



RĪGAS TEHNISKĀ UNIVERSITĀTE

Datorzinātnes un informācijas tehnoloģijas fakultāte

2. Praktiskais darbs
mācību priekšmetā
“Mākslīga intelekta pamati”

Izstrādāja: Mareks Beinarovičs

St. apl. Nr. 211RDB448

Saite uz visiem failiem: https://github.com/KaraSwag69/2.praktiskais_darbs_AI

2022/2023 māc. gads

Saturs

Darba uzdevums	3
I. daļa - Datu pirmapstrāde/izpēte	4
Atlasītā datu kopa	5
Datu kopa kategorijas	6
Visas izvēlētās datu bāzes kategorijas	6
Datu kopas vizuālais attēlojums un statistiskie rādītāji	10
I daļas atbildes un secinājumi	14
II daļa – Nepārraudzītā mašīnmācīšanās	16
Nepārraudzītās mašīnmācīšanās algoritmi	17
Hierarhiskās klasterizācijas algoritma eksperimenti	19
K-vidējo algoritma Silhouette Score	20
II daļas secinājumi	20
III daļa – Pārraudzītā mašīnmācīšanās	21
Pārraudzītās mašīnmācīšanās algoritmi	22
Eksperimenti	24
Apmācītu modeļu salīdzinājums pēc algoritmu veikspējas	27
Secinājumi	28
Avoti	29

Darba uzdevums

Šī darba izpildei studentiem ir nepieciešams izvēlēties datu kopu un izmantot tās apstrādei pārraudzītās un nepārraudzītās mašīnmācīšanās algoritmus. Darba mērķis ir attīstīt studentu prasmes izmantot mašīnmācīšanās algoritmus un analizēt iegūtos rezultātus. Šī darba galarezultāts ir studenta sagatavotā atskaite par darba izpildi.

Darba izstrādei studentiem ir ieteicams izmantot Orange rīks. Tā lietotāja pamācība ir pieejama e-studiju kursa sadala “Praktiskie darbi”. Darba izpildes kontekstā īpaši vērtīgi ir šādi Orange logrīki: File, Data table, Data Sampler, Bar Plot, Scatter plot, Feature Statistics, Distributions, Test and Score, Predictions, Confusion matrix, Silhouette plot, Roc analysis, kā arī dažādu mašīnmācīšanās algoritmu logrīki. Tajā pašā laikā students var izvēlēties izpildīt darbu Python valodā. Tomēr tālākais uzdevuma apraksts pamatā attiecas uz rīku Orange, bet tās pašas prasības tiek piemērotas, ja students izmanto Python valodu.

Ir jāņem vērā, ka darba izpildes nolūkam studentiem, iespējams, būs nepieciešams patstāvīgi meklēt un pētīt papildu informācijas avotus, lai atbildētu uz šī darba jautājumiem vai sniegtu iegūto rezultātu analīzi un interpretāciju. Lai atrastu datu kopu darba izpildei, studenti var izmantot šādas plaši zināmās krātuves:

- UC Irvine Machine Learning Repository <https://archive.ics.uci.edu/ml/index.php>
- R Datasets on Github <https://vincentarelbundock.github.io/Rdatasets/>
- Kaggle Datasets <https://www.kaggle.com/datasets>
- Awesome Lists: Public Datasets <https://github.com/caesar0301/awesome-public-datasets>
- Yahoo! Webscope Datasets <https://webscope.sandbox.yahoo.com/?guccounter=1>
- Reddit: <https://www.reddit.com/r/datasets>

Izvēloties datu kopu, studentiem ir jāņem vērā šādi aspekti:

- ir jāizvēlas datu kopa, kas ir piemērota klasifikācijas uzdevumam. Students nedrīkst izvēlēties Iris ziedu (Iris data set) vai Pingvīnu (Palmer Archipelago (Antarctica) penguin data) datu kopas. Turklāt ir jāpiedomā pie klasifikācijas jēgpilnuma, piemēram, klasificēt kontinentus pēc Covid-19 gadījumiem ir bezjēdzīgi, jo, pirmkārt, ir tikai 6 kontinenti un jaunie drīz vai tuvākajā laikā parādīsies un, otrkārt, Covid-19 gadījumu skaits nav kontinentu raksturojošā īpašība;
- ir vēlams izvēlēties datu kopu, kas jau ir dota .csv datu faila formātā;
- datu kopai ir jābūt labi dokumentētai (ir jābūt pieejamai informācija par datu kopas izveidotāju, laiku, kad tā tika izveidota, un datu avotu);
- datu kopai ir jābūt saprātīga izmēra (vismaz 200 datu objekti);
- datu kopai ir jābūt detalizētam aprakstam par datu kopā esošajām datu pazīmēm (atribūtiem) un to nozīmi;
- datu pazīmju (atribūtu) skaitam ir jābūt diapazonā no 5 līdz 15;
- datu kopai ir jāsaturs klašu iezīmes;
- studentiem ir jāizvairās no datu kopām, kurās ir daudz Būla tipa (patiess/nepatiess, 1/0 utt.) vai kategoriskā tipa pazīmju (atribūtu) vērtību. Ir vēlams izmantot datu kopas, kurās lielākā daļa no pazīmēm ir atspoguļota ar nepārtrauktām pazīmju vērtībām;
- studentiem ir jāizvairās no datu kopām, kurās klašu iezīmes nav dotas (piemēram, teksta korpusiem un neapstrādātiem attēliem).

I. daļa - Datu pirmapstrāde/izpēte

Lai izpildītu šī darba daļu, studentiem ir jāveic šādas darbības:

1. Ir jāizvēlas un jāapraksta datu kopa, pamatojoties uz informāciju, kas sniegta krātuvē, kurā datu kopa ir pieejama.
2. Ja no krātuves iegūtā datu kopa nav formātā, ar kuru ir viegli strādāt (piemēram, komatatzīmētās vērtības vai .csv fails), ir jāveic tās transformācija vajadzīgajā formātā.
3. Ja kādu pazīmju (atribūtu) vērtības ir tekstveida vērtības (piemēram, yes/no, positive/neutral/negative, u.c.), tās ir jātransformē skaitliskās vērtībās.
4. Ja kādiem datu objektiem trūkst atsevišķu pazīmju (atribūtu) vērtības, ir jāatrod veids, kā tās iegūt, studējot papildu informācijas avotus.
5. Ir jāatspoguļo datu kopa vizuāli un jāaprēķina statistiskie rādītāji:
 - ir jāizveido vismaz divas 2- vai 3-dimensiju izkliedes diagrammas (scatter plot), kas ilustrē klases atdalāmību, balstoties uz dažādām pazīmēm (atribūtiem); studentam ir jāizvairās izmantot datu objekta ID vai klases iezīmi kā mainīgo izkliedes diagrammā;
 - ir jāizveido vismaz 2 histogrammas, kas parāda klašu atdalīšanu, pamatojoties uz interesējošām pazīmēm (atribūtiem);
 - ir jāatspoguļo 2 interesējošo pazīmju (atribūtu) sadalījums;
 - ir jāaprēķina statistiskie rādītāji (vismaz vidējās vērtības un dispersiju).

Atlasītā datu kopa

Nosaukums: Mobile Price Classification

Autors: Abhishek Sharma

Avots: www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification?select=test.csv

Datu kopas autora apraksts:

Oriģinālvaloda: angļu

Bob has started his own mobile company. He wants to give tough fight to big companies like Apple, Samsung etc.

He does not know how to estimate price of mobiles his company creates. In this competitive mobile phone market you cannot simply assume things. To solve this problem he collects sales data of mobile phones of various companies.

Bob wants to find out some relation between features of a mobile phone(eg:- RAM, Internal Memory etc) and its selling price. But he is not so good at Machine Learning. So he needs your help to solve this problem.

In this problem you do not have to predict actual price but a price range indicating how high the price is.

Tulkojums latviešu valodā

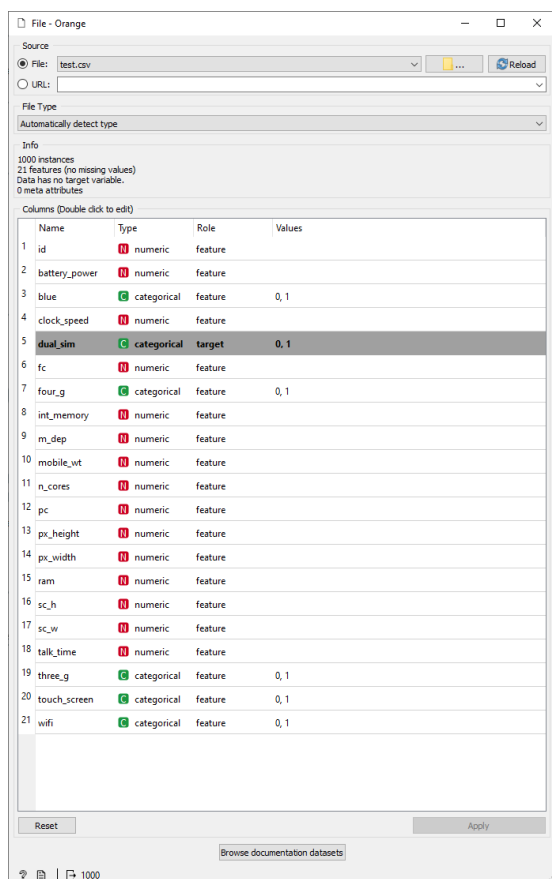
Bobs ir izveidojis savu mobilo uzņēmumu. Viņš vēlas, lai dotu grūts cīņa lieliem uzņēmumiem, piemēram, Apple, Samsung utt.

Viņš nezina, kā novērtēt mobilo sakaru cenu, ko rada viņa uzņēmums. Šajā konkurētspējīgajā mobilo tālrunu tirgū jūs nevarat vienkārši uzņemties lietas. Lai atrisinātu šo problēmu, viņš apkopo dažādu uzņēmumu mobilo tālrunu pārdošanas datus.

Bobs vēlas uzzināt kādu saistību starp mobilā tālruņa funkcijām (piemēram: RAM, iekšējā atmiņa utt.), bet viņš nav tik labs Mašīnmācībā. Tāpēc viņam ir nepieciešama jūsu palīdzība, lai atrisinātu šo problēmu.

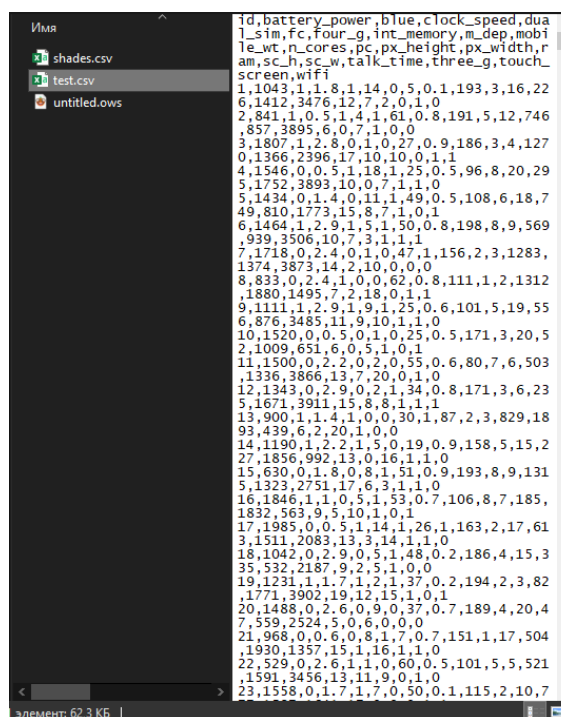
Šajā problēmā jums nav jāparedz faktiskā cena, bet cenu diapazons, kas norāda, cik augsta ir cena.

Datu kopa kategorijas



Name	Type	Role	Values
id	numeric	feature	
battery_power	numeric	feature	
blue	categorical	feature	0, 1
clock_speed	numeric	feature	
dual_sim	categorical	target	0, 1
fc	numeric	feature	
four_g	categorical	feature	0, 1
int_memory	numeric	feature	
m_dep	numeric	feature	
mobile_wt	numeric	feature	
n_cores	numeric	feature	
pc	numeric	feature	
px_height	numeric	feature	
px_width	numeric	feature	
ram	numeric	feature	
sc_h	numeric	feature	
sc_w	numeric	feature	
talk_time	numeric	feature	
three_g	categorical	feature	0, 1
touch_screen	categorical	feature	0, 1
wifi	categorical	feature	0, 1

1.att Kategorijas



```
id,battery_power,blue,clock_speed,dual_sim,fc,four_g,int_memory,m_dep,mobile_wt,n_cores,pc,px_height,px_width,ram,sc_h,sc_w,talk_time,three_g,touch_screen,wifi
1,1043,1,1,8,1,14,0,5,0,1,193,3,16,22
6,1412,3476,12,7,2,0,1,0
2,841,1,0,5,1,4,1,61,0,8,191,5,12,746
,857,3895,6,0,7,1,0,0
3,1807,1,2,8,0,1,0,27,0,9,186,3,4,127
0,1366,2996,17,10,10,0,1,1
4,1546,0,0,5,1,18,1,25,0,5,96,8,20,29
5,1752,3893,10,0,7,1,1,0
5,1434,0,1,4,0,11,1,49,0,5,108,6,18,7
49,810,1773,15,8,7,1,0,1
6,1464,1,2,9,1,5,1,50,0,8,198,8,9,569
939,3506,10,7,3,1,1,1
7,1718,0,2,4,0,1,0,47,1,156,2,3,1283,
1374,3873,14,2,10,0,0,0
8,833,0,2,4,1,0,0,62,0,8,111,1,2,1312
,1880,1495,7,2,18,0,1,1
9,1111,1,2,9,1,9,1,25,0,6,101,5,19,55
6,876,3485,11,9,10,1,1,0
10,1520,0,0,5,0,1,0,25,0,5,171,3,20,5
2,1009,651,6,0,5,1,0,1
11,1500,0,2,2,0,2,0,55,0,6,80,7,6,503
,1336,3866,13,7,20,0,1,0
12,1343,0,2,9,0,2,1,34,0,8,171,3,6,23
5,1671,3911,15,8,8,1,1,1
13,900,1,1,4,1,0,0,30,1,87,2,3,829,18
93,439,6,2,20,1,0,0
14,1190,1,2,2,1,5,0,19,0,9,158,5,15,2
27,1856,992,13,0,16,1,1,0
15,630,0,1,8,0,8,1,51,0,9,193,8,9,131
5,1323,2751,17,6,3,1,1,0
16,1846,1,1,0,5,1,53,0,7,106,8,7,185,
1832,563,9,5,10,1,0,1
17,1985,0,0,5,1,14,1,26,1,163,2,17,61
3,1511,2083,13,3,14,1,1,0
18,1042,0,2,9,0,5,1,48,0,2,186,4,15,3
35,532,2187,9,2,5,1,0,0
19,1231,1,1,7,1,2,1,37,0,2,194,2,3,82
,1771,3902,19,12,15,1,0,1
20,1488,0,2,6,0,9,0,37,0,7,189,4,20,4
7,559,2524,5,0,6,0,0,0
21,968,0,0,6,0,8,1,7,0,7,151,1,17,504
,1930,1357,15,1,16,1,1,0
22,529,0,2,6,1,1,0,60,0,5,101,5,5,521
,1591,3456,13,11,9,0,1,0
23,1558,0,1,7,1,7,0,50,0,1,115,2,10,7
```

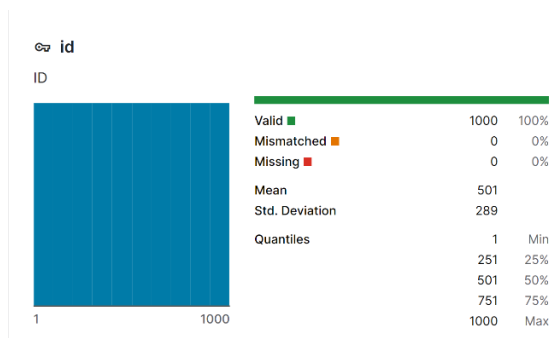
2.att Datu kopas .csv formātā

Manā izvēlētajā datu bāzē sākotnēji bija .CSV formātā.

Teksta vērtības, kas būtu jāmaina uz skaitliskām nebija, kā arī trūka vērtību

Visas izvēlētas datu bāzes kategorijas

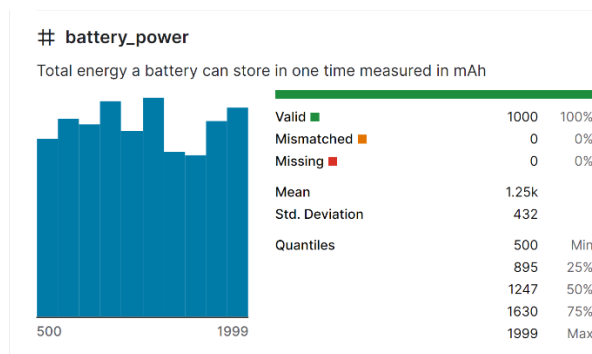
ID



3.att Kategorija – ID

Katra ieraksta numurs

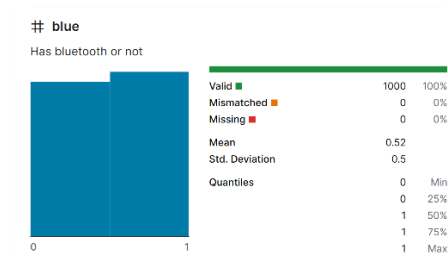
Battery power



4.att Kategorija – Battery power

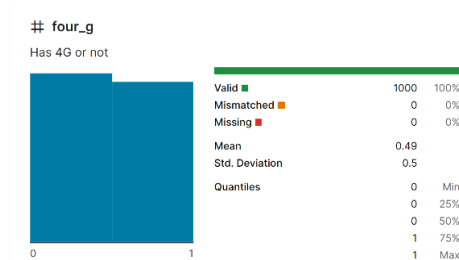
Kopējā enerģija, ko akumulators var uzglabāt vienā reizē, mērot mAh

Bluetooth



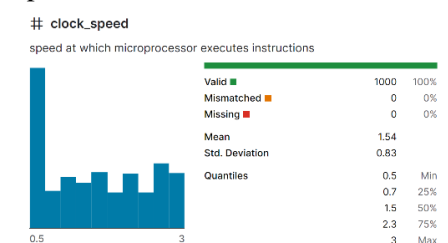
5.att Kategorija – Blue
Bluetooth klātbūtne

4G



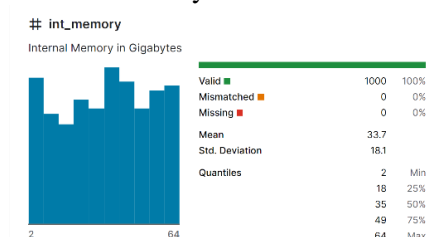
9.att Kategorija – four g
4G atbalsts

Speed



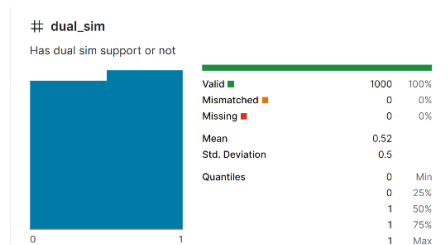
6.att Kategorija – Clock speed
Ātrums, kādā mikroprocesors izpilda instrukcijas

Internal memory



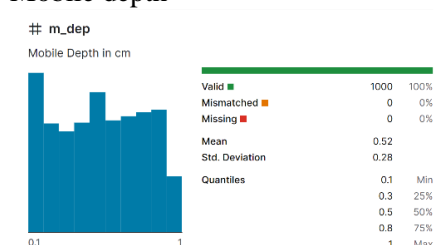
10.att Kategorija – int memory
Iekšējā atmiņa gigabaitos

Dual SIM-card



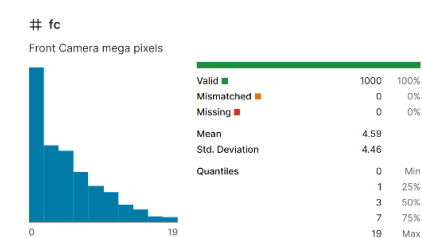
7.att Kategorija – Dual sim
Dual SIM atbalsts

Mobile depth



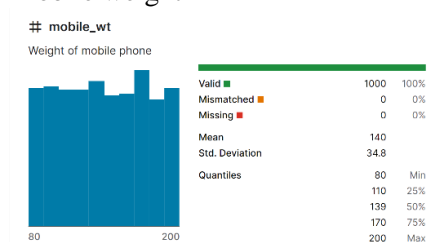
11.att Kategorija – m dep
Mobilais Dziļums cm

Front Camera



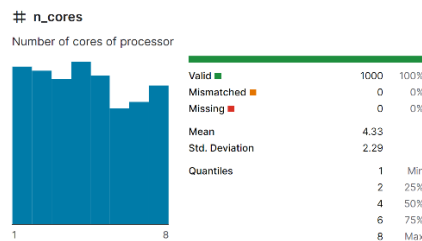
8.att Kategorija – fc
Priekšējās kameras megapikseļi

Mobile weight



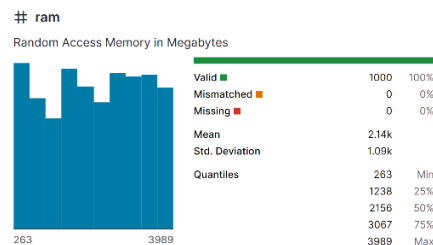
12.att Kategorija – mobile wt
Mobilā tālruņa svars

Number of cores



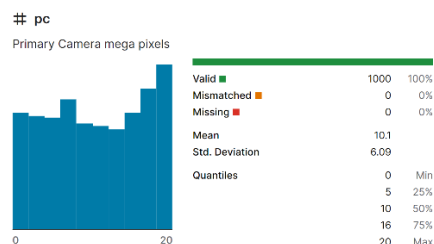
13.att Kategorija – n cores
Procesora kodolu skaits

RAM



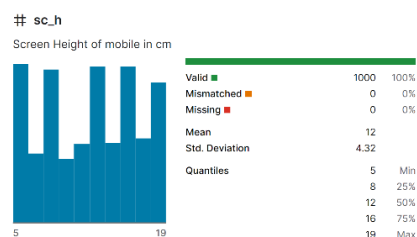
17.att Kategorija – ram
RAM atmiņa megabaitos

Primary camera



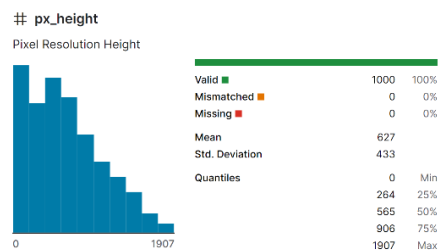
14.att Kategorija – pc
Primārās kameras megapikseļi

Screen height



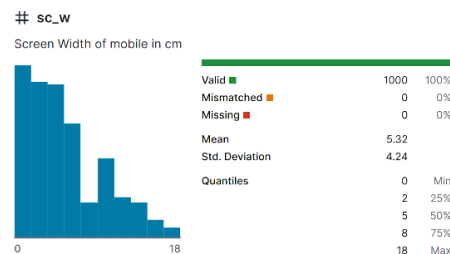
18.att Kategorija – sc h
Ekrāna Augstums cm

Pixel resolution height



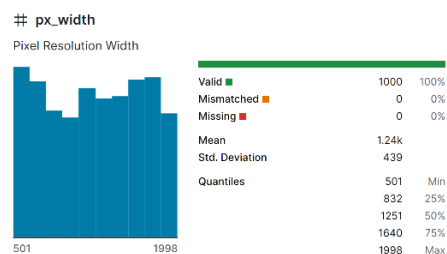
15.att Kategorija – px height
Pikseļu Izšķirtspējas Augstums

Screen width



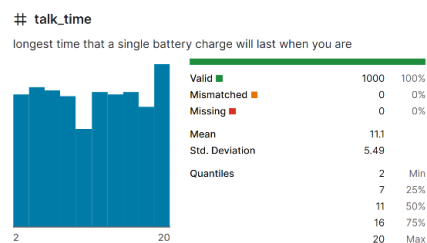
19.att Kategorija – sc w
Ekrāna platums cm

Pixel resolution width



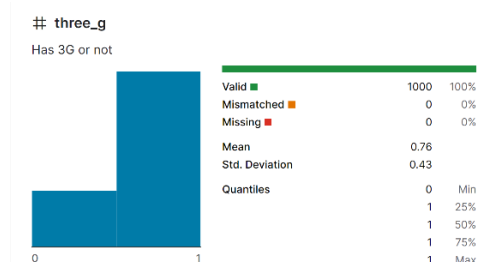
16.att Kategorija – px width
Pikseļu Izšķirtspējas Platums

Time of talk



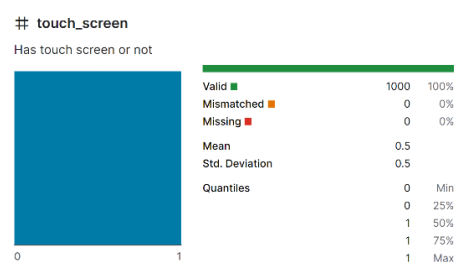
20.att Kategorija – talk time
Maksimālais tālruņa lietošanas laiks sarunā ar pilnu akumulatora uzlādi

3G



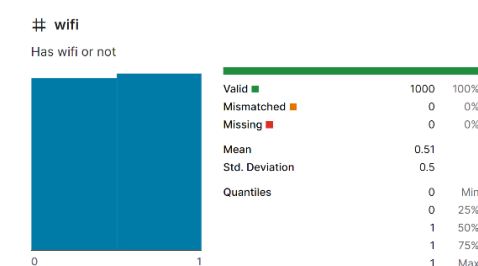
21.att Kategorija – three g
3G atbalsts

Touch screen



22.att Kategorija – touch screen
Skārienekrāns klātbūtne

Wi-Fi



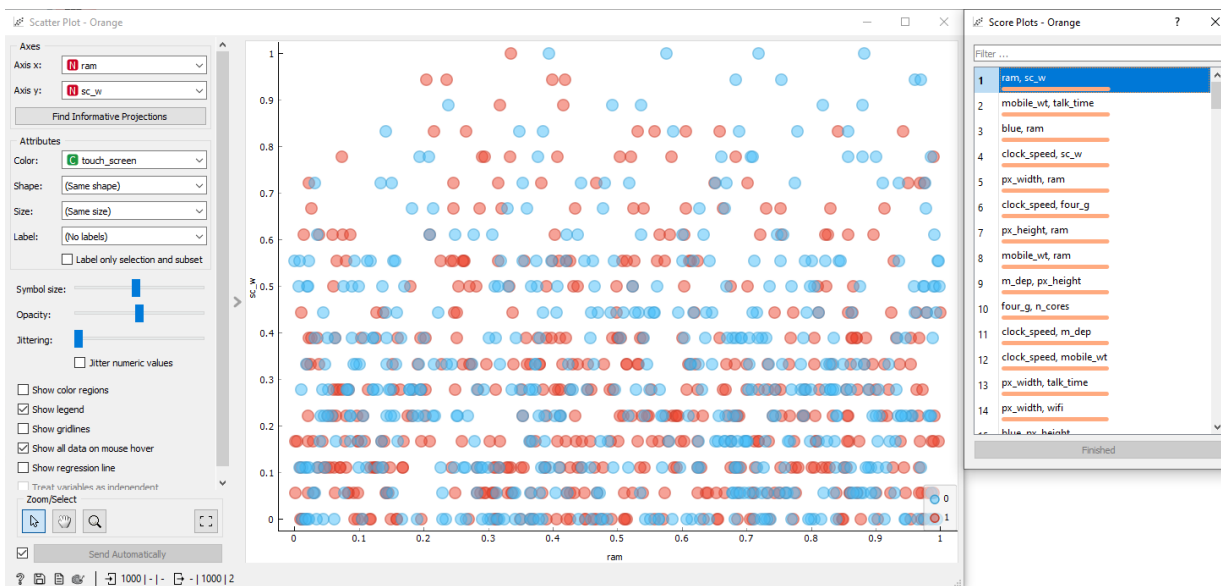
23.att Kategorija – wifi
Wi-Fi atbalsts

Atribūts	Paskaidrojums	Vērtību tips	Diapazons
id	Katra ieraksta numurs	Skaitlis	1-1000
battery_power	Kopējā enerģija, ko akumulators var uzglabāt vienā reizē, mērot mAh	Skaitlis	500-1999
blue	Bluetooth klātbūtne	Skaitlis	0-1
clock_speed	Ātrums, kādā mikroprocesors izpilda instrukcijas	Skaitlis	0.5-3
dual_sim	Dual SIM atbalsts	Skaitlis	0-1
fc	Priekšējās kameras megapikseli	Skaitlis	0-19
four_g	4G atbalsts	Skaitlis	0-1
int_memory	Iekšējā atmiņa gigabaitos	Skaitlis	2-64
m_dep	Mobilais Dziļums cm	Skaitlis	0.1-1
mobile_wt	Mobilā tālruņa svars	Skaitlis	80-200
n_cores	Procesora kodolu skaits	Skaitlis	1-8
pc	Primārās kameras megapikseli	Skaitlis	0-20
px_height	Pikseļu Izšķirtspējas Augstums	Skaitlis	0-1907
px_width	Pikseļu Izšķirtspējas Platums	Skaitlis	501-1998
ram	RAM atmiņa megabaitos	Skaitlis	263-3989
sc_h	Ekrāna Augstums cm	Skaitlis	5-19
sc_w	Ekrāna platums cm	Skaitlis	0-18
talk_time	Maksimālais tālruņa lietošanas laiks sarunā ar pilnu akumulatora uzlādi	Skaitlis	2-20
three_g	3G atbalsts	Skaitlis	0-1
touch_screen	Skārienekrāns klātbūtne	Skaitlis	0-1
wifi	Wi-Fi atbalsts	Skaitlis	0-1

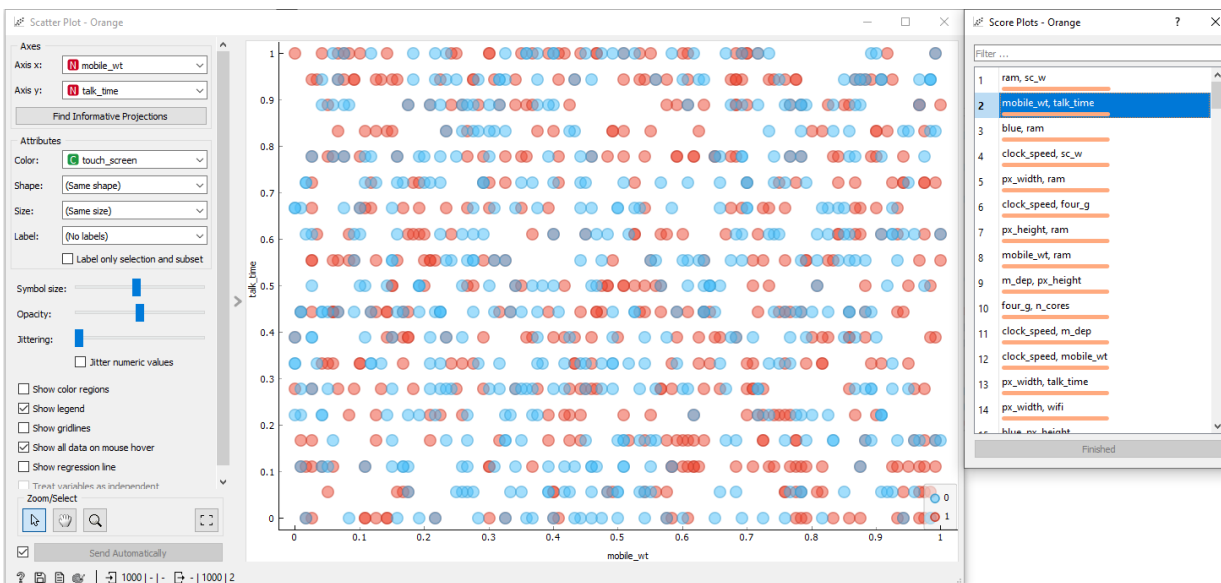
Datu kopas vizuālais attēlojums un statistiskie rādītāji

a) Izkliedes diagrammas

Kā izvēles atribūts tika izvēlēts skārienekrāna klātbūtne

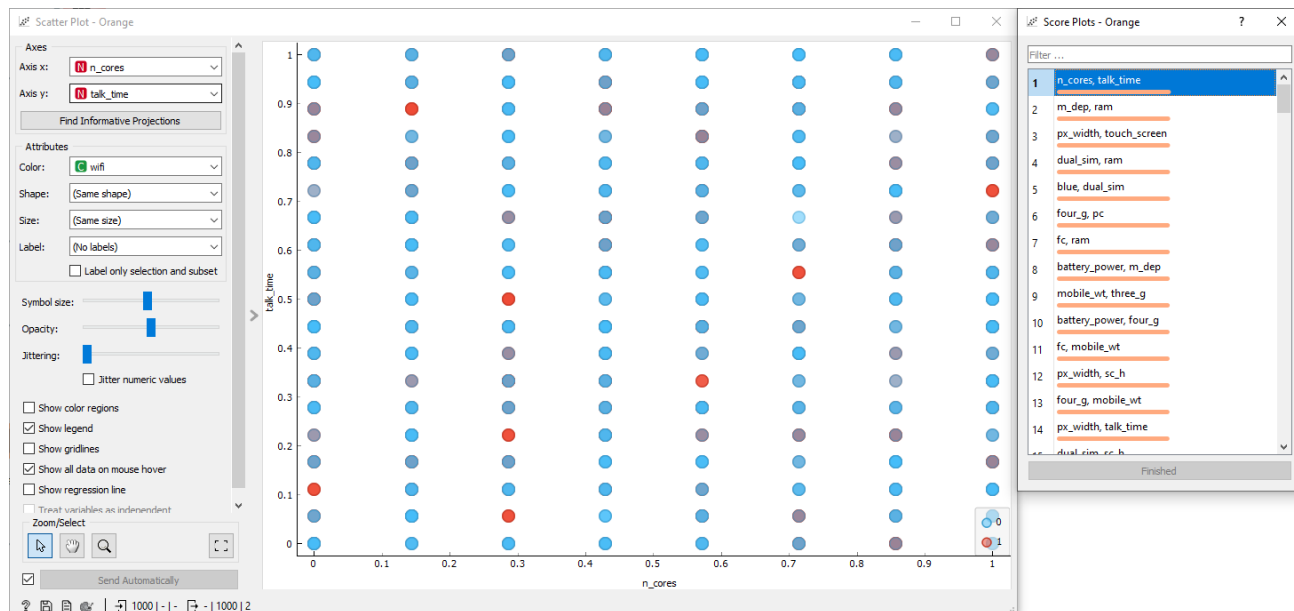


24.att Izkliedes diagramma 1

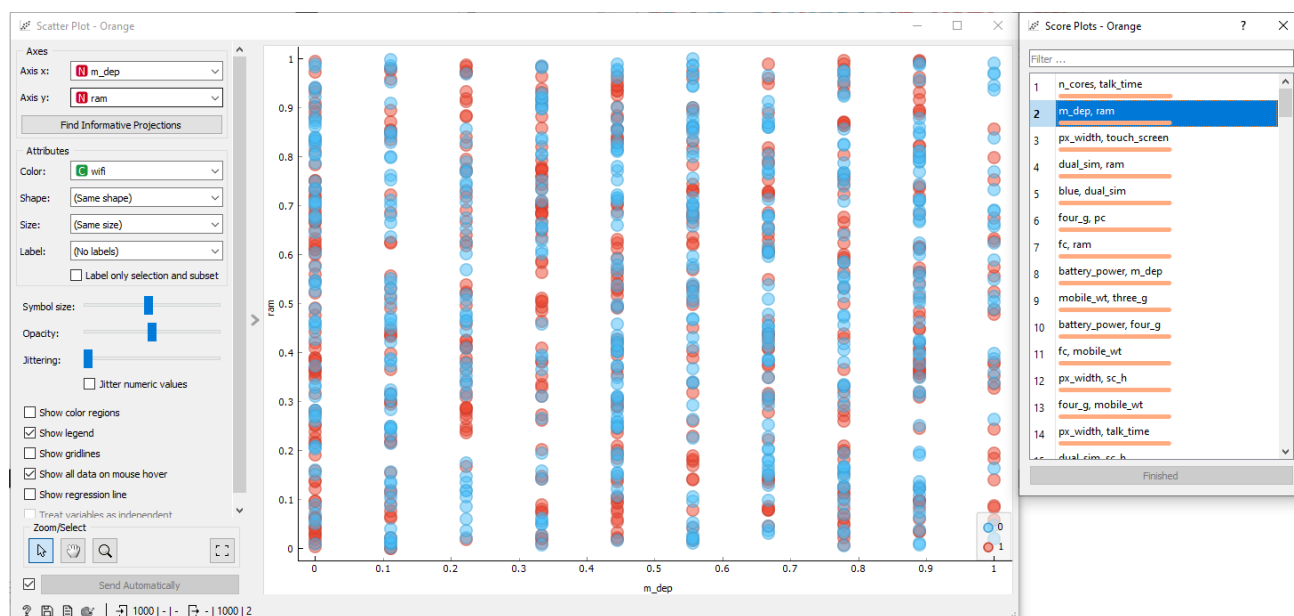


25.att Izkliedes diagramma 2

Kā izvēles atribūts tika izvēlēta Wi-Fi klātbūtne



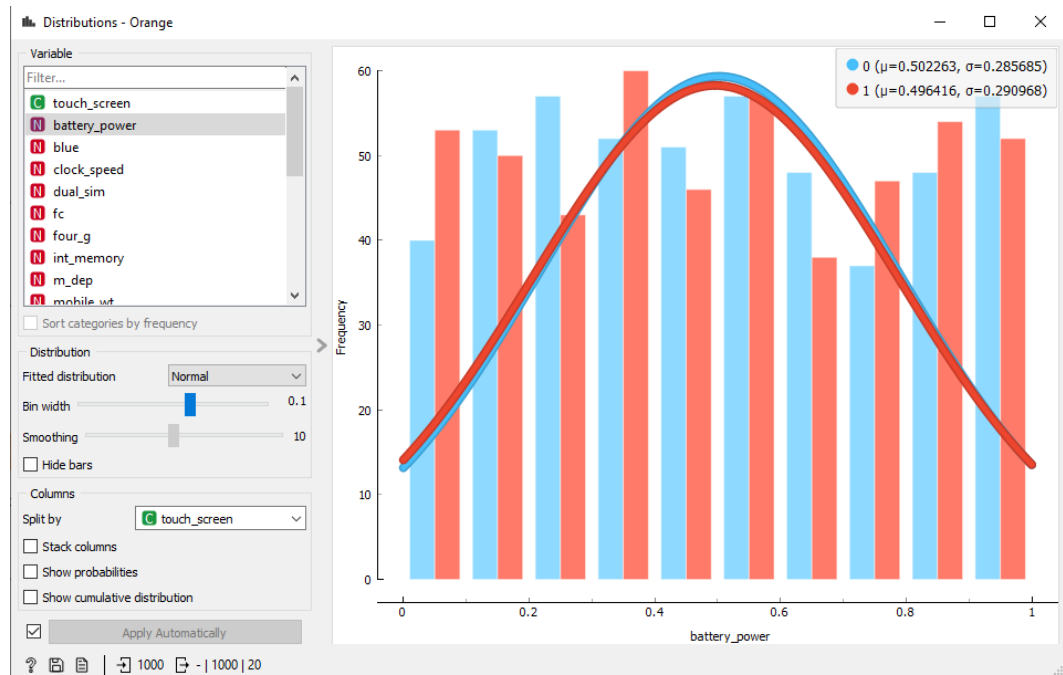
26.att Izklides diagramma 3



26.att Izklides diagramma 4

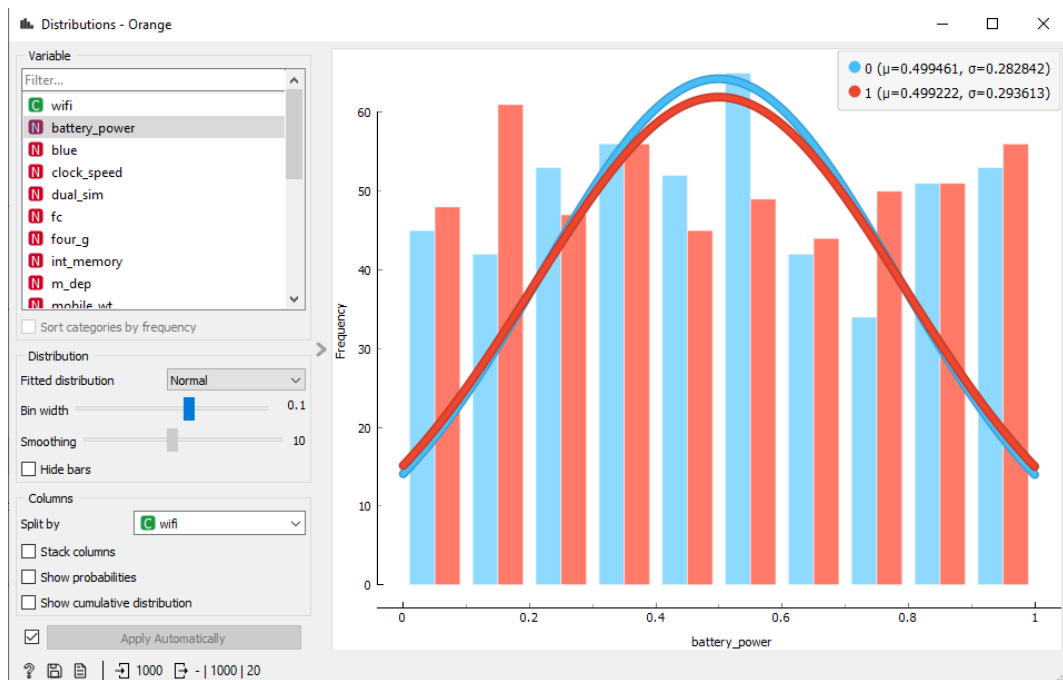
b) Histogrammas

Skārienekrāna klātbūtnes un akumulatora ietilpības sakaru histogramma



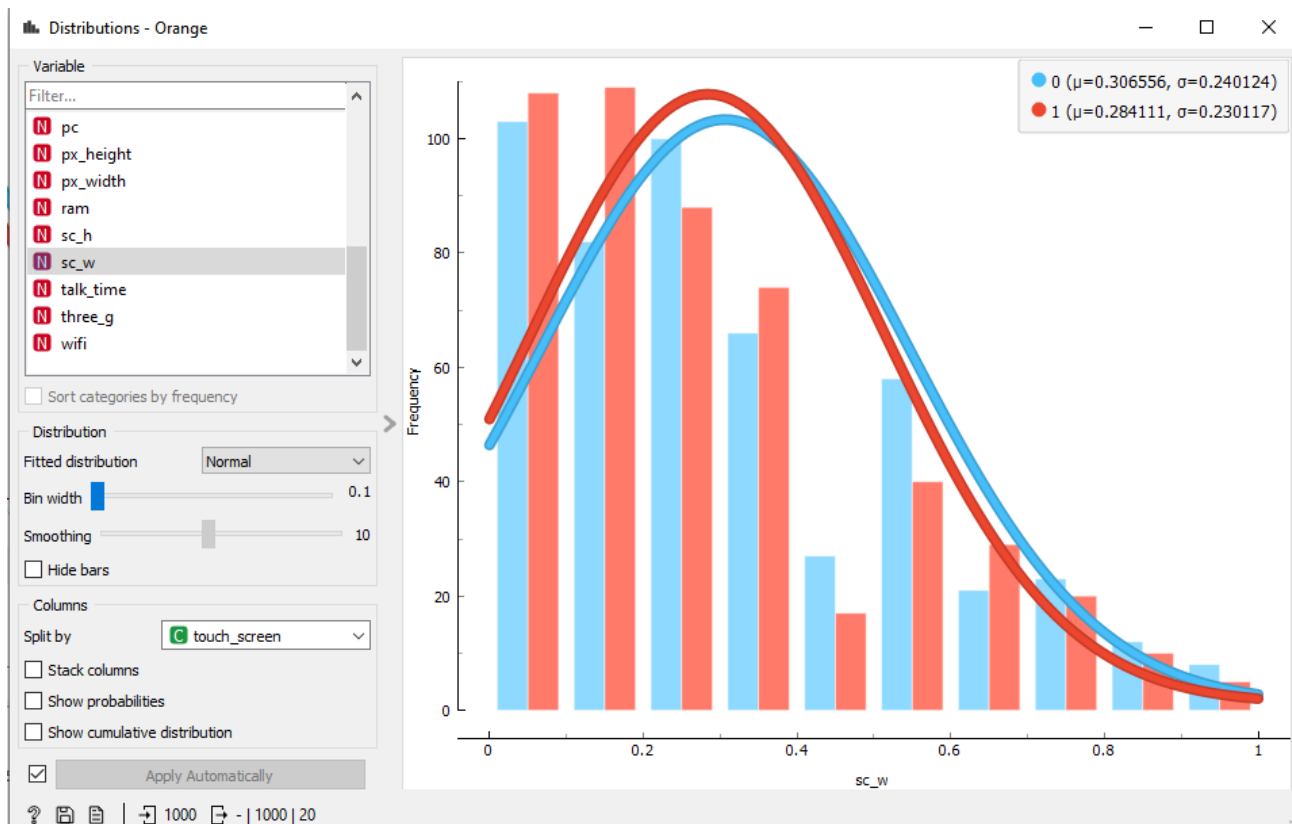
27.att Histogramma 1

Wi-Fi tehnoloģijas atbalsta pieejamība un akumulatora ietilpība sakaru histogramma



28.att Histogramma 2

c) Interesejošo pazīmju atspoguļojums



29.att Histogramma 3

d) Statistiskie rādītāji

Name	Distribution	Mean	Mode	Median	Dispersion	Min.	Max.	Missing
four_g		0.487	0.00	0.00	1.02635	0.00	1	0 (0 %)
wifi		0.507	1	1	0.986097	0.00	1	0 (0 %)
fc		0.241737	0.00	0.157895	0.971281	0.00	1	0 (0 %)
blue		0.516	1	1	0.968496	0.00	1	0 (0 %)
dual_sim		0.517	1	1	0.966559	0.00	1	0 (0 %)

Color: touch_screen

☒ Send Automatically

30.att Statistiskie rādītāji 1



31.att Statistiskie rādītāji 2

I daļas atbildes un secinājumi

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

Šajā 2. klases datu bāzē: tālruni ar un bez skārienpaliktņa. Spriežot pēc statistikas datiem, abas klases ir praktiski vienādas.

Vai datu vizuālais atspoguļojums ļauj redzēt datu struktūru?

Šīs datu kopas vizuālais atspoguļojums neļauj normāli redzēt datu struktūru, jo divu klašu dati Izklīdes diagrammā (att.24-att.26) sajaucas viens ar otru. Nav atkarības, pēc kuras varētu definēt datu struktūru. Arī histogrammā var redzēt, ka nevienai klasei nav pārsvara.

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

Šīs bāzes vizuālais attēlojums liek domāt, ka dati tiek sadalīti gandrīz vienmērīgi. Diagrammas arī parāda, ka dati tiek sakārtoti pēc konkrētas skaitliskas vērtības, veidojot sava veida datu grupas

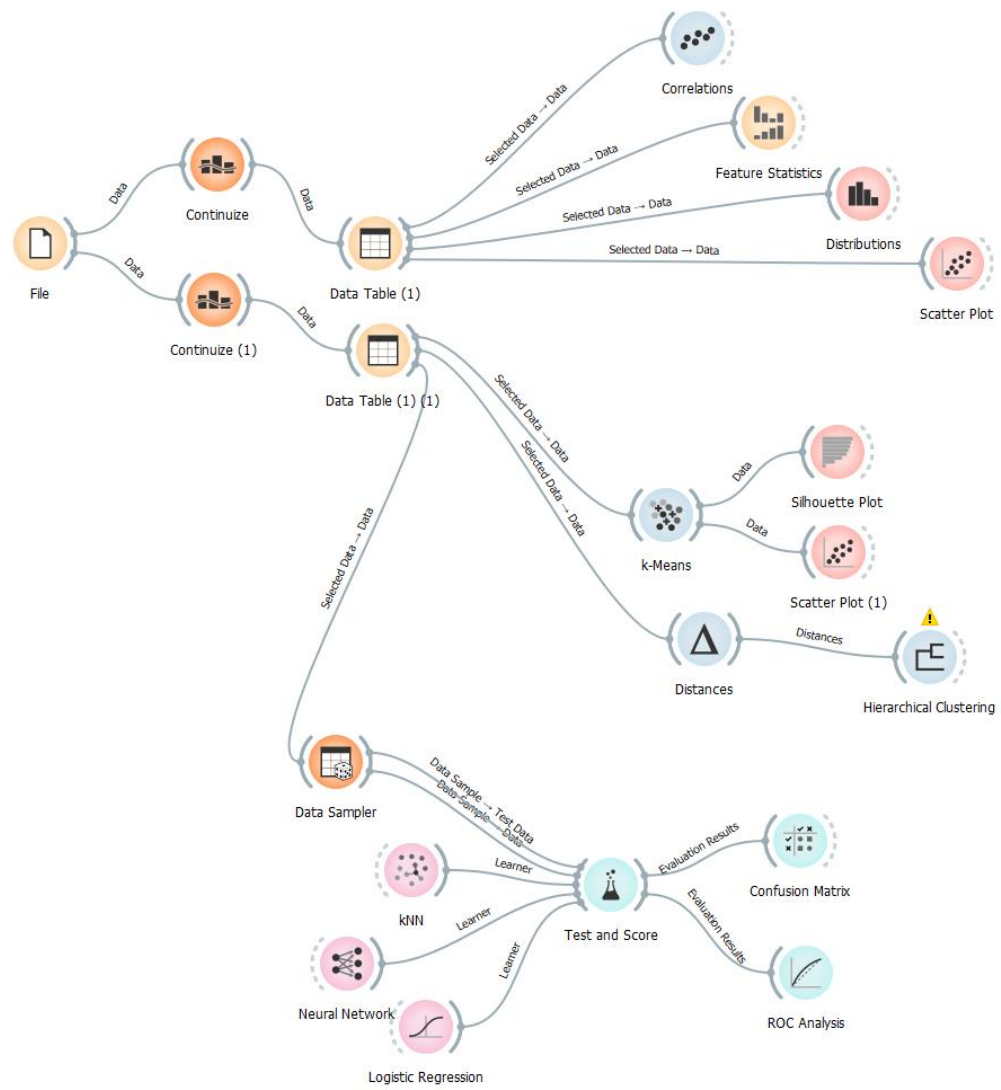
Vai identificētie datu grupējumi atrodas tuvu viens otram vai tālu viens no otra?

Identificētās datu grupas atrodas vienmērīgā attālumā viena no otras, tomēr pašas grupas dati nav vienmērīgi sadalīti

Secinājumi, kas izriet no statistisko rādītāju (vidējo vērtību un dispersijas vērtību) analīzes

Statistika liecina, ka maksimālā dispersija tālrunos, kas atbalsta 4G tehnoloģiju, kas nozīmē, ka tieši šis parametrs ir haotiskāks. Nav svarīgi, kuras opcijas ir ieslēgtas vai izslēgtas - 4G tehnoloģijas klātbūtne tālrunī nav atkarīga no citām tālruna specifikācijām.

Mediānas maksimālā vērtība tādos atribūtos kā wi-Fi, bluetooth, dual-sim, 3G. tā kā manas vērtības ir normalizētas no 0 līdz 1, tas nozīmē, ka šo atribūtu klātbūtne ir biežāka nekā nav.



32.att Orange modelis

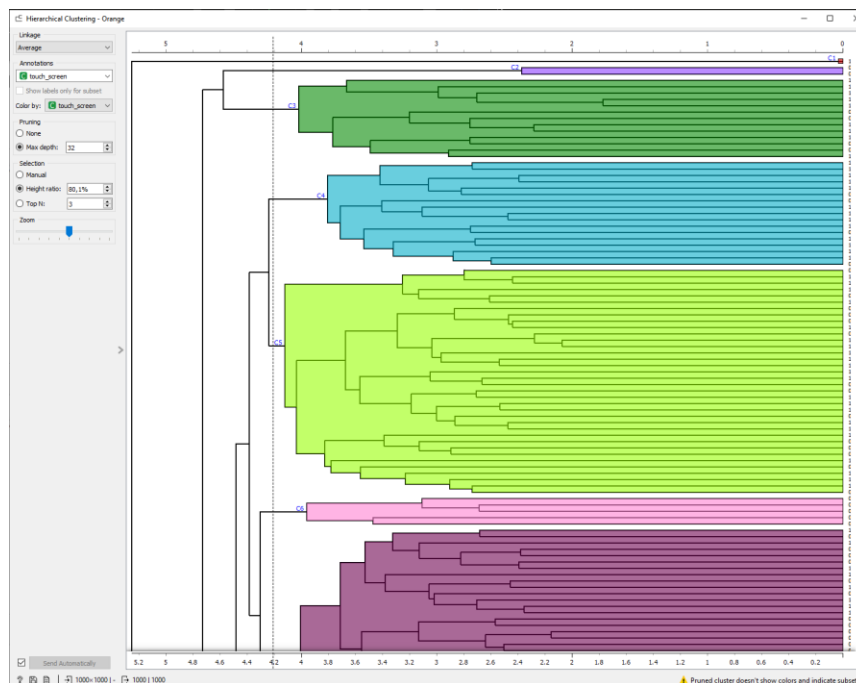
II daļa – Nepārraudzītā mašīnmācīšanās

Šajā darba daļā studenti veiks iepriekš izvēlētās datu kopas klasterizāciju. Darba I daļa sniedza studentiem izpratni par to, kādas pazīmes (atribūti) un klases ir datu kopā un cik labi datu objekti sadalās klasēs. Šīs darba daļas mērķis ir, izmantojot klasterizācijas metodes, vēl vairāk izpētīt datu kopu, lai noskaidrotu, vai iepriekš izdarītie secinājumi par datu kopas struktūru ir spēkā.

Lai izpildītu šo darba daļu, studentiem ir jāveic šādas darbības:

1. Jāpielieto divi studiju kursā apskatītie nepārraudzītās mašīnmācīšanās algoritmi: (1) hierarhiskā klasterizācija un (2) K-vidējo algoritms.
2. Hierarhiskās klasterizācijas algoritmam ir jāveic vismaz 3 eksperimenti, brīvi pārvietojot atdalošo līniju un analizējot, kā mainās klasteru skaits un saturs;
3. K-vidējo algoritmam ir jāaprēķina Silhouette Score vismaz 5 dažādām k vērtībām, un jāanalizē algoritma darbība.

Nepārraudzītās mašīnmācīšanās algoritmi



33.att Hierarhiskā klasterizācija 1

Orange rīkā pieejamie hiperparametri

Linkage - Logrīks atbalsta šādus attālumu mērīšanas veidus starp kopām:

- **Single** - aprēķina attālumu starp abu kopu tuvākajiem elementiem.
- **Average** - aprēķina vidējo attālumu starp abu kopu elementiem.
- **Weighted** - izmanto WPGMA metodi.
- **Complete** - aprēķina attālumu starp klasteru visattālākajiem elementiem.
- **Ward** - aprēķina kvadrātu kļūdas summas pieaugumu. Citiem vārdiem sakot, palātas minimālās dispersijas kritērijs samazina kopējo klasteru dispersiju.

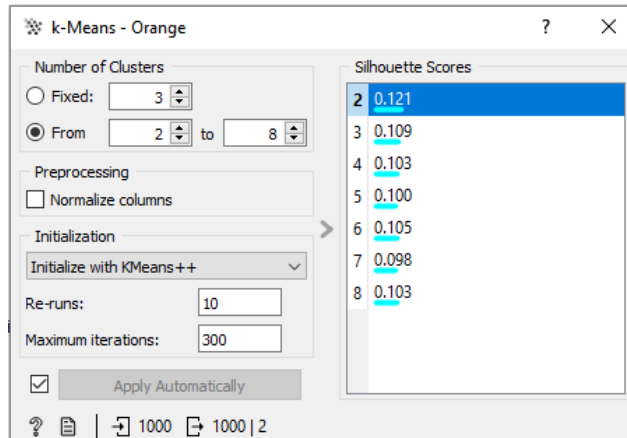
Annotations - Dendrogrammas mezglu etiķetes var izvēlēties Anotācijas lodziņā.

Pruning - Atzarošanas lodziņā var apgriezt milzīgas dendrogrammas, izvēloties maksimālo dendrogrammas dziļumu. Tas ietekmē tikai displeju, nevis faktisko kopu veidošanu. (**Max depth** vai **None**).

Selection - Logrīks piedāvā trīs dažādas atlasēšanas metodes:

- **Manual** - Noklikšķinot dendrogrammas iekšpusē, tiks atlasīts klasteris. Vairākas kopas var izvēlēties, turot Ctrl / Cmd. Katrs atlasītais klasteris tiek parādīts citā krāsā un tiek uzskatīts par atsevišķu kopu izvadē.
- **Height ratio** - Noklikšķinot uz dendrogrammas apakšējā vai augšējā lineāla, grafikā tiek ievietota griezumuma līnija. Tiek atlasīti vienumi pa labi no līnijas.
- **Top N** - Atlasa augšējo mezglu skaitu.

Zoom - Izmantojiet tālummaiņu un ritiniet, lai tuvinātu vai tālinātu.



34.att K-vidējo algoritms 1

Orange rīkā pieejamie hiperparametri

Number of Clusters - Izvēlieties kopu skaitu

- **Fixed** - algoritms klasteru datus uz noteiktu skaitu klasteru.
- **From X to Y** - logrīks parāda klasterizācijas rādītājus izvēlētajam klasteru diapazonam, izmantojot Silhouette score.

Preprocessing (Normalize columns) - Ja opcija ir atlasīta, kolonnas tiek normalizētas (Vidējais centrēts uz 0 un standartnovirze mērogs līdz 1).

Initialization - Inicializācijas metode (veids, kā algoritms sāk klasterizāciju)

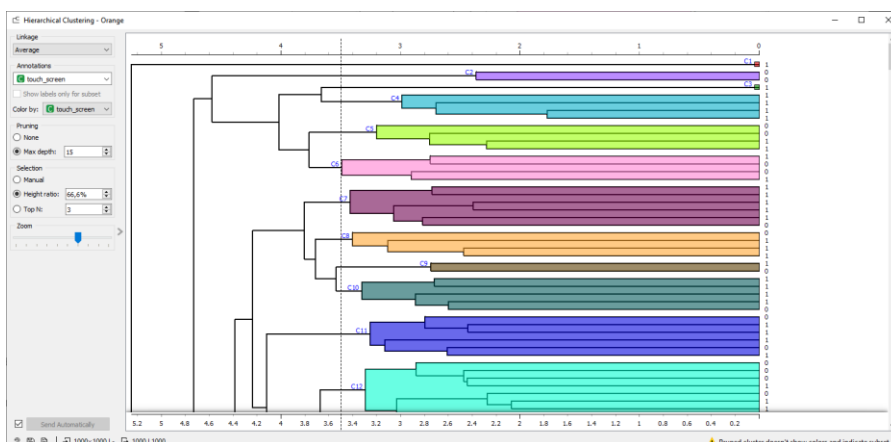
- **Initialize with Kmeans++** - pirmais Centrs tiek izvēlēts nejauši, pēc tam tiek izvēlēti no atlikušajiem punktiem ar varbūtību, kas ir proporcionāla attālumam kvadrātā no tuvākā centra
- **Random initialization** - klasteri vispirms tiek piešķirti nejauši un pēc tam atjaunināti ar turpmākām iterācijām

Re-runs - cik reizes algoritms tiek palaists no nejaušām sākotnējām pozīcijām; tiks izmantots rezultāts ar zemāko kvadrātu kopu summu.

Maximum iterations - maksimālo atkārtojumu skaitu katrā algoritma izpildē

Silhouette Scores - kontrastē vidējo attālumu līdz elementiem tajā pašā klasterī ar vidējo attālumu līdz elementiem citās kopās.

Hierarhiskās klasterizācijas algoritma eksperimenti

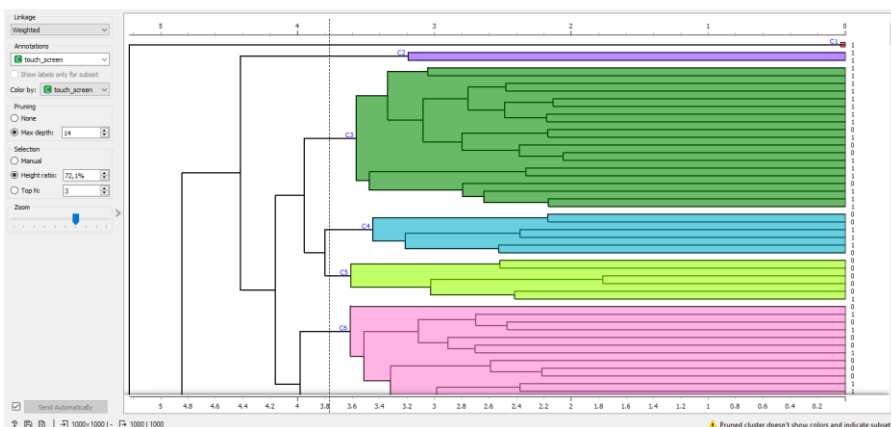


Linage: Average

Max Depth: 15

Height ratio: 66,6%

35.att Hierarhiskā klasterizācija 2

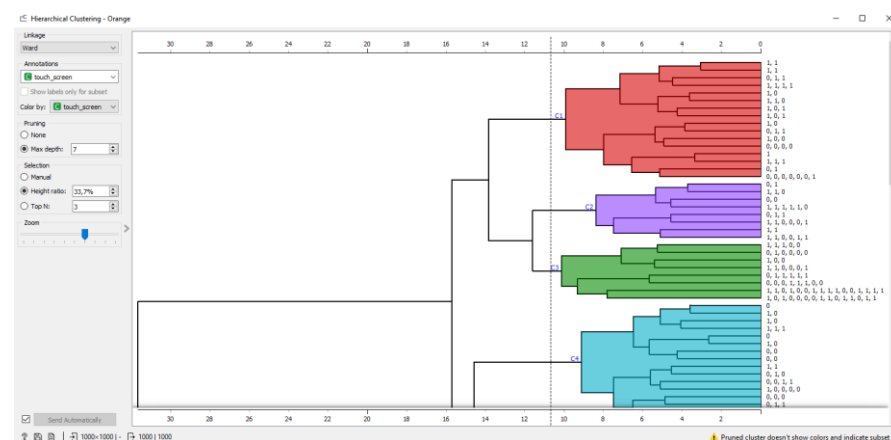


Linage: Weighted

Max Depth: 14

Height ratio: 72,1%

36.att Hierarhiskā klasterizācija 3



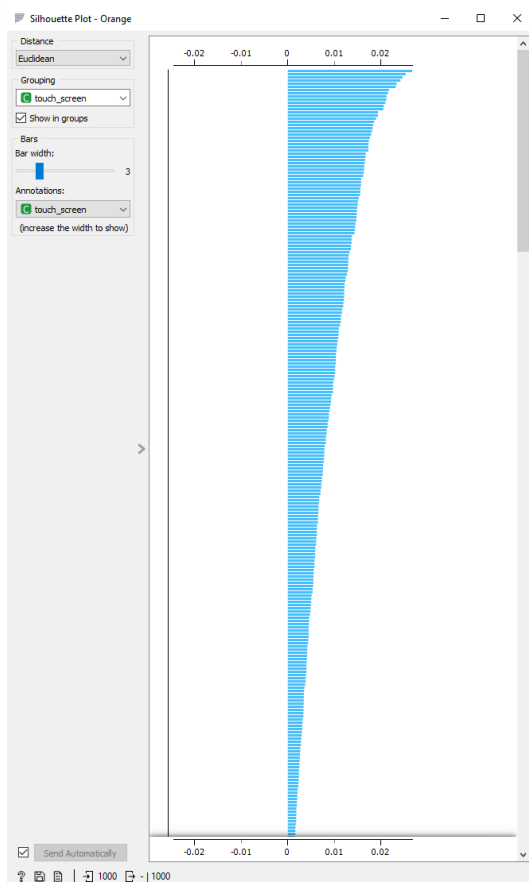
Linage: Ward

Max Depth: 7

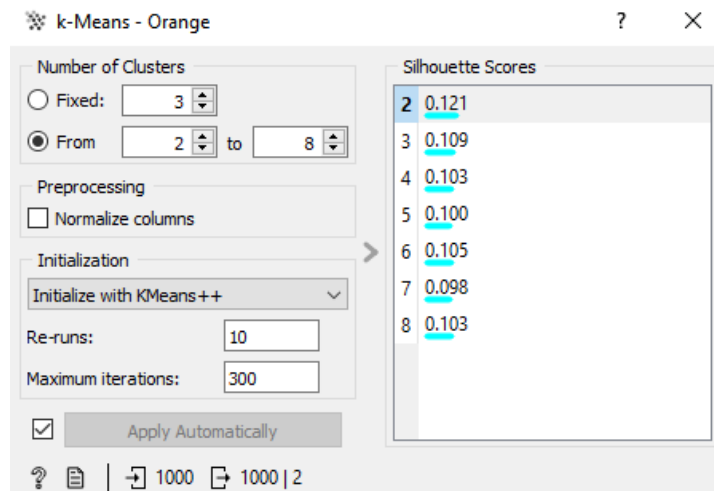
Height ratio: 33,7%

37.att Hierarhiskā klasterizācija 4

K-vidējo algoritma Silhouette Score



38.att Silhouette Score



39.att K-vidējo algoritms 2

Kaminsa algoritmam (att.38) tika izvēlēts diapazons no 2 līdz 8. Labākais rezultāts bija 1 klasteris ar augstāko vērtību (0,121). Rezultāti tiek parādīti, izmantojot Silhouette Plot rīku.

II daļas secinājumi

Pamatojoties uz darba otrās daļas rezultātiem, es saņēmu ne pārāk labus datus. Nekontrolēta mašīnmācīšanās darbojas ātri, bet dod neprecīzu rezultātu. Šīs darba daļas rezultātā man šķita, ka šie algoritmi nav piemēroti precīzākām darbībām ar datu apjomu profesionālā līmenī.

III daļa – Pārraudzītā mašīnmācīšanās

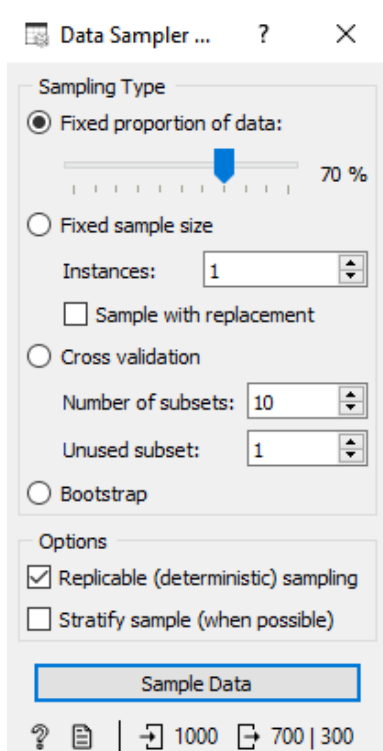
Šajā darba daļā studentiem ir jāpielieto vismaz 3 klasifikācijas algoritmi iepriekš izvēlētajai datu kopai. Viens no algoritmiem, kura izmantošana ir obligāta, ir mākslīgie neironu tīkli. Divus citus algoritmus studenti var brīvi izvēlēties.

Lai izpildītu šo darba daļu, studentiem ir jāveic šādas darbības:

1. Ir jāizvēlas vismaz divi pārraudzītās mašīnmācīšanās algoritmi, kas ir paredzēti klasifikācijas uzdevumam. Studenti drīkst izmantot studiju kursā aplūkotos algoritmus vai arī jebkurus citus algoritmus, kuri ir paredzēti klasifikācijas uzdevumam.
2. Ir jāsadala datu kopa apmācību un testa datu kopās.
3. Katram algoritmam, lietojot apmācību datu kopu, ir jāveic vismaz 3 eksperimenti, mainot algoritma hiperparametru vērtības un analizējot algoritmu veikspējas metrikas;
4. Katram algoritmam ir jāizvēlas tas apmācītais modelis, kas nodrošina labāko algoritma veikspēju;
5. Katra algoritma apmācītais modelis ir jāpielieto testa datu kopai.
6. Ir jānovērtē un jāsalīdzina apmācīto modeļu veikspēja.

Pārraudzītās mašīnmācīšanās algoritmi

Tika izvēlēti kNN algoritmi un loģistiskā regresija, kā arī obligātais algoritms mākslīgie neironu tīkli.



40.att Data Sampler

kNN algoritms

K-tuvāko kaimiņu algoritms (kNN) ir viens no vienkāršajiem mašīnmācīšanās algoritmiem, ko izmanto klasifikācijai un regresijai. Tas balstās uz objektu tuvuma principu daudzdimensionālā datu telpā.

kNN algoritmu ir viegli ieviest, un tam ir vairākas priekšrocības, piemēram, nav datu priekšapstrādes vai modeļa apmācības prasības. Tomēr tam ir arī daži trūkumi, tostarp augsta skaitļošanas sarežģītība

Es izvēlējos kNN algoritmu, jo mēs to pētījām lekcijās un tas man ir vairāk pazīstams nekā pārējie. Arī tāpēc, ka tas ir viegli saprotams algoritms un elastīgs un var pielāgoties dažāda veida uzdevumiem.

kNN hiperparametri

Number of neighbors – lestatiet tuvāko kaimiņu skaitu. un svarus kā modeļa kritērijus

Metric - attāluma parametrs.

- **Euclidian** - "taisna līnija", attālums starp diviem punktiem.
- **Manhattan** - visu atribūtu absolūto atšķirību summa.
- **Chebyshev** - lielākā no absolūtajām atšķirībām starp atribūtiem.
- **Mahalanobis** - attālums starp punktu un sadalījumu.

Weight - modeļa kritēriji

- **Uniform** - visi punkti katrā apkārtnē tiek svērti vienādi.
- **Distance** - vaicājuma punkta tuvākiem kaimiņiem ir lielāka ietekme nekā kaimiņiem tālāk.

Logistiskā regresija

Logiskā regresija ir mašīnmācīšanās algoritms, ko izmanto, lai atrisinātu binārās klasifikācijas problēmas, kad ir jāparedz novērošanas piederība vienai no divām klasēm. Tas ir balstīts uz 1. klases varbūtības koeficienta (Logit) logaritma modelēšanu.

Logiskā regresija ir viens no visplašāk izmantotajiem algoritmiem binārajai klasifikācijai, pateicoties tā vienkāršībai, rezultātu interpretācijai un labai veiktspējai daudzos uzdevumos. Tomēr tam ir savi priekšnoteikumi, piemēram, atkarības linearitāte starp pazīmēm un mērķa mainīgo, un tas var nedarboties efektīvi sarežģītu pazīmju mijiedarbību gadījumā.

Es izvēlējos logistikas regresijas algoritmu, jo tas man šķita visvieglāk saprotams, kā arī efektīvs ar lielu datu kopu.

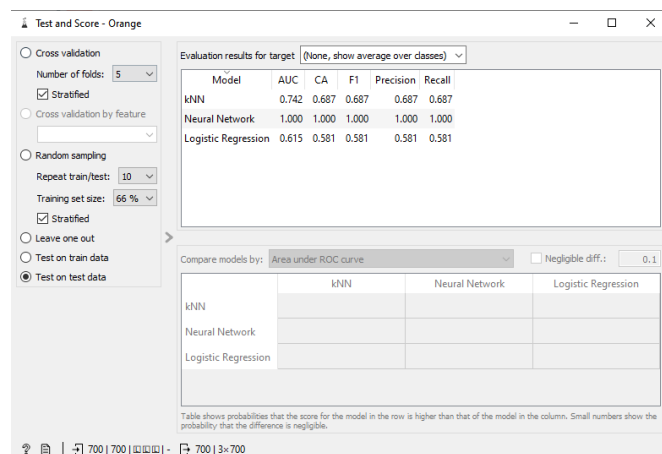
Logistiskā regresija hiperparametri

Regularization type – Ridge vai Lasso regulēšanu.

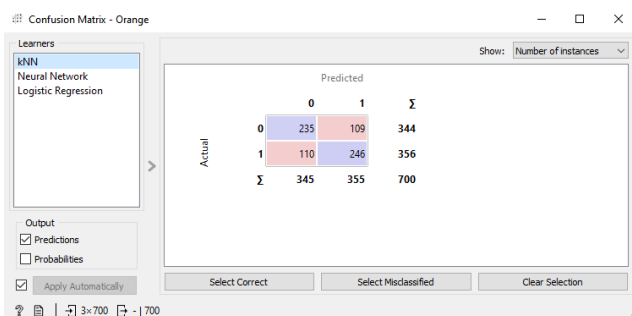
Strength - var mainīt spēku, $C=1$ kā noklusējuma vērtību.

Eksperimenti

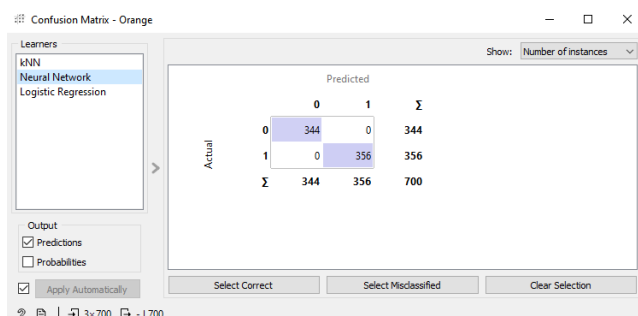
1. Eksperiments



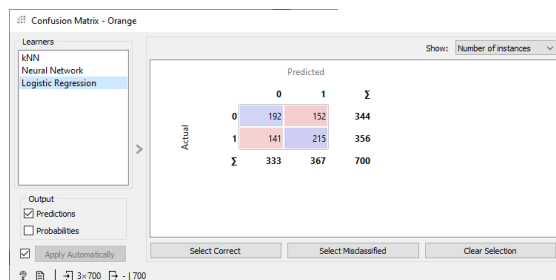
41.att Test and Score 1



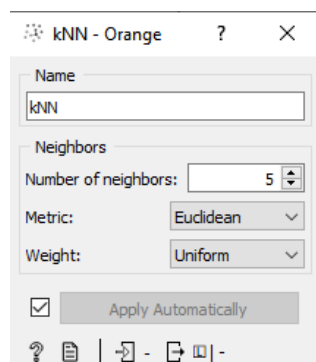
42.att Confusion matrix (kNN) 1



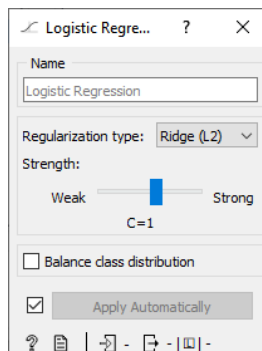
43.att Confusion matrix (Mākslīgie neironu tīkli) 1



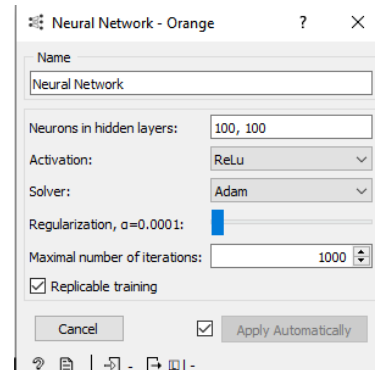
44.att Confusion matrix (Logistiskā regresija) 1



45.att kNN iestatījums 1

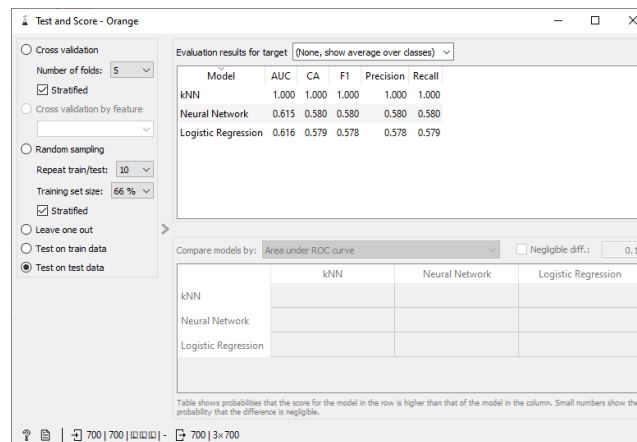


46.att Logistiskā regresija iestatījums 1

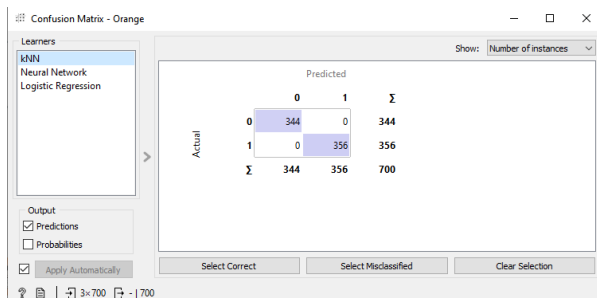


47.att Neironu tīkli iestatījums 1

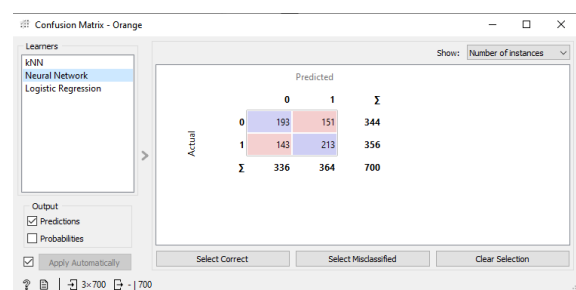
2. Eksperiments



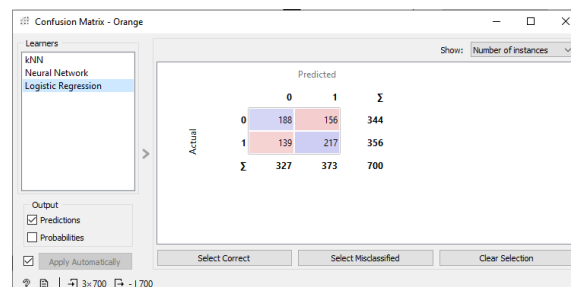
48.att Test and Score 2



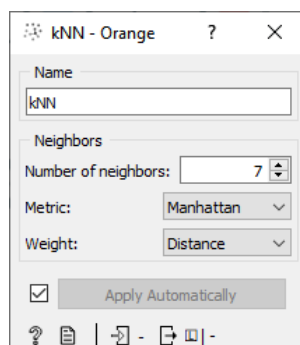
49.att Confusion matrix (kNN) 2



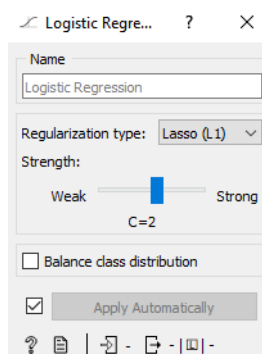
50.att Confusion matrix (Mākslīgie neironu tīkli) 2



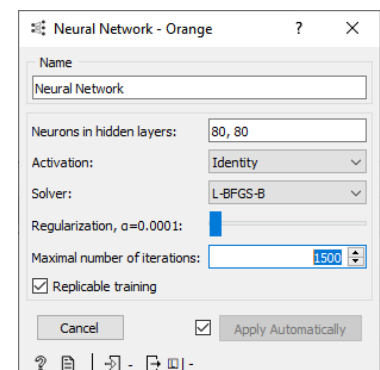
51.att Confusion matrix (Logistiskā regresija) 2



52.att kNN iestatījums 2

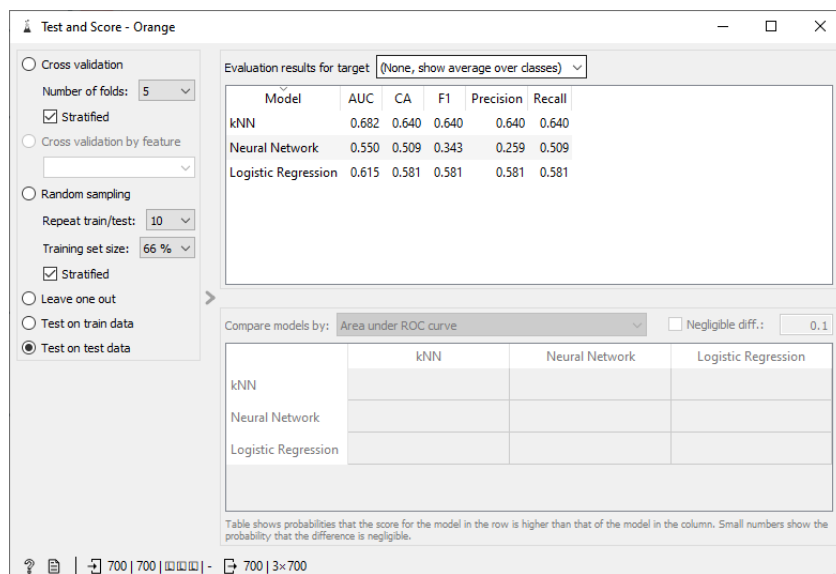


53.att Loģistiskā regresija iestatījums 2

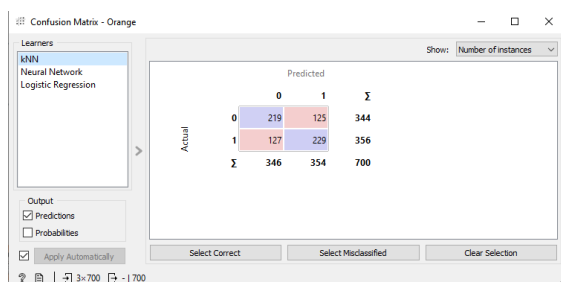


54.att Neironu tīkli iestatījums 2

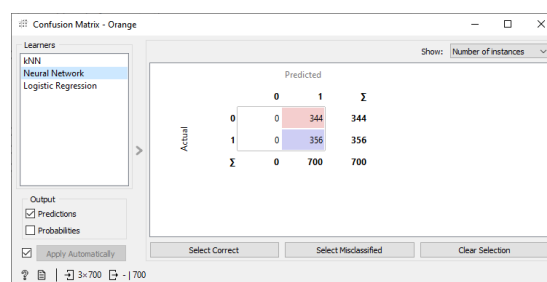
3. Eksperiments



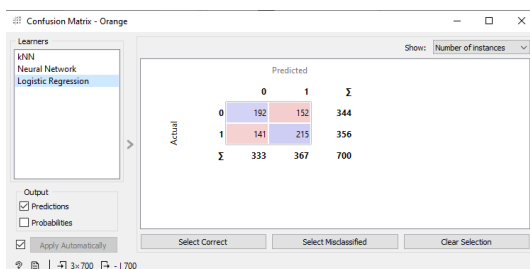
55.att Test and Score 3



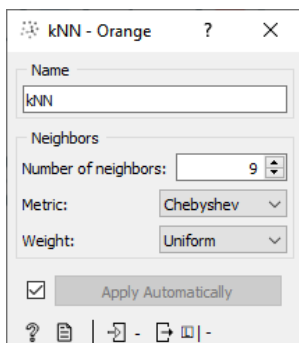
56.att Confusion matrix (kNN) 3



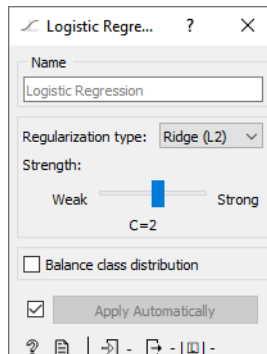
57.att Confusion matrix (Mākslīgie neironu tīkli) 3



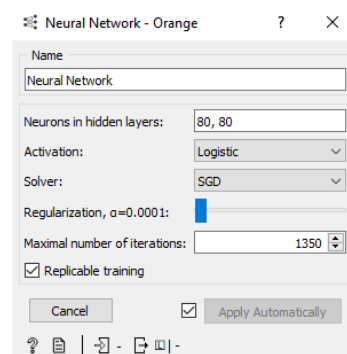
58.att Confusion matrix (Loģistiskā regresija) 3



59.att kNN iestatījums 3



60.att Loģistiskā regresija iestatījums 3



61.att Neironu tīkli iestatījums 3

Apmācītu modeļu salīdzinājums pēc algoritmu veikspējas

Tika veikti 3 testi, lai pārbaudītu algoritmu precizitāti un efektivitāti.

1. Eksperimenta rezultāti:

Pirmajā eksperimentā neironu tīklu algoritms uzrādīja labākos rādītājus (41.att). Visi rādītāji, ieskaitot CA un Precision, bija 100%, savukārt loģistiskajā regresijā CA un Precision bija 58,1%, bet kNN-68,7%. Tas ir ļoti labs rezultāts.

2. Eksperimenta rezultāti:

Uzsākot otro eksperimentu (48.att), es nolēmu mainīt pirmā testa uzvarētāja ievadi, lai redzētu kā viņš šoreiz rīkosies. Tas patiešām ietekmēja rezultātu, un tagad neironu tīklu algoritma CA un Precision parametri ir pasliktinājušies-58%.

Arī es mainīju citu algoritmu parametrus, un par pārsteigumu tagad kNN algoritms uzrādīja perfektu rezultātu 100%, bet loģistiskās regresijas algoritms mainījās nedaudz: CA - 57,9% un Precision - 67,8%.

3. Eksperimenta rezultāti:

Trešajā eksperimentā es atkal nomainīju visu algoritmu ievadi un tas bija visveiksmīgākais eksperiments, jo visu algoritmu dati pasliktinājās(55.att).

	CA	Precision
kNN	64%	64%
Mākslīgie neironu tīkli	50,9%	25,9%
Loģistiskā regresija	58,1%	58,1%

Tomēr vislabākais rezultāts bija loģistiskās regresijas algoritmā ar rezultātu 58%.

Secinājumi

Darba laikā tika analizēta datu kopa, kā arī izmantotas dažādas mašīnmācīšanās metodes. Šajā darbā mēs izmantojām nepārraudzītās algoritmus (hierarhiska klasterizācija un K-vidējo algoritmi) un pārraudzītās (kNN, Loģistikas regresija un neironu savienojumi). Šajā darbā es daudz uzzināju par mašīnmācīšanās algoritmiem gan teorijā, gan praksē.

Viss darbs tika veikts Orange programmā, kas tikai palielināja interesi par darba izpildi, jo šī programma man patika ar savu vienkāršību, ērtību un patīkamo saskarni. Šī programma patiešām palīdz tikt galā ar noteiktu datu masīvu.

Tomēr, neskatoties uz visiem darba plusiem, tomēr visgrūtāk bija atrast piemērotu datu bāzi, jo ne visi var atbilst darba kritērijiem (digitālās vērtības, datu apjoms).

Apkopojot darbu, man šķita, ka pārraudzītās mašīnmācīšanās metodes ir precīzākas un labāk veic uzdevumu nekā nepārraudzītās. Tomēr, lai sasniegtu vēlamo rezultātu, pārraudzītiem mašīnmācīšanās algoritmiem ir nepieciešama pareiza konfigurācija.

Avoti

- <https://www.kaggle.com/datasets/iabhishekoofficial/mobile-price-classification?select=test.csv>
- <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/knn.html>
- <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/model/logisticregression.html>
- <https://towardsdatascience.com/data-science-made-easy-data-modeling-and-prediction-using-orange-f451f17061fa>
- <https://orangedatamining.com/widget-catalog/unsupervised/kmeans/>
- <https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/unsupervised/hierarchicalclustering.html>