



Vilniaus Universitetas

softmax

# Dirbtinio intelekto pagrindai

3 užduotis. Tiesioginio sklaidimo DNT naudojant sistemą WEKA

Darbą atliko:

Vilniaus universiteto matematikos ir informatikos fakulteto

duomenų mokslo 3 kurso 2 grupės studentas

Matas Amšiejus

Vilnius, 2021 11 16

## Turinys

Tikslas.....	3
1 uždutis. Duomenų paruošimas.....	4
2 uždutis. Mokslinio darbo sekos sukonstravimas .....	4
3 uždutis. Neuroninio tinklo parametrų parinkimas .....	5
4 uždutis. Naujų duomenų klasifikavimas.....	6
5 uždutis. Klasifikavimas ir testavimas .....	6
6 uždutis. Neurono išėjimo reikšmių perskaičiavimas Excel .....	8
Išvados .....	12
Priedai .....	13
4 užduoties testinio failo klasifikavimo rezultatai .....	13

## Tikslas

Sukurti ir apmokyti dirbtinį neuroninį tinklą, skirtą klasifikuoti vilkdalgių duomenis į tris grupes. Išbandyti tinklo veikimą praktiškai. Palyginti sistemos WEKA ir rankiniu būdu gautus rezultatus.

## 1 užduotis. Duomenų paruošimas

Šiame projekte bus naudojami modifikuoti Iris.arff duomenys, paimti iš WEKA duomenų aplanko. Sukursiu 2 naujus duomenų failus: *iris\_train\_test.arff* ir *iris\_new.arff*. Pirmajame bus atsitiktinai parinkti 40 įrašų (gėlių) kiekvienai klasei (viso 120), o antrajame – likę 10 (viso 30). Duomenų atrinkimui naudosiu R programinę kalbą. Pirma sukūriau *iris.arff* failo kopiją *iris - Copy.csv* (taip pat ištryniau nereikalingą informaciją). Sukūrus failą, jis buvo .csv formate, todėl rankiniu būdu įkliajavau būtiną .arff failo veikimui kodą, nustačiau atitinkamus pavadinimus. Duomenis įkėliau atgal į WEKA duomenų aplanką. Programos kodas:

```
library(readr)
library(dplyr)
#nuskaityme duomenis
iris_Copy <- read_csv("iris - Copy.csv", col_names = FALSE)
#priskiriame laukeliams id
iris_Copy$id <- 1:nrow(iris_Copy)

#atrenkame tik setosa geles
setosa <- iris_Copy %>% filter(X5 == "Iris-setosa")
#sukuriame setosu mok_test ir new aibes
trte_setosa <- setosa %>% dplyr::sample_frac(.80)
test_setosa <- dplyr::anti_join(setosa, trte_setosa, by = 'id')

#atrenkame tik setosa geles
versicolor <- iris_Copy %>% filter(X5 == "Iris-versicolor")
#sukuriame setosu mok_test ir new aibes
trte_versicolor <- versicolor %>% dplyr::sample_frac(.80)
test_versicolor <- dplyr::anti_join(versicolor, trte_versicolor, by = 'id')

#atrenkame tik setosa geles
virginica <- iris_Copy %>% filter(X5 == "Iris-virginica")
#sukuriame setosu mok_test ir new aibes
trte_virginica <- virginica %>% dplyr::sample_frac(.80)
test_virginica <- dplyr::anti_join(virginica, trte_virginica, by = 'id')

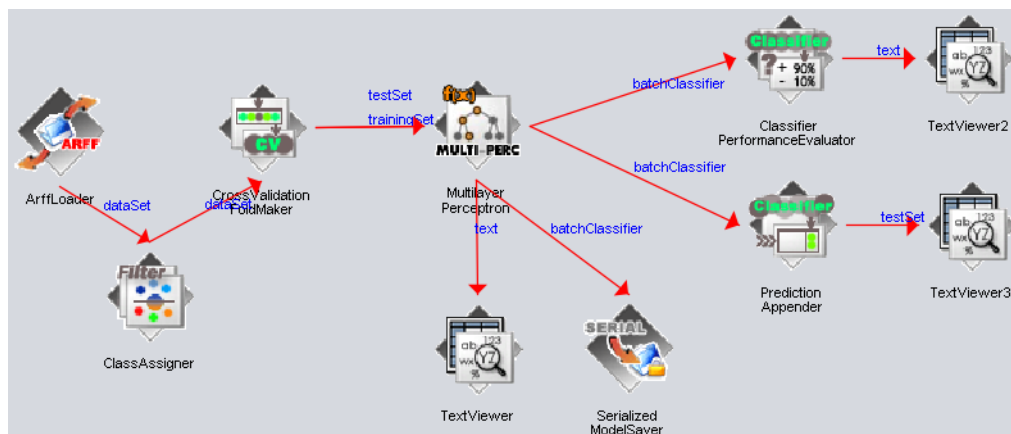
#sujungiamo train test (trte) i viena faila, test i kita
trte <- rbind(trte_setosa, trte_versicolor, trte_virginica)
test <- rbind(test_setosa, test_versicolor, test_virginica)
#ismetame nebereikalinga id stulpeli
trte <- trte[-ncol(trte)]
test <- test[-ncol(test)]

#issaugome i naujus duomenu failus
write.table(trte,"iris_train_test.csv", quote = F, sep = ",", row.names = F, col.names = F)
write.table(test,"iris_new.csv", quote = F, sep = ",", row.names = F, col.names = F)
```

## 2 užduotis. Mokslinio darbo sekos sukonstravimas

Sistemoje WEKA sukūriau mokslinio darbo seką. *ArffLoader* atlieka failo parinkimo funkciją, *ClassAssigner* pasako, kad paskutinis atributas yra klasė, *Cross Validation Foldmaker* pasako, kad bus atliekama kryžminė patikra (aš visur naudosiu numatytąją – dešimties bloką). *Multilayer Perceptron* yra daugiasluoksnis perceptronas (pas mus jis bus naudojamas kaip tiesioginio sklaidimo neuroninis tinklas). Šioje dalyje bus keičiami mūsų neuroninio tinklo parametrai (paslėptų sluoksnių skaičius, mokymosi greitis, momentas...). *TextViever* mums parodys jau po apmokymo parinktus svorius (kiekvienam kryžminės patikros etapui). *SerializedModelSaver* išsaugo visus apskaičiuotus

modelius (mūsų atveju 10). *ClassifierPerformanceEvaluator* atlieka klasifikatoriaus įvertinimo funkciją – mums parodo klasifikavimo tikslumą, klasifikavimo matricą ir daug kitų parametrų. *PredictionAppender* spausdina testavimo duomenis ir tikimybes būti priskirtiems kokiai nors klasei (jei *Append Probabilities* yra *TRUE*).



1 pav. Mokslinio darbo seka trečiai užduočiai

### 3 užduotis. Neuroninio tinklo parametrų parinkimas

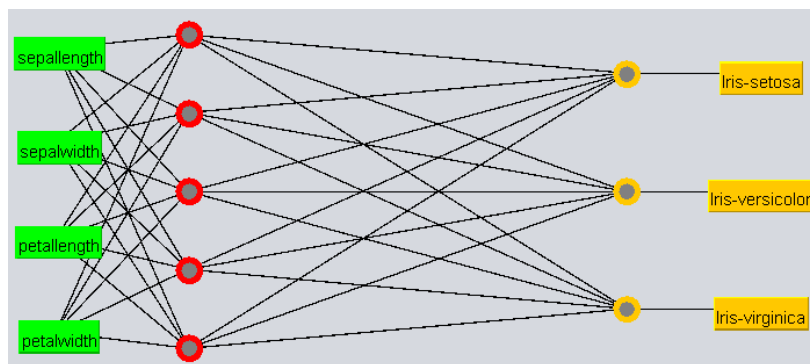
Užduočiai naudosime ankstesniame punkte sukurtą darbo seką su mokymosi greičiu  $\eta=0,3$  ir momento konstanta  $\alpha=0,2$ . Su 1 paslėptu neuronu iš 120 gėlių klaidingai išskirstomos 4-ios (klas. tikslumas 96,67 %). Su 2 paslėptais neuronais – 1 klaida (klas. tikslumas 99,17 %). Toliau iki 10 (daugiau netestavau) klasifikavimo tikslumas išlieka toks pats (119 teisingai suklasifikuotų, 1 neteisingai). Toliau naudosis 5 paslėptus neuronus.

Pabanžius padidinti sluoksnių skaičių (su 5 paslėptais neuronais kiekviename) klasifikavimo tikslumas nepakito, o paėmus 2 sluoksnius su 3 paslėptais neuronais – nukrito iki 98,33 %. Toliau imsiu vieną sluoksnį paslėptų neuronų.

Pabandžius keisti mokymosi greitį gavau tuos pačius klasifikavimo tikslumo rezultatus nuo  $\eta=0,1$  iki  $0,9$  lygius 99,17 %. Sumažinus mokymosi greitį iki  $0,01$  gavau prastesnius rezultatus: teisingai suklasifikuota buvo tik 112 gėlių. Toliau imsiu  $\eta=0,3$ .

Keičiant momento konstantą ( $\alpha$ ) klasifikavimo tikslumas išlieka toks pats nuo  $0,1$  iki  $0,8$  (119 teisingai ir 1 neteisingai (99,17 %)), o nuo  $0,9$  klaidingai suklasifikuotų duomenų skaičius didėja. Tolimesniam tikrinimui fiksuosiu  $\alpha=0,2$ .

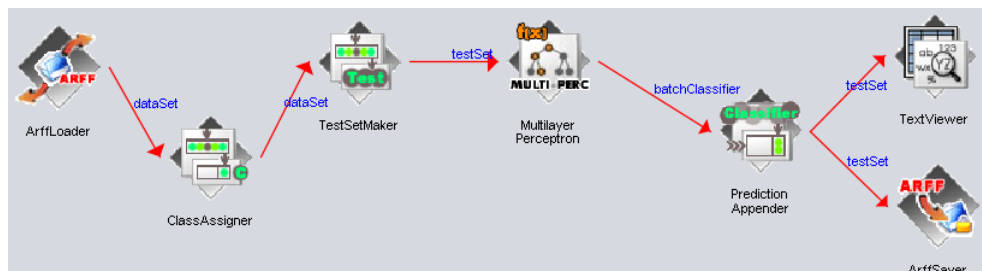
Galima padaryti išvadą, kad mūsų uždaviniui parametrų keitimas neturėjo didelės įtakos. Svarbiausia nepaimti per daug ekstremalių reikšmių, kaip labai mažas ar didelis mokymosi greitis, vienas paslėptas neuronas ar panašiai. Išsaugome neuroninio tinklo modelį su mūsų užfiksuotais parametrais.



2 pav. Parinkto neuroninio tinklo vaizdas.

#### 4 užduotis. Naujų duomenų klasifikavimas

Šiai užduočiai naudosime naują darbo seką. ArffLoader dabar panaudosime iris\_new.arff duomenis (30 gėlių), kurie bus lyg realaus neuroninio tinklo simuliacija. Atsiranda nauja komponentė TestSetMaker, kuri nurodo, kad duomenys bus skirti tik testavimui. Multilayer Perceptron sluoksnyje nurodome 3 užduotyje sukurtą modelį.

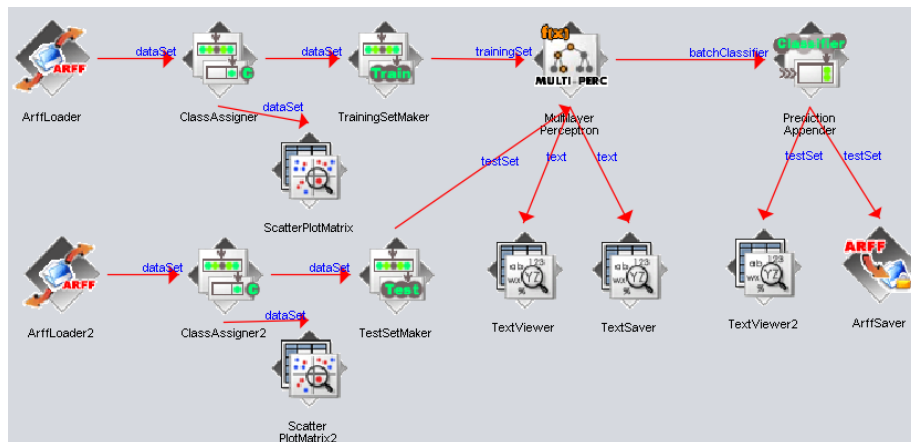


3 pav. Mokslinio darbo seka ketvirtai užduočiai

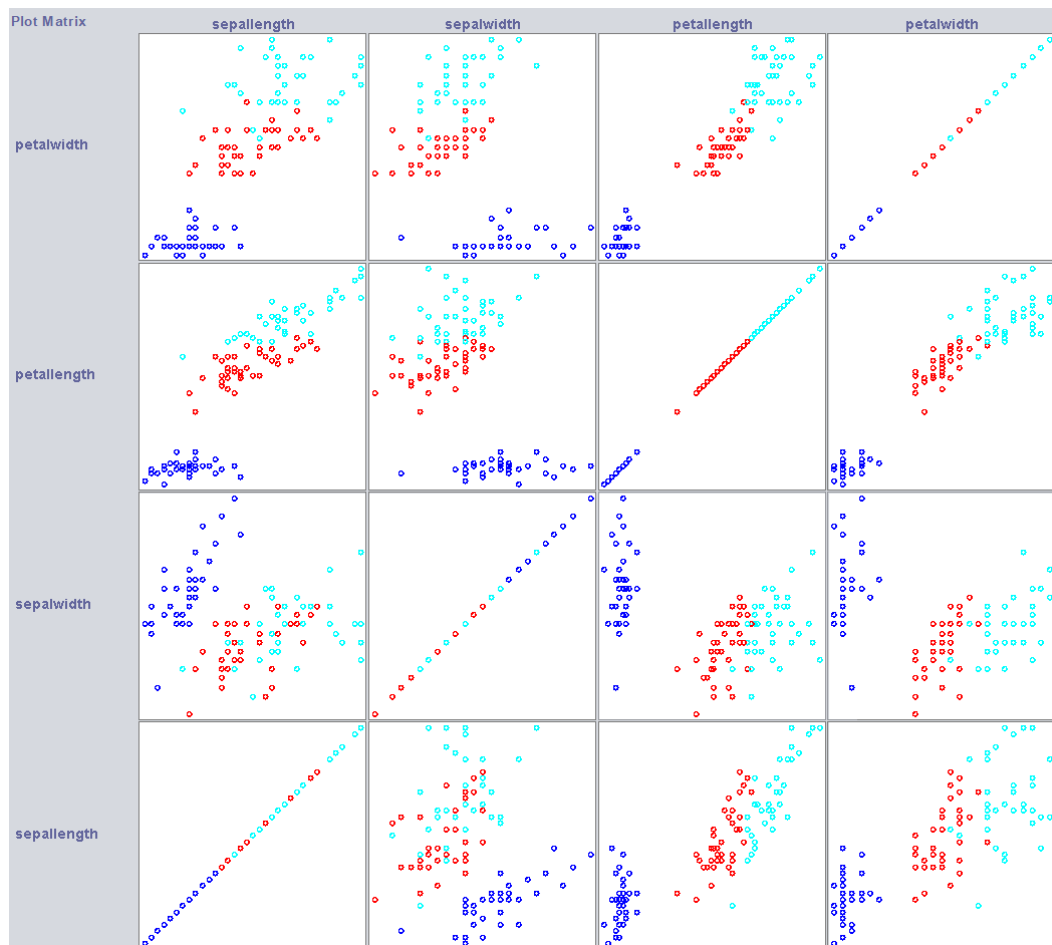
Gauname, kad naudojant 3 užduoties modelį ant mūsų „naujų“ vilkdalgių duomenų, klasifikavimo tikslumas yra 90 %, t. y. 27 teisingai suklasifikuotos gėlės iš 30. Kaip ir buvo galima tikėtis iš trečios užduoties modelio klasifikavimo matricos, problematiškiausios buvo virginica ir versicolor klasės (žr. 4 užduoties testinio failo klasifikavimo rezultatai)

#### 5 užduotis. Klasifikavimas ir testavimas

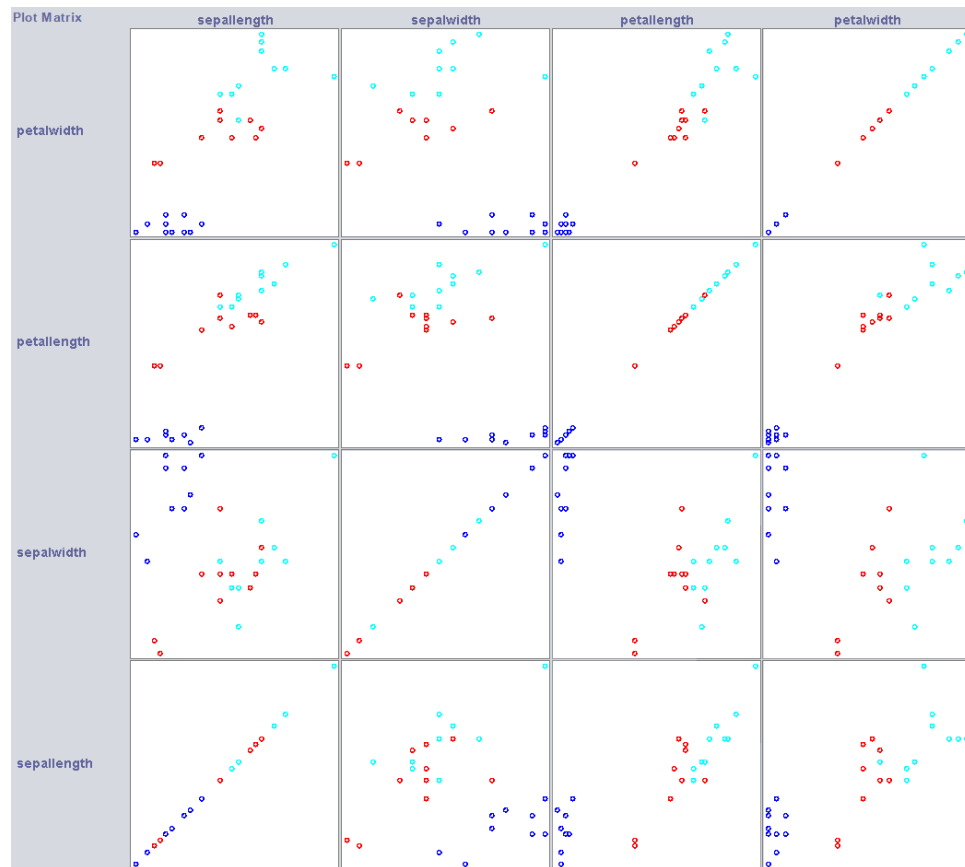
Šiai užduočiai naudosime naują darbo seką. ArffLoader panaudosime iris\_train\_test.arff, o ArffLoader2 panaudosime iris\_new.arff duomenis. Pirmieji bus naudojami mokymui (120 gėlių), o pastarieji – testavimui (30 gėlių). Taip pat su ScatterPlotMatrix brėšime vilkdalgių požymių porų vaizdus Dekarto koordinatų sistemoje. Paduodame duomenis į daugiasluksnį perceptroną. Gausime 2 duomenų failus. Viename bus rasti svoriai kiekvienam neuronui, kitame – testavimo duomenų klasifikavimo rezultatai (su tikimybėmis būti priskirti kiekvienam gėlės tipui). Gauname, kad teisingai suklasifikuota 93,33% (28 iš 30 duomenų). Žvelgiant į duomenis matome, kad vieną kartą versicolor buvo priskirta virginica, o antrą kartą priešingai. Taip pat 25 eilutėje matome, kad virginica buvo priskirta teisingai, tačiau gana „neužtikrintai“ – tik su 66 % tikimybe.



4 pav. Mokslinio darbo seka penktai užduočiai



5 pav. Vilkdagių iris\_train\_test požymių porų vaizdai Dekarto koordinacių sistemoje. Čia ir toliau tamsiai mėlyna yra setosa, raudona – versicolor, žydra - virginica



6 pav. Vilkdagių iris\_new požymių porų vaizdai Dekarto koordinacių sistemoje

## 6 uždutis. Neurono išėjimo reikšmių perskaičiavimas Excel

1 lentelė. 5 uždutyje gauti testuojamų gėlių parametrai, sunormuoti parametrai, tikrosios reikšmės ir priskyrimo tikimybės

	x1	x2	x3	x4	Norm x1	Norm x2	Norm x3	Norm x4	Tikra reikšmė	Setosa tik.	Versicolor tik.	Virginica tik.
X1	5,4	3,7	1,5	0,2	-0,515152	0,8666667	-0,921569	-1	Iris-setosa	0,989222	0,010777	0,000001
X2	5,7	3,8	1,7	0,3	-0,333333	1	-0,843137	-0,913043	Iris-setosa	0,988445	0,011555	0,000001
X3	5,1	3,8	1,5	0,3	-0,69697	1	-0,921569	-0,913043	Iris-setosa	0,989317	0,010683	0,000001
X4	5,1	3,7	1,5	0,4	-0,69697	0,8666667	-0,921569	-0,826087	Iris-setosa	0,988582	0,011418	0,000001
X5	5,2	3,4	1,4	0,2	-0,636364	0,4666667	-0,960784	-1	Iris-setosa	0,988643	0,011356	0,000001
X6	5,4	3,4	1,5	0,4	-0,515152	0,4666667	-0,921569	-0,826087	Iris-setosa	0,986265	0,013735	0,000001
X7	5,5	3,5	1,3	0,2	-0,454545	0,6	-1	-1	Iris-setosa	0,988895	0,011104	0,000001
X8	4,8	3	1,4	0,3	-0,878788	-0,066667	-0,960784	-0,913043	Iris-setosa	0,985939	0,01406	0,000001
X9	5,1	3,8	1,6	0,2	-0,69697	1	-0,882353	-1	Iris-setosa	0,989504	0,010496	0,000001
X10	4,6	3,2	1,4	0,2	-1	0,2	-0,960784	-1	Iris-setosa	0,988512	0,011487	0,000001
X11	6,5	2,8	4,6	1,5	0,1515152	-0,333333	0,2941176	0,1304348	Iris-versicolor	0,003164	0,995309	0,001526
X12	4,9	2,4	3,3	1	-0,818182	-0,866667	-0,215686	-0,304348	Iris-versicolor	0,025301	0,974617	0,000082
X13	6,6	2,9	4,6	1,3	0,2121212	-0,2	0,2941176	-0,043478	Iris-versicolor	0,005098	0,994555	0,000347



X14	6,7	3,1	4,4	1,4	0,2727273	0,0666667	0,2156863	0,0434783	Iris-versicolor	0,005962	0,993772	0,000266
X15	6	2,9	4,5	1,5	-0,151515	-0,2	0,254902	0,1304348	Iris-versicolor	0,003785	0,995035	0,00118
X16	6	2,7	5,1	1,6	-0,151515	-0,466667	0,4901961	0,2173913	Iris-versicolor	0,000408	0,104949	0,894643
X17	6	3,4	4,5	1,6	-0,151515	0,4666667	0,254902	0,2173913	Iris-versicolor	0,005758	0,993722	0,00052
X18	5	2,3	3,3	1	-0,757576	-1	-0,215686	-0,304348	Iris-versicolor	0,018736	0,981164	0,0001
X19	5,7	2,9	4,2	1,3	-0,333333	-0,2	0,1372549	-0,043478	Iris-versicolor	0,007627	0,99213	0,000243
X20	6,2	2,9	4,3	1,3	-0,030303	-0,2	0,1764706	-0,043478	Iris-versicolor	0,00647	0,993275	0,000255
X21	7,1	3	5,9	2,1	0,5151515	-0,066667	0,8039216	0,6521739	Iris-virginica	0,00009	0,000635	0,999275
X22	6,2	2,8	4,8	1,8	-0,030303	-0,333333	0,372549	0,3913043	Iris-virginica	0,000407	0,107874	0,891719
X23	7,9	3,8	6,4	2	1	1	1	0,5652174	Iris-virginica	0,000135	0,001629	0,998235
X24	6,3	2,8	5,1	1,5	0,030303	-0,333333	0,4901961	0,1304348	Iris-virginica	0,00122	0,914696	0,084083
X25	6	3	4,8	1,8	-0,151515	-0,066667	0,372549	0,3913043	Iris-virginica	0,000626	0,343308	0,656066
X26	6,9	3,1	5,4	2,1	0,3939394	0,0666667	0,6078431	0,6521739	Iris-virginica	0,000117	0,001165	0,998718
X27	6,7	3,1	5,6	2,4	0,2727273	0,0666667	0,6862745	0,9130435	Iris-virginica	0,000083	0,000545	0,999371
X28	6,7	3,3	5,7	2,5	0,2727273	0,3333333	0,7254902	1	Iris-virginica	0,000083	0,000529	0,999388
X29	6,7	3	5,2	2,3	0,2727273	-0,066667	0,5294118	0,826087	Iris-virginica	0,000096	0,000696	0,999208
X30	6,3	2,5	5	1,9	0,030303	-0,733333	0,4509804	0,4782609	Iris-virginica	0,000125	0,001598	0,998278

Lentelėje tikra reikšmė reiškia tikrąją gėlės reikšmę (tačiau WEKA šios informacijos nepanaudojo, ji skirta tik mums suprasti klasifikavimo tikslumą). Priskiriu spalvas gėlėms, kad būtų lengviau suprasti informaciją tolimesniame tyrime. Taip pat iš lentelės matome, kad X16, nors turėtų būti žalia (versicolor), yra priskirta raudonai (virginica) su labai didele, 89,5 % tikimybe. Priešingai priskirtas X24 objektas (vietoje virginica - versicolor) su 91,4 % tikimybe. Taip pat galime atkreipti dėmesį į X25, kur, nors ir gėlė buvo priskirta teisingai klasei, tačiau gana „neužtikrintai“ – 65,6 % tikimybe. Dabar iš WEKA nurašome svorius neuronams.

2 lentelė. Paskutinio sluoksnio neuronų svoriai

	Threshold	Node 3	Node 4	Node 5	Node 6	Node 7
Neuronai	w0	w1	w2	w3	w4	w5
Node 0	-0,46004	-5,00900	-1,93836	-2,42913	1,03119	4,52073
Node 1	-0,71707	4,76901	-0,09237	-11,60537	3,36924	-7,29546
Node 2	-3,18934	1,52096	1,18877	9,78636	-7,40802	-3,83918

3 lentelė. Paslėpto sluoksnio neuronų svoriai

	Threshold	Sep. Len	Sep. Wid	Pet. Len	Pet. Wid
Neuronai	w0	w1	w2	w3	w4
Node 3	2,10198	0,78622	-2,07917	2,85404	3,03306
Node 4	-0,52353	0,03716	-0,85810	1,25524	1,16635
Node 5	-7,33565	-1,51306	-3,91129	10,25556	9,24504
Node 6	3,85133	0,99114	1,89742	-5,43278	-4,91406
Node 7	-2,31444	-0,94251	2,35735	-3,27227	-3,34406

Paslėptuose neuronuose buvo skaičiuojamos sumos. Kiekviename langelyje buvo laikomasi tipinio neurono sumos skaičiavimo (svoriai sudauginta su įėjimais), tačiau svoriai atitinkamai keitėsi kiekviename stulpelyje (pagal node, t. y. WEKA priskirtus svorius kiekvienam neuronui (žr. 3 lentelę)). Įėjimai prieš tai buvo sunormaliuoti (žr. 1 lentelę).

4 lentelė. Sumos gautos paslėptuose neuronuose

		Sumos a				
k \ j		node 3	node 4	node 5	node 6	node 7
X1		-5,76824	-3,60949	-28,6422	14,90592	6,5738119
X2		-5,41493	-3,51728	-27,8306	14,48571	6,169324
X3		-5,92467	-3,62924	-28,0847	14,55139	6,7687029
X4		-5,3837	-3,41341	-26,7593	13,87109	6,1636015
X5		-5,1438	-3,31998	-27,2965	14,23986	5,873439
X6		-4,40909	-3,06341	-25,4699	13,29233	5,0492956
X7		-5,38999	-3,47686	-28,4953	14,88611	6,1447115
X8		-3,96176	-2,76991	-24,0397	12,56032	4,5538841
X9		-6,07649	-3,68144	-28,4864	14,76565	6,9311666
X10		-4,87525	-3,10467	-25,7033	13,37347	5,5875425
X11		4,149205	0,289453	-2,03892	1,130182	-4,641641
X12		1,721976	-0,43596	-7,7336	4,063325	-1,862793
X13		3,392143	-0,02555	-4,25996	2,297861	-3,802873
X14		2,925246	-0,24915	-5,39511	2,8627	-3,265507
X15		3,52181	0,114555	-2,5041	1,295878	-3,913394
X16		5,011539	0,740153	1,755899	-0,91571	-5,602754
X17		2,39944	-0,35609	-4,30771	2,133517	-2,632614
X18		2,046849	-0,31929	-7,3038	3,870404	-2,234228
X19		2,515602	-0,24272	-5,04337	2,609442	-2,775481
X20		2,865775	-0,18223	-5,09969	2,696736	-3,189414
X21		6,918126	1,322592	6,419694	-3,33694	-7,768686
X22		5,021336	0,685411	1,452291	-0,75805	-5,599287
X23		5,377413	0,570004	2,721026	-1,47041	-6,061985
X24		4,613521	0,531074	0,155376	-0,05521	-5,169018
X25		4,37159	0,452081	0,592681	-0,37221	-4,856417
X26		5,985988	0,957549	4,070696	-2,13883	-6,698508
X27		6,905768	1,355759	7,470205	-3,967	-7,713276
X28		6,72699	1,277579	7,63329	-4,10138	-7,503761
X29		6,471555	1,171851	5,579078	-2,94048	-7,223505
X30		6,388245	1,230775	4,53338	-2,31037	-7,146786

Gavus sumas buvo skaičiuojama sigmoidinė funkcija kiekviename langelyje pagal formulę

$$f(a) = \frac{1}{1 + e^{-a}}.$$

5 lentelė. Sigmoidinės funkcijos reikšmės paslėptuose neuronuose

k \ j	node 3	node 4	node 5	node 6	node 7
X1	0,003116	0,026352	3,64E-13	1	0,9986055
X2	0,00443	0,028825	8,19E-13	0,999999	0,9979117
X3	0,002666	0,02585	6,35E-13	1	0,9988521
X4	0,00457	0,031879	2,39E-12	0,999999	0,9978998
X5	0,005802	0,034892	1,4E-12	0,999999	0,9971947
X6	0,01202	0,044642	8,68E-12	0,999998	0,993627
X7	0,004541	0,029978	4,21E-13	1	0,9978598
X8	0,018674	0,058972	3,63E-11	0,999996	0,9895834
X9	0,002291	0,024568	4,25E-13	1	0,9990241
X10	0,007575	0,042915	6,87E-12	0,999998	0,9962697
X11	0,984468	0,571862	0,115177	0,755873	0,0095498
X12	0,848383	0,392705	0,000438	0,983099	0,1343778
X13	0,967458	0,493613	0,013926	0,9087	0,0218199
X14	0,94908	0,438032	0,004518	0,945971	0,0367736
X15	0,971302	0,528607	0,075571	0,78514	0,0195815
X16	0,993383	0,677029	0,852695	0,285832	0,0036741
X17	0,916785	0,411906	0,013285	0,894118	0,0670687
X18	0,885629	0,420848	0,000673	0,979576	0,0967186
X19	0,925228	0,439616	0,006411	0,931467	0,0586636
X20	0,946128	0,454567	0,006062	0,936834	0,0395661
X21	0,999011	0,789613	0,998373	0,034326	0,0004226
X22	0,993448	0,664945	0,810351	0,319069	0,0036869
X23	0,995401	0,638764	0,938256	0,18688	0,0023244
X24	0,990181	0,629734	0,538766	0,486202	0,005658
X25	0,987526	0,611134	0,64398	0,408006	0,0077183
X26	0,997493	0,722631	0,983221	0,105379	0,0012312
X27	0,998999	0,79507	0,999431	0,018578	0,0004467
X28	0,998803	0,782037	0,999516	0,01628	0,0005507
X29	0,998456	0,76348	0,996238	0,050188	0,0007287
X30	0,998322	0,773954	0,98937	0,090268	0,0007868

Tada buvo skaičiuojamos paskutinio sluoksnio neuronų sumos. Šį kartą svariai buvo imami atitinkamai iš 2 lentelės, o įvestys – iš 5 lentelės. Informacija buvo apdorojama eilutėmis. Tada kiekvienam stulpeliui buvo skaičiuojama sigmoidinė funkcija (funkcijos formulė identiška kaip anksčiau, tačiau  $a$  pas mus dabar jau yra iš 6 lentelės). Gauname tikimybės objektui būti priskirtam atitinkamai klasei (spalvų žymėjimas toks pats kaip anksčiau (*node 0* – setosa, *node 1* – versicolor, *node 2* - virginica)).

6 lentelė. Paskutinio sluoksnio neuronų gautos sumos

	node 0	node 1	node 2
X1	5,018884511	-4,620691443	-14,39510992
X2	5,004371819	-4,609590439	-14,38750698
X3	5,0232262	-4,624590782	-14,39733688
X4	4,997695977	-4,609119837	-14,383614
X5	4,98249859	-4,598378899	-14,37545412
X6	4,916321753	-4,543599322	-14,34070098
X7	5,001344358	-4,608786647	-14,38576871
X8	4,836932903	-4,483695293	-14,2980079
X9	5,02836581	-4,627513036	-14,40009197
X10	4,953879819	-4,583916107	-14,35966093
X11	-5,956885222	5,065420879	-5,521204612
X12	-3,850619479	5,619472834	-9,226573855
X13	-5,260991067	6,591975248	-7,810226175
X14	-4,932311748	6,63513208	-8,329840693
X15	-5,635353657	5,491690458	-6,235576164
X16	-8,508186811	-5,001761562	5,339620361
X17	-4,657712609	5,986057305	-8,056398908
X18	-4,266186054	6,054644239	-8,963491878
X19	-4,736503802	6,290705895	-8,322303958
X20	-4,950124973	6,550455278	-8,242597644
X21	-9,38252595	-7,499636188	8,783324682
X22	-8,34789362	-4,397026518	4,664688025
X23	-8,760101694	-6,305112544	6,872749205
X24	-7,422304016	-0,708781697	0,714341666
X25	-7,699855371	-2,219259353	2,28921994
X26	-9,131350595	-7,091342125	8,023628121
X27	-9,411739304	-7,565697337	8,916700078
X28	-9,387605165	-7,57492285	8,918372992
X29	-9,306158558	-7,423878927	8,611829002
X30	-9,267515188	-7,211161715	8,259727302

7 lentelė. Paskutinio sluoksnio neuronų sigmoidinės f-jos reikšmės

	node 0	node 1	node 2
X1	0,993432	0,00975	5,6E-07
X2	0,993336	0,009858	5,64E-07
X3	0,99346	0,009712	5,59E-07
X4	0,993292	0,009862	5,67E-07
X5	0,99319	0,009968	5,71E-07
X6	0,992727	0,010523	5,91E-07
X7	0,993316	0,009866	5,65E-07
X8	0,992131	0,011166	6,17E-07
X9	0,993493	0,009684	5,57E-07
X10	0,992993	0,010112	5,8E-07
X11	0,002581	0,993728	0,003985
X12	0,020824	0,996387	9,84E-05
X13	0,005163	0,998631	0,000405
X14	0,007158	0,998688	0,000241
X15	0,003557	0,995896	0,001955
X16	0,000202	0,006681	0,995225
X17	0,009399	0,997493	0,000317
X18	0,013841	0,997659	0,000128
X19	0,008693	0,99815	0,000243
X20	0,007033	0,998573	0,000263
X21	8,42E-05	0,000553	0,999847
X22	0,000237	0,012164	0,990666
X23	0,000157	0,001824	0,998965
X24	0,000597	0,329868	0,67136
X25	0,000453	0,098034	0,90798
X26	0,000108	0,000832	0,999672
X27	8,18E-05	0,000518	0,999866
X28	8,37E-05	0,000513	0,999866
X29	9,09E-05	0,000596	0,999818
X30	9,44E-05	0,000738	0,999741

Didžiausi pasikeitimai skaičiuojant Excel nuo WEKA buvo tai, jog su Excel gavome tik vieną klaidingai suklasifikuotą gėlę (X24 dabar priskiriama teisingai klasei su tikimybe 67,1 %). Labai keistas pastebėjimas, nes nesitikėjau tokio reikšmingo pokyčio. Taip pat Excel skaičiuotos tikimybės virur buvo „užtikrintesnės“ – tikimybės buvo arčiau 1. Mano anksčiau dėmesį patraukusi reikšmė (1 lentelė X25) vietoje 65,6 % įgijo net 90,7 % tikimybę būti priskirtai virginica klasei.

## Išvados

Tyrimas parodė, kad darbas su programomis, skirtomis apmokyti neuroninius tinklus, yra žymiai paprastesnis nei programavimas visko nuo nulio. Programa veikė greičiau, buvo galima

lengviau patikrinti klasifikavimo tikslumą naudojant kryžminę patikrą. Patys neuroniniai tinklai mums leido klasifikuoti duomenis jau nebe į dvi, o į kiek norim klasių (mūsų atveju 3). Taip pat gavome panašius į darbo su vienu neuronu (2 lab. darbo) rezultatus, kur virginica ir versicolor nulėmė visą dalį klaidų. Iš grafikų matyti, kad daug jų parametrų persipina, dėl to neuroniniui tinklui žymiai sunkiau atskirti ribinius atvejus.

## Priedai

### 4 užduoties testinio failo klasifikavimo rezultatai

5.4,3.7,1.5,0.2,Iris-setosa,Iris-setosa

5.7,3.8,1.7,0.3,Iris-setosa,Iris-setosa

5.1,3.8,1.5,0.3,Iris-setosa,Iris-setosa

5.1,3.7,1.5,0.4,Iris-setosa,Iris-setosa

5.2,3.4,1.4,0.2,Iris-setosa,Iris-setosa

5.4,3.4,1.5,0.4,Iris-setosa,Iris-setosa

5.5,3.5,1.3,0.2,Iris-setosa,Iris-setosa

4.8,3,1.4,0.3,Iris-setosa,Iris-setosa

5.1,3.8,1.6,0.2,Iris-setosa,Iris-setosa

4.6,3.2,1.4,0.2,Iris-setosa,Iris-setosa

6.5,2.8,4.6,1.5,Iris-versicolor,Iris-versicolor

4.9,2.4,3.3,1,Iris-versicolor,Iris-versicolor

6.6,2.9,4.6,1.3,Iris-versicolor,Iris-versicolor

6.7,3.1,4.4,1.4,Iris-versicolor,Iris-versicolor

6,2.9,4.5,1.5,Iris-versicolor,Iris-versicolor

6,2.7,5.1,1.6,Iris-versicolor,Iris-virginica

6,3.4,4.5,1.6,Iris-versicolor,Iris-versicolor

5,2.3,3.3,1,Iris-versicolor,Iris-versicolor

5.7,2.9,4.2,1.3,Iris-versicolor,Iris-versicolor

6.2,2.9,4.3,1.3,Iris-versicolor,Iris-versicolor

7.1,3,5.9,2.1,Iris-virginica,Iris-virginica  
6.2,2.8,4.8,1.8,Iris-virginica,Iris-virginica  
7.9,3.8,6.4,2,Iris-virginica,Iris-virginica  
6.3,2.8,5.1,1.5,Iris-virginica,Iris-versicolor  
6,3,4.8,1.8,Iris-virginica,Iris-versicolor  
6.9,3.1,5.4,2.1,Iris-virginica,Iris-virginica  
6.7,3.1,5.6,2.4,Iris-virginica,Iris-virginica  
6.7,3.3,5.7,2.5,Iris-virginica,Iris-virginica  
6.7,3,5.2,2.3,Iris-virginica,Iris-virginica  
6.3,2.5,5,1.9,Iris-virginica,Iris-virginica