Distributions
Scatter Plot
Neural Network - Orange
Name
Neural Network
Neurons in hidden layers: 100,200,100
Activation: ReLu
Solver: Adam
Regularization, α=0.0001:
Maximal number of iterations: 20
Replicable training
Cancel
Apply Automatically
kNN - Orange
Name
kNN
Neighbors
Number of neighbors: 13
Metric: Mahalanobis
Weight: Uniform
Apply Automatically
Logistic Regr...
Name
Logistic Regression
Regularization type: Lasso (L1)
Strength:
Weak
Strong
C=110
Balance class distribution
Apply Automatically
Test and Score - Orange
File Edit View Window Help
Cross validation
Number of folds: 5
Stratified
Cross validation by feature
Random sampling
Repeat train/test: 50
Training set size: 80 %
Stratified
Leave one out
Test on train data
Test on test data
Evaluation results for target (None, show average over classes)
Model AUC CA F1 Prec Recall MCC
kNN 0.747 0.753 0.681 0.752 0.753 0.235
Neural Network 0.817 0.764 0.770 0.782 0.764 0.442
Logistic Regression 0.851 0.764 0.759 0.756 0.764 0.378
Compare models by: Area under ROC curve
Negligible diff.: 0.1
kNN
Neural Network
Logistic Regress...
Logistic Regression
Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.
210 | 89 | 89 | 3×89

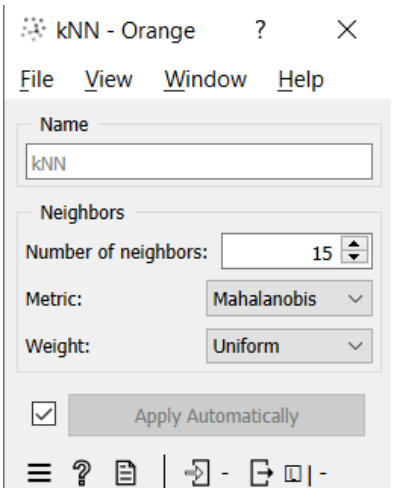

Data Table
Correlations
Feature Statistics
Distributions
Scatter Plot
Neural Network - Orange
File View Window Help
Name
Neural Network
Neurons in hidden layers: 100,200,100
Activation: ReLu
Solver: Adam
Regularization, α=0.0001:
Maximal number of iterations: 20
Replicable training
Cancel
Apply Automatically
kNN - Orange
File View Window Help
Name
kNN
Neighbors
Number of neighbors:
Metric: Mahalanobis
Weight: Uniform
Apply Automatically
Logistic Regr...
Name
Logistic Regression
Regularization type: Lasso (L1)
Strength:
Weak
Strong
C=110
Balance class distribution
Apply Automatically
Test and Score - Orange
File Edit View Window Help
Cross validation
Number of folds: 5
Stratified
Cross validation by feature
Random sampling
Repeat train/test: 50
Training set size: 80 %
Stratified
Leave one out
Test on train data
Test on test data
Evaluation results for target (None, show average over classes)
Model AUC CA F1 Prec Recall MCC
kNN 0.753 0.753 0.666 0.815 0.753 0.250
Neural Network 0.817 0.764 0.770 0.782 0.764 0.442
Logistic Regression 0.851 0.764 0.759 0.756 0.764 0.378
Compare models by: Area under ROC curve
Negligible diff.: 0.1
kNN
Neural Network
Logistic Regress...
Logistic Regression
Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.
210 | 89 | 89 | 3×89

Testēšanai izvēlētais modelis:

Number of neighbors = 15

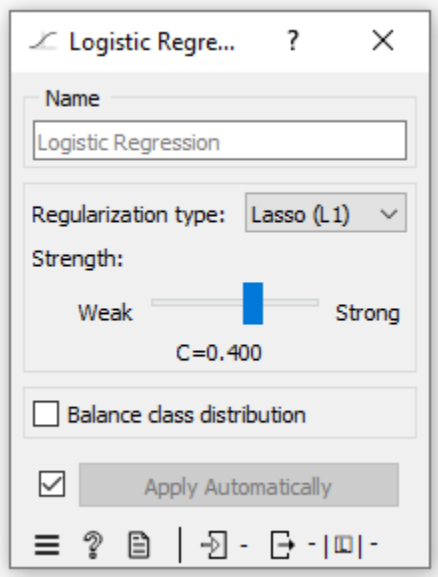
Metric = Mahalanobis

Weight = Uniform

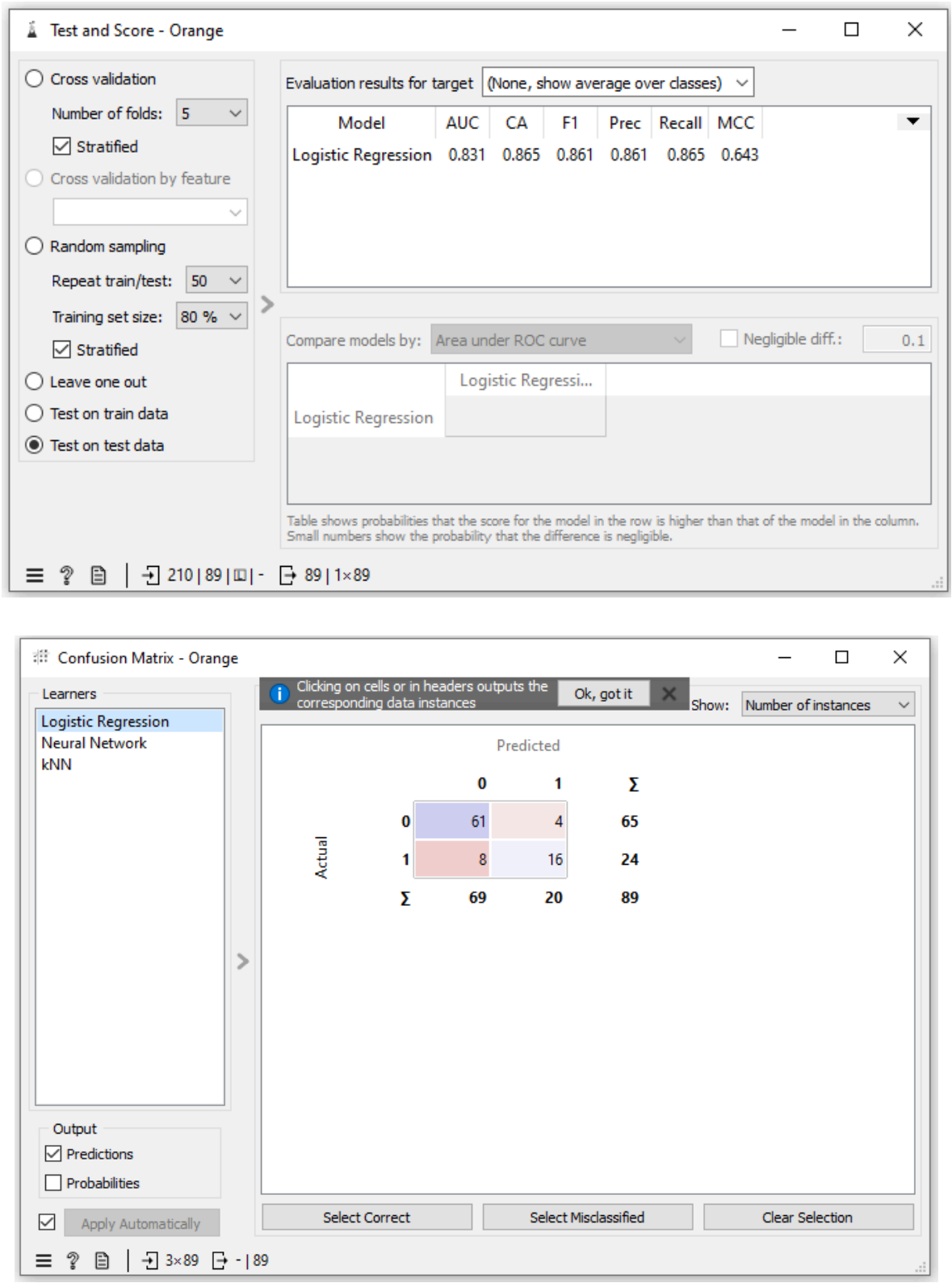


Eksperimenti ar loģistiskā regresija

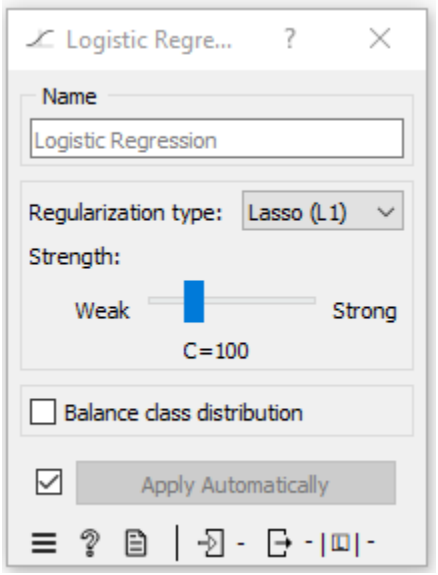
Eksperiments	Hiperparametru vērtības
1.eksperiments	Strength, C = 0.400
2.eksperiments	Strength, C = 100
3.eksperiments	Strength, C = 0.01



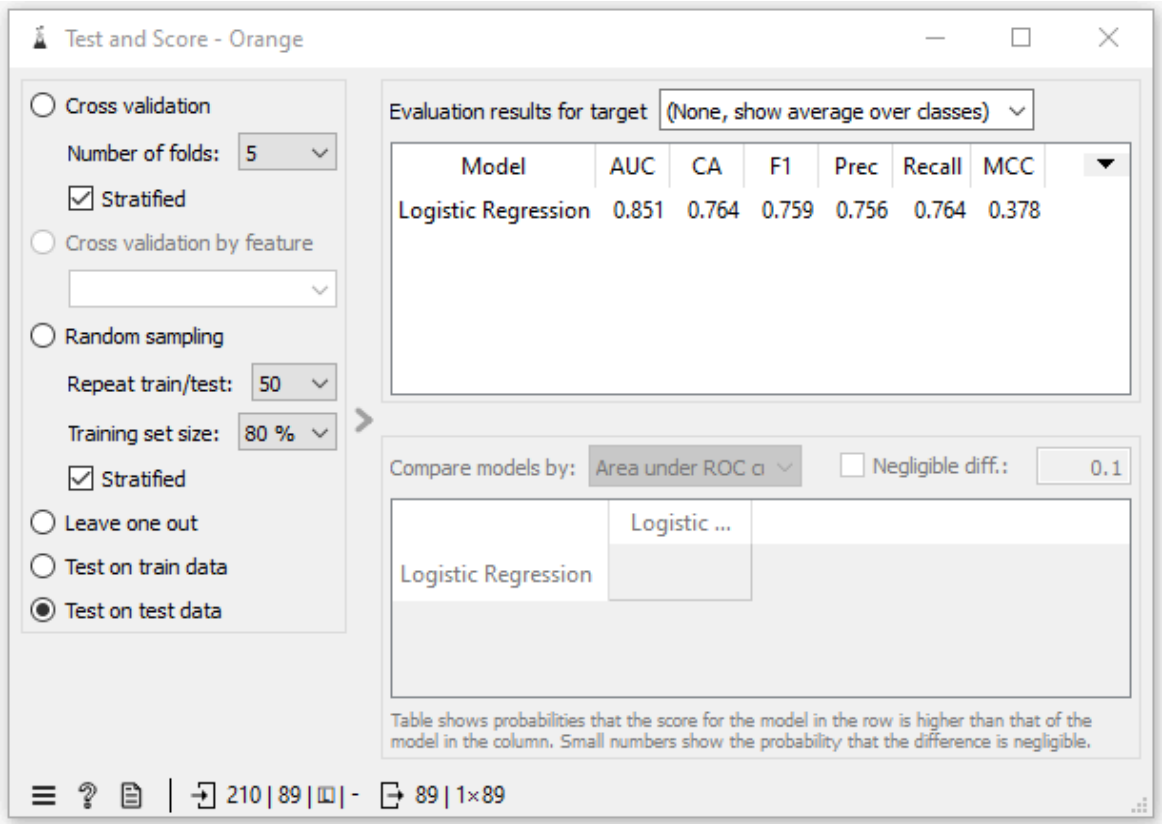
ekrānuzņēmums 1.eksperimenta hiperparametru vērtībām

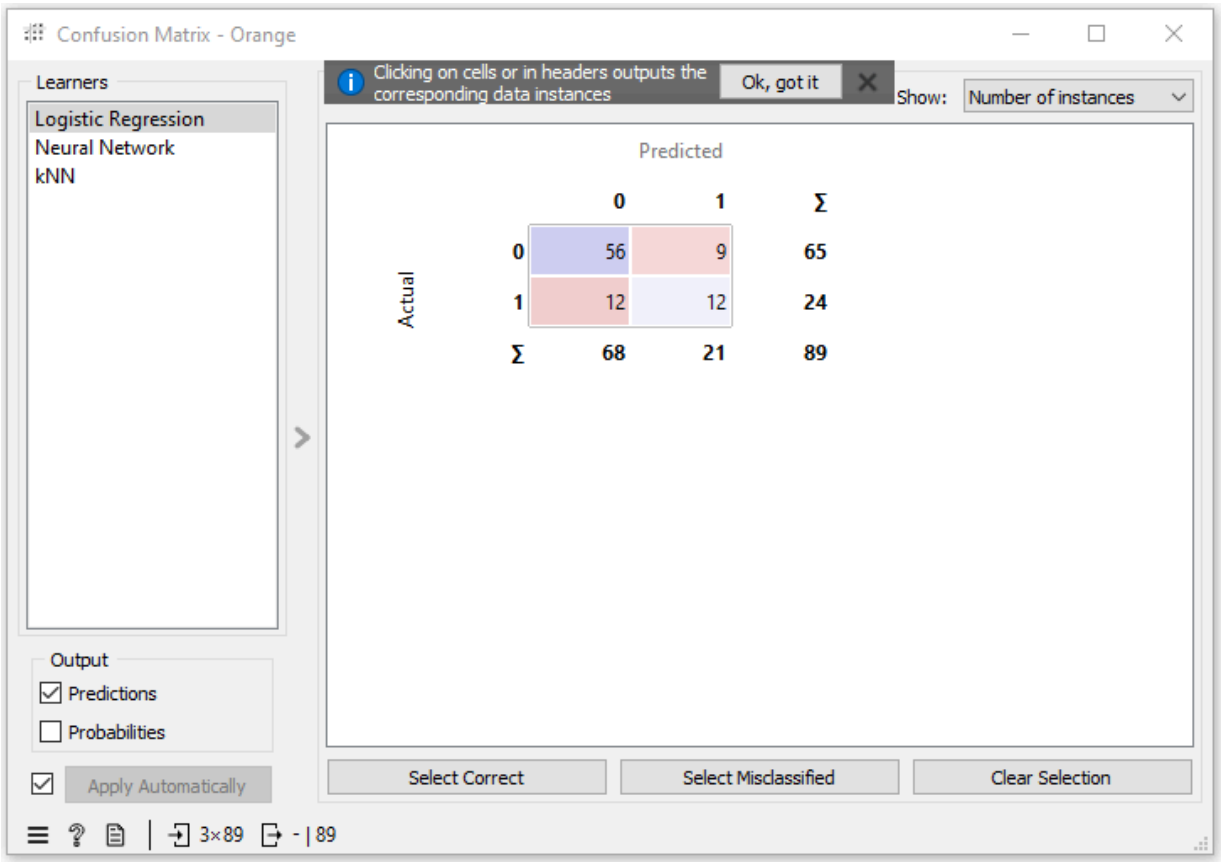


ekrānuzņēmumi 1.esperimenta veiktspējas metrikām

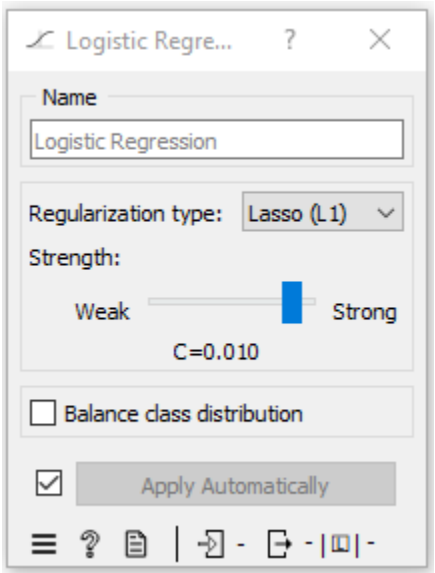


ekrānuzņēmums 2.eksperimenta hiperparametru vērtībām

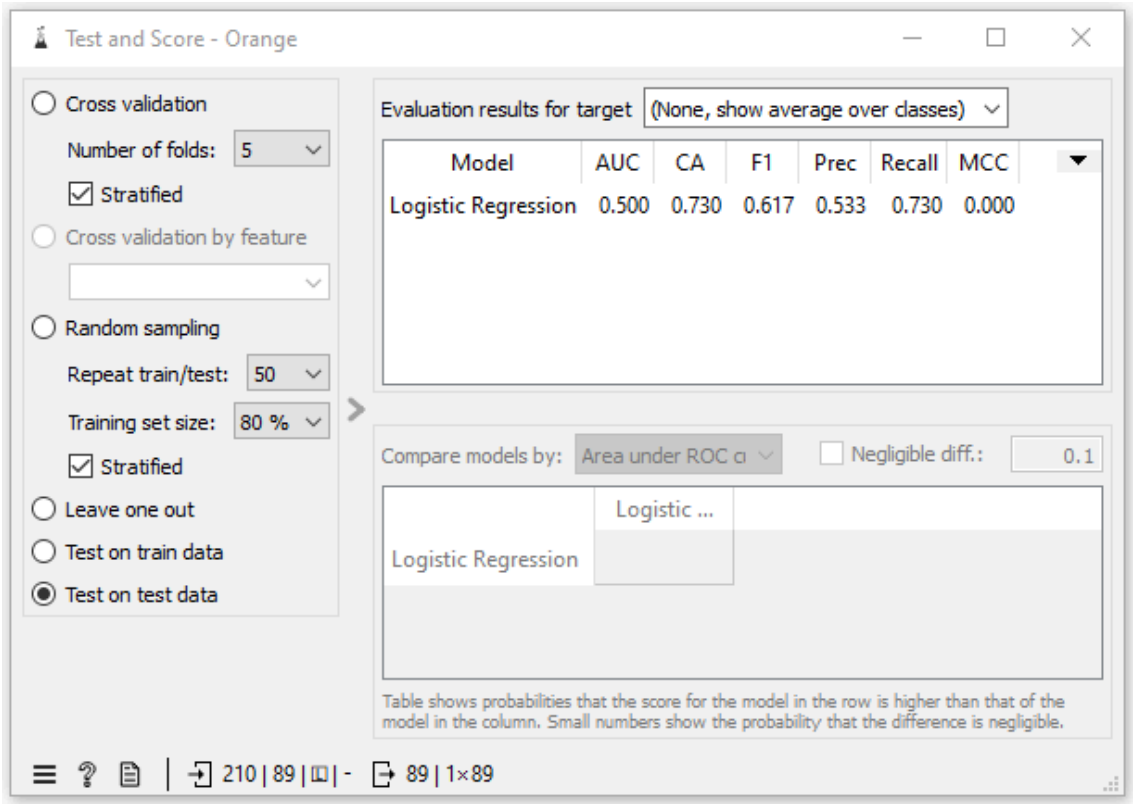


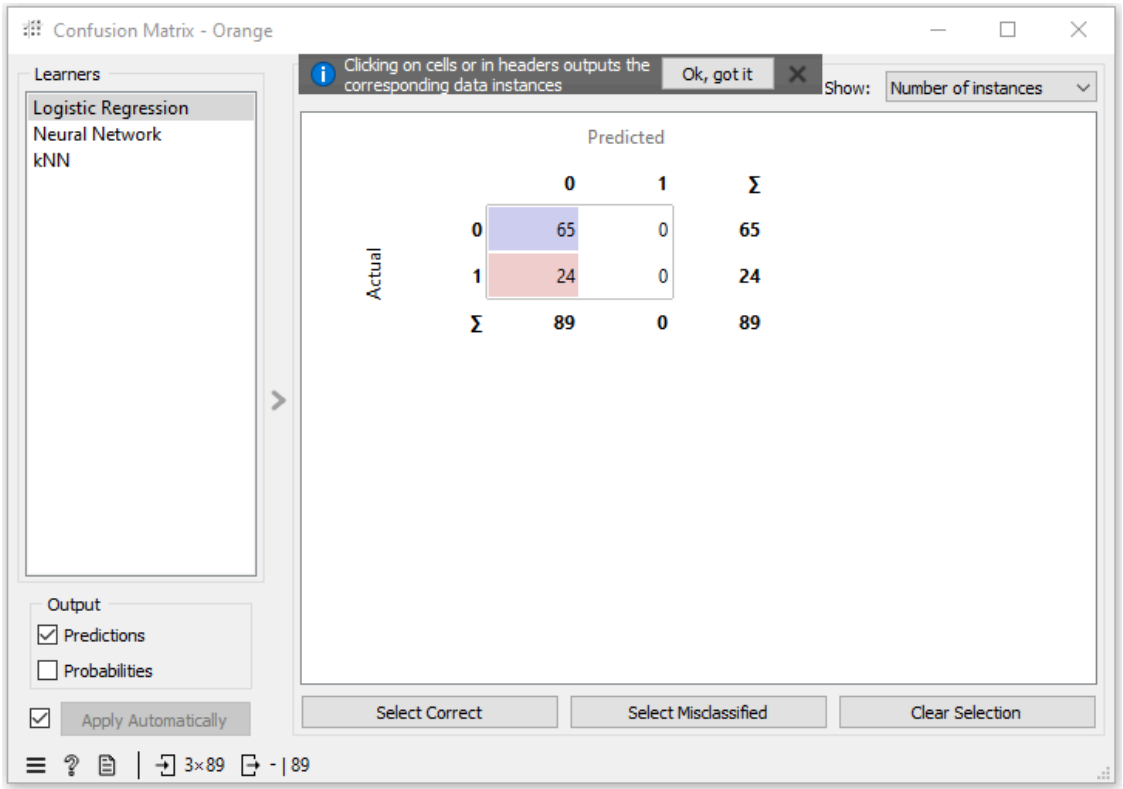


ekrānuzņēmums 2.esperimenta veiktspējas metrikām



ekrānuzņēmums 3.eksperimenta hiperparametru vērtībām





ekrānuzņēmums 3.esperimenta veiktspējas metrikām

Secinājumi no eksperimentiem:

Izdarot secinājumus no šī eksperimenta, varam teikt, ka šim algoritmam galvenais, kas jādara, ir izvēlēties pareizo algoritma stiprumu. Apskatot visus 3 eksperimentus, var saprast, ka tieši 1 eksperimentā(ekrānuzņēmums 1.eksperimenta hiperparametru vērtībām) tika iegūts labākais rezultāts, tieši tāpēc mūsu gadījumā zelta vidusmēra vērtība būs C = 0.400, pie kuras precizitāte bija maksimālā, proti, 0.861(ekrānuzņēmums 1.esperimenta veiktspējas metrikām).

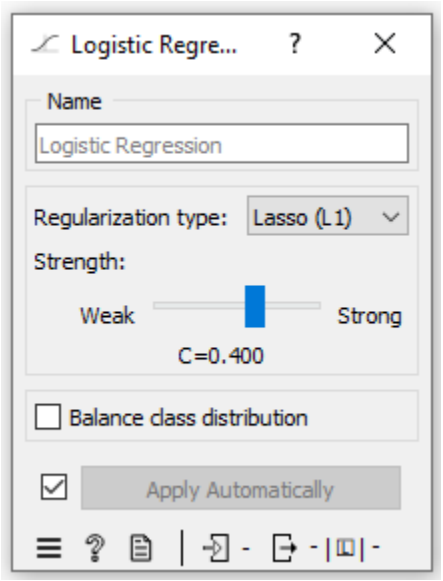
Kopumā var pamanīt, ka algoritms daudz ātrāk zaudē precizitāti, ja spēks pārsniedz noteiktu vērtību, pēc tam tā precizitāte strauji pazeminās (redzams ekrānuzņēmuma 2.esperimenta veiktspējas metrikām). Tāpēc labāk izvēlēties spēku no mazākās puses, kur precizitāte būs lielāka nekā pēc kritiena ar pārāk spēcīgu spēku(redzams ekrānuzņēmuma 3.esperimenta veiktspējas metrikām).

.

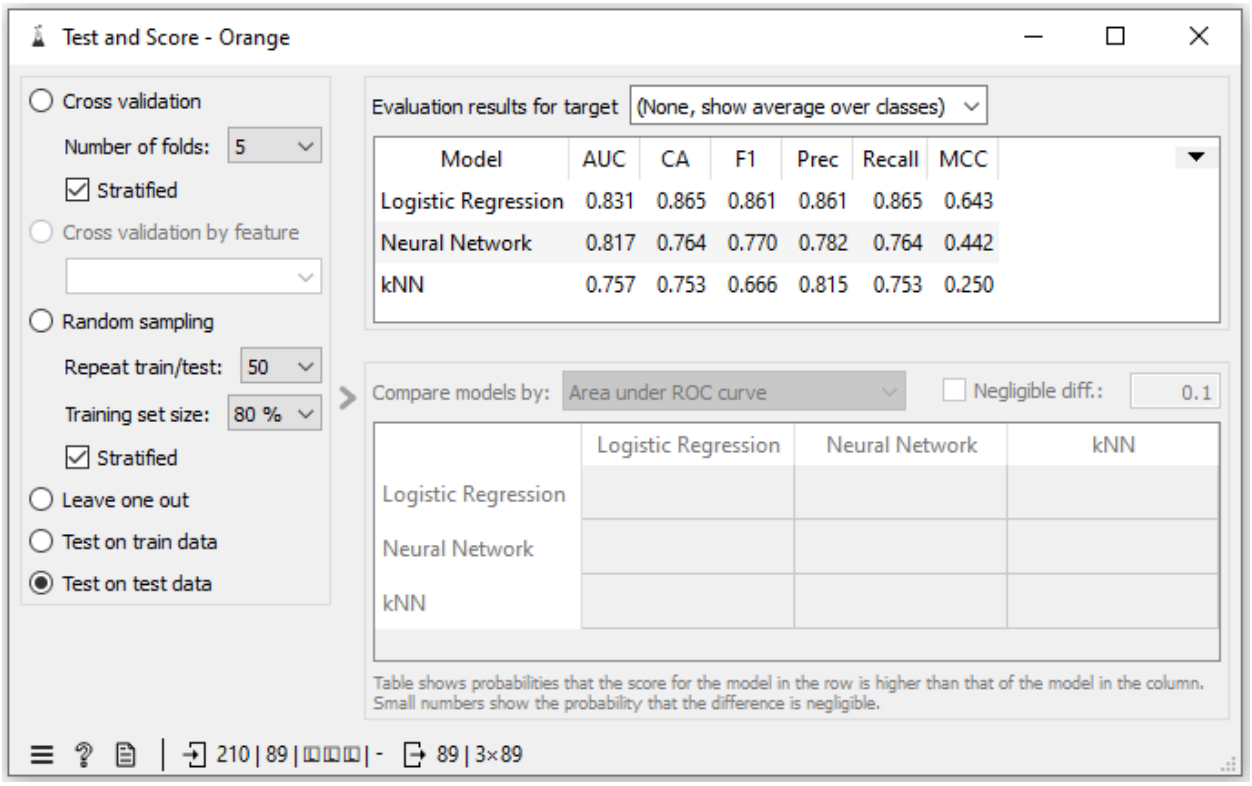
Testēšanai izvēlētais modelis:

Regularization type: Lasso (L1)

Strength: 0.400



Apmācīto modeļu testēšanas rezultāti



ekrānuzņēmums testēšanai izvēlēto modeļu veiktspējas metrikām

Secinājumi pēc testēšanas:

Salīdzinot mūsu algoritmu labākās versijas, jūs varat redzēt, ka tie visi darbojas ar aptuveni 80% precizitāti, visticamāk, tas ir saistīts ar mūsu datiem, kas nav viendimensionāli, un tajos ir anomālijas, šīs neprecizitātes vai anomālijas lielā mērā ietekmē gan apmācības procesu (kas principā liek algoritmam kļūt mazāk precīzam), gan testēšanas procesu, kur algoritms vienkārši nevar pareizi paredzēt rezultātu.

Loģistikas regresijas algoritms uzdevumu izpildīja vislabāk, uzrādot precizitāti 0,861, kas, ņemot vērā mūsu datus, ir diezgan laba precizitāte. Tūlīt pēc tam nāk kNN algoritms ar precizitāti 0,815, šis algoritms ļoti labi atrod līdzīgus rezultātus, un, tā kā mūsu datos lielākā daļa datu ir diezgan tuvu, šis algoritms lieliski tiek galā ar šo uzdevumu. Un pēdējā vietā ir neironu tīkli ar precizitāti 0,782, kas joprojām ir augsta, taču to var saistīt ar faktu, ka neironu tīkls tika apmācīts uz mūsu datiem, no kuriem lielākā daļa ir viena veida dati, kas to apgrūtina lai tas darbotos ar cita veida datiem, kur pacients miris.

Izdarot secinājumus no šī eksperimenta, es teiktu, ka loģistikas regresijas algoritms šai datubāzei darbojās vislabāk, turklāt tam bija visaugstākā precizitāte, tam bija vismazāk nepareizo nāves prognožu. Lai gan to joprojām bija diezgan daudz, no izvēlētajiem algoritmiem tas vislabāk tika galā ar sniegtajiem iestatījumiem.

Jāatzīmē, ka ir svarīgi atzīmēt, ka daži algoritmi tika parādīti nevis ar labākajiem iestatījumiem, bet gan ar labākajiem iestatījumiem, ko mēs atradām, kas nozīmē, ka, ja jūs tālāk eksperimentējat ar tiem, varat sasniegt vēl labākus rezultātus.

Izmantotie informācijas avoti

1)”Orange Visual Programming documentation - Neural Network” - https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/neuralnetwork.html

2)”Orange Visual Programming documentation - kNN” - https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/knn.html

3)”Orange Visual Programming documentation - Logistic Regression” - https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/logisticregression.html

4) Saite uz github, kur atrodas orange fails un excel fails - https://github.com/ArtursMelmanis/MI_25grupa

5) Saite uz youtube video, “How to download file from UCI…” (velāk tas nebija izmantots) - https://www.youtube.com/watch?v=jfXRDQJG4fw&ab_channel=DiscomensClass

6) Pirmajai praktiska uzdevumā daļai, tika izmantots pasniedzēja piedāvātājs youtube video - https://www.youtube.com/watch?v=bmwH3EcTBEM&ab_channel=AllaAnohina-Naumeca

7) Informacija par K-tuvāko kaimiņu metode video no pasniedzeja - https://www.youtube.com/watch?v=fX3XoxDLzg8

8)Informacija par K-tuvāko kaimiņu metode , lai formulet skaidrojumi -

K-Nearest Neighbor(KNN) Algorithm - GeeksforGeeks

9) Loģistiska regresija video nopasniedzēja -

https://www.youtube.com/watch?v=1NMymBvxaPw

10) Loģistiska regresija informacija , lai formulet skaidrojums -

https://www.sciencedirect.com/topics/computer-science/logistic-regression

11) Informacija par K-vidējo algorimta hiperparametri

https://orangedatamining.com/widget-catalog/unsupervised/kmeans/

12) Informacija par scatter plot algorimta hiperparametri

https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/visualize/scatterplot.html

13) Raksts par EF (ejection fraction), kura dati izmantoti, rakstot izkliedes diagrammas diagrammas izvadi ar labākai vērtībai no siluetu koeficientiem

https://www.webmd.com/heart-disease/heart-failure/features/ejection-fraction