

# Vieno neurono (perceptrono) mokymas sprendžiant klasifikavimo uždavinį

Pijus Petkevičius

2023 m. lapkričio 27 d.

# Turinys

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Įvadas</b>   | <b>3</b>  |
| 1.1      | Tikslas . . . . .   | 3         |
| 1.2      | Uždaviniai . . . . .  | 3         |
| 1.3      | Sąvokos . . . . .   | 3         |
| <b>2</b> | <b>Eksperimentų vykdymas</b>  | <b>3</b>  |
| 2.1      | Duomenų paruošimas . . . . .  | 3         |
| 2.1.1    | Irisų duomenų aibė . . . . .  | 3         |
| 2.1.2    | Krūties vėžio duomenų aibė . . . . .                                | 4         |
| 2.2      | Duomenų ir aktyvacijos funkcijos grupės . . . . .                   | 4         |
| 2.3      | Parametrai . . . . .  | 4         |
| 2.4      | Mokymo ir testavimo duomenų aibės . . . . .                         | 5         |
| <b>3</b> | <b>Programos kodas</b>  | <b>5</b>  |
| <b>4</b> | <b>Rezultatai</b>   | <b>9</b>  |
| 4.1      | Klasifikavimo tikslumas ir santykinė paklaida . . . . .             | 9         |
| 4.2      | Mokymosi greitis . . . . .  | 10        |
| 4.3      | Aktyvacijos funkcija . . . . .                                      | 11        |
| 4.4      | Tiksliausi klasifikavimo rezultatai ir mažiausia paklaida . . . . . | 13        |
| 4.4.1    | Irisų duomenų aibė . . . . .  | 13        |
| 4.4.2    | Krūties vėžio duomenų aibė . . . . .                                | 15        |
| <b>5</b> | <b>Išvados</b>  | <b>19</b> |

# 1 Įvadas

## 1.1 Tikslas

Užduoties tikslas – apmokyti vieną neuroną (perceptroną) spręsti dviejų klasių uždavinį, ištestuoti apmokyto neurono klasifikavimo tikslumą ir paklaidą, atlikti tyrimą su dviem duomenų aibėmis.

## 1.2 Uždaviniai

- Parsisiųsti ir paruošti duomenis krūties vėžio ir irisų duomenų aibėms.
- Sukurti programą, kuri įgyvendintų neurono mokymą ir testavimo procesus.
- Iširti neurono klasifikavimo tikslumo priklausomybę nuo epochų skaičiaus.
- Iširti, kaip paklaidos reikšmės priklauso nuo epochų skaičiaus.
- Iširti, kaip klasifikavimo rezultatai priklauso nuo skirtingų mokymosi greičių.
- Iširti, kaip rezultatai priklauso nuo aktyvacijos funkcijos (slenkstinės, sigmoidinės).
- Rasti tokį mokymosi greitį, epochų skaičių, aktyvacijos funkciją, svorius, kad gauti klasifikacijos rezultatai būtų tiksliausi.

## 1.3 Sąvokos

- **Epocha** yra dirbtinių neuronų mokymo dalis, kai visas įėjimo vektorių rinkinys pereinamas vieną kartą.
- **Iteracija** - vienas įėjimo vektorių rinkinys pereinamas vieną kartą. vektoriaus apdorojimas.

# 2 Eksperimentų vykdymas

## 2.1 Duomenų paruošimas

Iš gautų duomenų buvo pašalinti nereikalingi stulpeliai (Krūties vėžio atveju - ID) pervadinti klasių pavadinimai į 0 arba 1.

### 2.1.1 Irisų duomenų aibė

Irisų duomenų aibė turi 3 klases: Setosa, Versicolor ir Virginica. Tyrimui buvo panaudoti tik duomenys, turintys Versicolor ir Virginica klases.

Irisų duomenų aibėje:

- 100 įrašų
- 4 požymiai
- Versicolor pervadinta į 0, Virginica - 1
- 50 pirmos(1) klasės duomenų
- 50 0 klasės duomenų

### 2.1.2 Krūties vėžio duomenų aibė

Krūties vėžio duomenų aibėje:

- 569 įrašai
- 30 požymių
- B(Benign) buvo pervadinta į 0, o M(Malignant) - 1
- 357 gerybiniai(B) augliai
- 212 piktybiniai(M) augliai

## 2.2 Duomenų ir aktyvacijos funkcijos grupės

Eksperimentai buvo vykdomi skirtingoms aktyvacijos funkcijoms ir duomenų aibėms:

- Irisų duomenų aibė, slenkstinė aktyvacijos funkcija
- Irisų duomenų aibė, sigmoidinė aktyvacijos funkcija
- Krūties vėžio duomenų aibė, slenkstinė aktyvacijos funkcija
- Krūties vėžio duomenų aibė, sigmoidinė aktyvacijos funkcija

## 2.3 Parametrai

- epochų skaičius - 40
- mokymosi greitis - 0.9
- Paklaidos minimizavimui perceptrono mokymo metu naudojama ADALINE mokymo taisyklė.
- Prieš perceptrono mokymą buvo atstiktinai parinktos svorių reikšmės intervale (0; 1).
- Neuronų mokymas ir jo testavimas buvo vykdomas 20 kartų, kiekvieną kartą sugeneruojant naujas svorių reikšmes. Rezultatai paskaičiuojami kiekvienos reikšmės (vidurkis, santykinė paklaida) aritmetiniu vidurkiu, kuris pavaizduojamas kiekvienoje lentelėje ir grafikuose.

## 2.4 Mokymo ir testavimo duomenų aibės

Programoje, prieš pradedant neurono mokymą, duomenų aibės eilutės sumaišomos, kad panašios klasės duomenų eilutės rečiau pasitaikytų. Duomenų aibės padalinamos į treniravimo ir testavimo aibes santykiu 70:30.

## 3 Programos kodas

```
1 import random
2
3 # Krūties vėžio duomenų aibės klasės pervadinamos į skaičius, kad
4   ↳ būtų galima apdoroti M - 1, B - 0,
5 # po to duomenys sumaišomi, pirmas stulpelis(Id) išmetamas,
6   ↳ klasės reiksmė išskeliama į atskirą masyvą,
7 # prie įvesties pridedama 1(poslinkio reikšmę norint traktuoti
8   ↳ kaip svorį).
9 def prepareBreastCancerData(inputFile: str) -> (np.ndarray,
10   ↳ [int]):
11     outputFile= inputFile.replace('.data', '2.csv')
12     with open(inputFile, 'r') as fileInput:
13         with open(outputFile, 'w') as file:
14             for row in fileInput:
15                 modifiedString = row.replace('M',
16                 ↳ '1').replace('B', '0')
17                 file.write(modifiedString)
18     inputData = readDataFromFile(outputFile, 0)
19     np.random.shuffle(inputData)
20     inputData = np.delete(inputData, 0, axis=1)
21     output = inputData[:, 0]
22     inputData = np.delete(inputData, 0, axis=1)
23     inputData = np.concatenate((np.array([1 for _ in
24     ↳ range(inputData.shape[0])])[:, np.newaxis], inputData)
25     ↳ axis=1)
26     return inputData, output
27
28 # Irisų duomenų aibėje išmetamos Iris-setosa klasės reikšmės, kad
29   ↳ būtų galima apdoroti Iris-versicolor, Iris-virginica, kaip 0
30   ↳ ir 1,
31 # po to duomenys sumaišomi, paskutinis stulpelis(klasės reikšmė)
32   ↳ perkeliama į atskirą masyvą ir išmetama.
33 def prepareIrisData(inputFile: str) -> (np.ndarray, [int]):
34     outputFile= inputFile.replace('.data', '2.csv')
35     with open(inputFile, 'r') as fileInput:
36         with open(outputFile, 'w') as file:
37             for row in fileInput:
```

```

28         if 'Iris-setosa' in row:
29             continue
30         modifiedString = row.replace('Iris-versicolor',
31                                     ↪ '0').replace('Iris-virginica', '1')
32         file.write(modifiedString)
33     inputData = readDataFromFile(outputFile, 0)
34     np.random.shuffle(inputData)
35     output = inputData[:, -1]
36     inputData = np.delete(inputData, -1, axis=1)
37     inputData = np.concatenate((np.array([1 for _ in
38     ↪ range(inputData.shape[0])])[:, np.newaxis], inputData)
39     ↪ axis=1)
40     return inputData, output
41
42 # Paimamos įvesties ir svorių reikšmės, sudauginama i-oji
43 ↪ įvesties reikšmė su i-ojo svorio reikšme ir taip su visomis
44 ↪ įvesties reikšmėmis, jos susumuojamos ir grąžinama reikšmė
45 # (arba galima panaudoti scalarinių sandaugą)
46 def countA(inputValues: [float], weights: [float] ) -> float:
47     return np.dot(inputValues, weights)
48
49 # Apskaičiuojama slenkstinės funkcijos reikšmė
50 def threshold(x) -> int:
51     return 1 if x >= 0 else 0
52
53 # Apskaičiuojama sigmoidinės funkcijos reikšmė
54 def sigmoid(x) -> float:
55     return 1 / (1 + np.exp(-x))
56
57 # Sugeneruojami svoriai intervale [0, 1] su 2 skaičių po kablelio
58 def generateWeights(count: int) -> [float]:
59     return [ round(random.uniform(0, 1) 2) for _ in range(count)]
60
61 # Apmokomas neuromas su viena duomenų aibės įvestimi ir išvestimi
62 def trainOneIteration(weights: [float], inputRow: [float],
63     ↪ expectedValue: float, learningRate: float,
64     ↪ activationFunction: callable) -> [float]:
65     y = activationFunction(countA(inputRow, weights))
66     error = round(expectedValue) - y
67
68     if error != 0:
69         for i in range(len(weights)):
70             weights[i] = weights[i] + learningRate *
71             ↪ (expectedValue-y) * inputRow[i]
72     return weights

```

```

66 # Apmokomas neuromas su viena duomenų aibės įvestimi ir išvestimi
67 def trainOneEpoch(weights: [float], input: np.ndarray, output:
    ↳ [float], learningRate: float, activationFunction: callable)
    ↳ -> [float]:
68     for i in range(input.shape[0]):
69         weights = trainOneIteration(weights, input[i], output[i],
    ↳ learningRate, activationFunction)
70     return weights
71
72 # Apmokomas neuromas su viena duomenų aibės įvestimi ir išvestimi
    ↳ epochų kartų skaičių
73 def trainModel(weights: [float], input: np.ndarray, output:
    ↳ [float], learningRate: float, activationFunction: callable,
    ↳ epochs: int) -> [float]:
74     for _ in range(epochs):
75         weights = trainOneEpoch(weights, input, output,
    ↳ learningRate, activationFunction)
76     return weights
77
78 # Testuojamas apmokytas modelis su testiniais duomenimis,
    ↳ grąžinama tikslumo reikšmė ir paklaida
79 def testModel(weights: [float], input: np.ndarray, output:
    ↳ [float], activationFunction: callable) -> (float, float): #
    ↳ returns accuracy on test data
80     countMatches = 0
81     error = 0.0
82     for i in range(input.shape[0]):
83         predictedOutput = activationFunction(countA(input[i],
    ↳ weights))
84         error += pow(predictedOutput - output[i], 2) #  $E(W) =$ 
    ↳  $\sum (t - y)^2$ 
85         if round(predictedOutput) == output[i]:
86             countMatches += 1
87     return countMatches / input.shape[0], error / input.shape[0]
88
89 # Apmokomas ir testuojamas modelis su duomenimis iš failo su
    ↳ slenkstine ir sigmoidine funkcijomis
90 def runTrainAndTest(fileName: str, prepareData: callable):
91     learningPercentace = 0.7
92     learningRate = 0.9
93     epochs = 20
94     input, output = prepareData(fileName)
95     weights = generateWeights(input.shape[1])
96     splitPart = round(input.shape[0]*learningPercentace)
97     trainInput = input[:splitPart]
98     trainOutput = output[:splitPart]

```

```

99     testInput = input[splitPart:]
100     testOutput = output[splitPart:]
101
102     print(fileName.replace('.data', ''))
103     weights = trainModel(weights, trainInput, trainOutput,
104         ↪ learningRate, threshold, epochs)
104     accuracy = testModel(weights, testInput, testOutput,
105         ↪ threshold)
105     print(accuracy)
106
107     weights = generateWeights(input.shape[1])
108     weights = trainModel(weights, trainInput, trainOutput,
109         ↪ learningRate, sigmoid, epochs)
109     accuracy = testModel(weights, testInput, testOutput, sigmoid)
110     print(accuracy)
111
112 runTrainAndTest("wdbc.data", prepareBreastCancerData)
113 runTrainAndTest("iris.data", prepareIrisData)

```

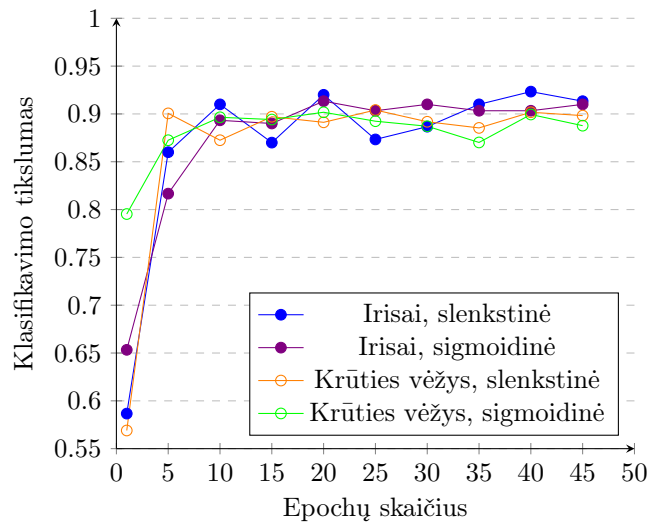


## 4 Rezultatai

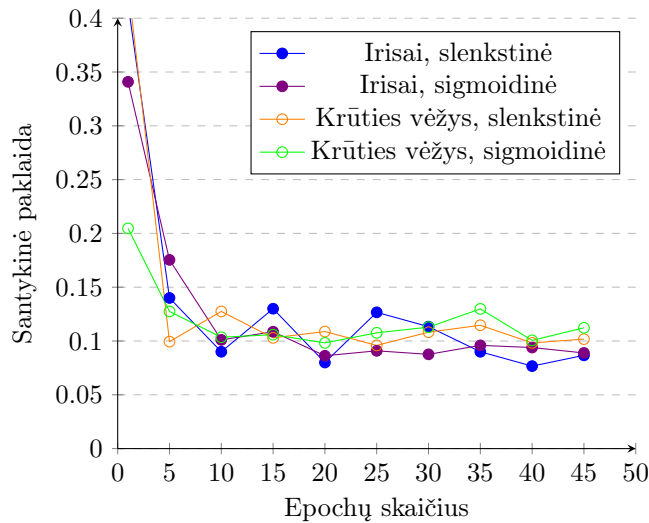
### 4.1 Klasifikavimo tikslumas ir santykinė paklaida

1 pav. ir 2 pav. galime matyti, kad su krūties vėžio duomenų rinkiniu klasifikavimo tikslumas ir santykinė paklaida beveik nebekinta nuo 10 epochos (oranžinė ir žalia linijos) todėl, kad pasiektume norimą tikslumą su krūties vėžio duomenų aibe, pakanka 10 epochų. Aktyvacijos funkcijos parinkimas beveik neturi įtakos klasifikavimo tikslumui.

Irisų duomenų rinkinio klasifikavimo tikslumas yra tikslesnis su slenkstine aktyvacijos funkcija. Geriausias rezultatas yra pasiekiamas 40 epochų metu.



1 pav.: Klasifikavimo tikslumo priklausomybė nuo epochų skaičiaus



2 pav.: Klasifikavimo paklaidos priklausomybė nuo epochų skaičiaus

## 4.2 Mokymosi greitis

1 lentelėje paryškinti rezultatai yra geriausi tos grupės rezultatai. Su krūtys vėžio duomenų rinkiniu, slenkstine funkcija skirtingi mokymosi greičiai daro labai mažus pokyčius, kurie siekia vos 0,005, todėl mokymosi greičio pasirinkimas nedaro didelės įtakos šios grupės klasifikavimo tikslumui ir paklaidai. Tam pačiam duomenų rinkiniui naudojant sigmoidinę funkciją skirtingi mokymosi greičiai gali keisti klasifikavimo tikslumą ir paklaidą iki 0,005, tad pasiekiamas gana mažas skirtumas.

Didžiausi skirtumai tarp mokymosi greičio pasirinkimų pasiekti su irisų duomenų aibe, slenkstine funkcija, kur su  $\eta = 0,7$  galima gauti iki 0,033 didesnę klasifikavimo tikslumą ir mažesnę paklaidą. Geriausiai šis duomenų rinkinys klasifikuojamas su slenkstine aktyvacijos funkcija,  $\eta = 0,3$ .

1 lentelė: Klasifikavimo tikslumas ir paklaida su skirtingais mokymosi greičiais

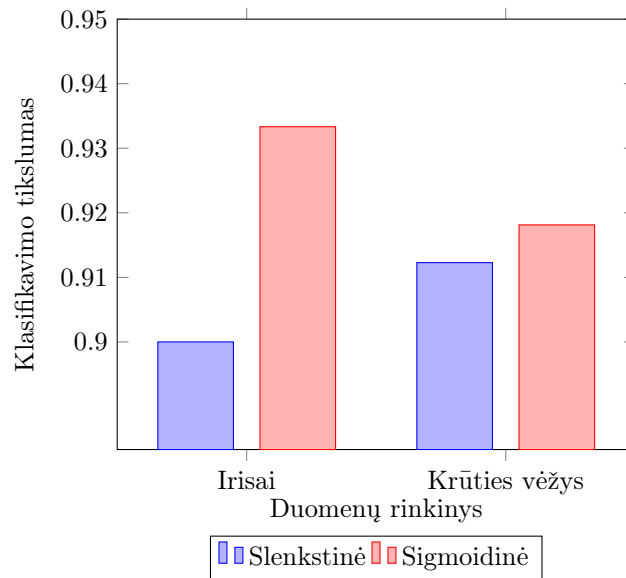
| Duomenų aibė  | Aktyvacijos funkcija | Mokymosi greitis | Klasifikavimo tikslumas | Santykinė paklaida |
|---------------|----------------------|------------------|-------------------------|--------------------|
| Irisai        | Slenkstinė           | 0.1              | 0.8033                  | 0.1967             |
|               |                      | 0.2              | 0.89                    | 0.11               |
|               |                      | <b>0.3</b>       | <b>0.9467</b>           | <b>0.0533</b>      |
|               |                      | 0.4              | 0.91                    | 0.09               |
|               |                      | 0.5              | 0.8633                  | 0.1367             |
|               |                      | 0.6              | 0.92                    | 0.08               |
|               |                      | 0.7              | 0.9                     | 0.1                |
|               |                      | 0.8              | 0.93                    | 0.07               |
|               |                      | 0.9              | 0.86                    | 0.14               |
| Irisai        | Sigmoidinė           | 0.1              | 0.9267                  | 0.0619             |
|               |                      | 0.2              | 0.9233                  | 0.0667             |
|               |                      | 0.3              | 0.8967                  | 0.0942             |
|               |                      | 0.4              | 0.8733                  | 0.1141             |
|               |                      | 0.5              | 0.9233                  | 0.0721             |
|               |                      | 0.6              | 0.9067                  | 0.0924             |
|               |                      | <b>0.7</b>       | <b>0.9167</b>           | <b>0.0797</b>      |
|               |                      | 0.8              | 0.87                    | 0.1235             |
|               |                      | 0.9              | 0.9                     | 0.0952             |
| Krūties vėžys | Slenkstinė           | 0.1              | 0.8959                  | 0.1041             |
|               |                      | 0.2              | 0.8912                  | 0.1088             |
|               |                      | 0.3              | 0.8795                  | 0.1205             |
|               |                      | 0.4              | 0.8795                  | 0.1205             |
|               |                      | 0.5              | 0.8953                  | 0.1047             |
|               |                      | 0.6              | 0.8901                  | 0.1099             |
|               |                      | 0.7              | 0.9018                  | 0.0982             |
|               |                      | 0.8              | 0.8982                  | 0.1018             |
|               |                      | <b>0.9</b>       | <b>0.9035</b>           | <b>0.0965</b>      |
| Krūties vėžys | Sigmoidinė           | 0.1              | 0.8918                  | 0.1082             |
|               |                      | 0.2              | 0.8977                  | 0.1023             |
|               |                      | <b>0.3</b>       | <b>0.9082</b>           | <b>0.0918</b>      |
|               |                      | 0.4              | 0.8784                  | 0.1216             |
|               |                      | 0.5              | 0.8947                  | 0.1053             |
|               |                      | 0.6              | 0.9053                  | 0.0947             |
|               |                      | 0.7              | 0.9012                  | 0.0988             |
|               |                      | 0.8              | 0.8889                  | 0.1111             |
|               |                      | 0.9              | 0.8959                  | 0.1041             |

### 4.3 Aktyvacijos funkcija

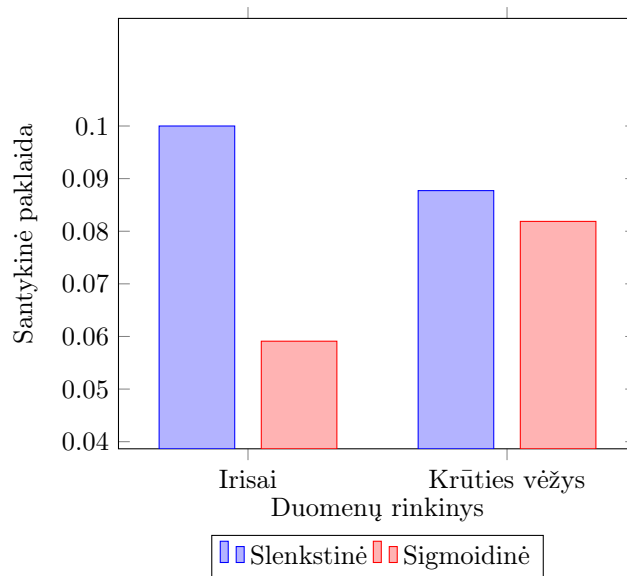
Iš 3 pav. ir 4 pav. galima matyti, kad su krūties vėžio duomenų rinkiniu aktyvacijos funkcijos pasirinkimas nesukuria reikšmingo pokyčio klasifikavimo

tikslumo ir santykinės paklaidos atžvilgiu, su sigmoidine funkcija klasifikavimo tikslumas iki 0,006 didesnis (paklaida iki 0,006 mažesnė).

Irisų duomenų rinkiniui galima pasiekti šiek tiek didesnę (iki 0,033) klasifikavimo tikslumą ir mažesnę paklaidą su sigmoidine aktyvacijos funkcija.



3 pav.: Klasifikavimo tikslumas su skirtingomis aktyvacijos funkcijomis



4 pav.: Santykinė paklaida su skirtingomis aktyvacijos funkcijomis

## 4.4 Tiksliausi klasifikavimo rezultatai ir mažiausia paklaida

### 4.4.1 Irisų duomenų aibė

Didžiausią klasifikavimo tikslumą (0,933) ir mažiausią paklaidą (0,066) irisų duomenų rinkinyje galima gauti su tokiais parametrais:

- epochų skaičius - 40;
- mokymosi greitis  $\eta = 0,3$ ;
- slenkstine aktyvacijos funkcija;
- svorių rinkiniu  $w_0 = -9.5$ ,  $w_1 = -14.14$ ,  $w_2 = -7.58$ ,  $w_3 = 17.27$ ,  $w_4 = 25.28$ .

2 lentelėje galima matyti vieną iš bandymų su šiais parametrais, kur buvo priskirta kita klasė nei trokštama 1/10 duomenų (eilutė paryškinta).

2 lentelė: Irisų duomenų testinės aibės norimos ir gautos klasifikavimo reikšmės

| Nr.       | Trokštama reikšmė | Gauta reikšmė |
|-----------|-------------------|---------------|
| <b>1</b>  | <b>1</b>          | <b>0</b>      |
| 2         | 1                 | 1             |
| 3         | 1                 | 1             |
| 4         | 1                 | 1             |
| 5         | 0                 | 0             |
| <b>6</b>  | <b>0</b>          | <b>1</b>      |
| 7         | 0                 | 0             |
| 8         | 0                 | 0             |
| 9         | 1                 | 1             |
| 10        | 1                 | 1             |
| 11        | 0                 | 0             |
| 12        | 1                 | 1             |
| 13        | 0                 | 0             |
| 14        | 1                 | 1             |
| 15        | 1                 | 1             |
| 16        | 1                 | 1             |
| 17        | 1                 | 1             |
| 18        | 1                 | 1             |
| 19        | 1                 | 1             |
| 20        | 1                 | 1             |
| 21        | 1                 | 1             |
| 22        | 0                 | 0             |
| 23        | 0                 | 0             |
| 24        | 1                 | 1             |
| <b>25</b> | <b>0</b>          | <b>1</b>      |
| 26        | 0                 | 0             |
| 27        | 0                 | 0             |
| 28        | 0                 | 0             |
| 29        | 0                 | 0             |
| 30        | 1                 | 1             |

#### 4.4.2 Krūties vėžio duomenų aibė

Didžiausią klasifikavimo tikslumą (0,91) ir mažiausią paklaidą (0,088) krūties vėžio duomenų rinkinyje galima gauti su tokiais parametrais:

- epochų skaičius - 20;
- mokymosi greitis  $\eta = 0,3$ ;
- sigmoidine aktyvacijos funkcija;
- svorių rinkiniu  $w_0 = -90,56$ ,  $w_1 = -686,35$ ,  $w_2 = -819,58$ ,  $w_3 = -3962,23$ ,  $w_4 = -1347,75$ ,  $w_5 = -5,82$ ,  $w_6 = 3,54$ ,  $w_7 = 13,73$ ,  $w_8 = 6,54$ ,  $w_9 = -11,31$ ,  $w_{10} = -4,31$ ,  $w_{11} = 4,53$ ,  $w_{12} = -37,72$ ,  $w_{13} = 69,48$ ,  $w_{14} = 2007,86$ ,  $w_{15} = 0,46$ ,  $w_{16} = 0,87$ ,  $w_{17} = 1,27$ ,  $w_{18} = 0,62$ ,  $w_{19} = -0,03$ ,  $w_{20} = 0,53$ ,  $w_{21} = -709,34$ ,  $w_{22} = -1024,96$ ,  $w_{23} = -3888,92$ ,  $w_{24} = 1889,39$ ,  $w_{25} = -6,70$ ,  $w_{26} = 8,7$ ,  $w_{27} = 22,3$ ,  $w_{28} = 7,35$ ,  $w_{29} = -12,46$ ,  $w_{30} = -4,75$ .

[3 lentelėje](#) galima matyti vieną iš bandymų su šiais parametrais, kur buvo priskirta kita klasė nei trokštama 17/171 duomenų (eilutė paryškinta).

3 lentelė: Krūties vėžio duomenų testinės aibės norimos ir gautos klasifikavimo reikšmės

| Nr.       | Trokštama reikšmė | Gauta reikšmė |
|-----------|-------------------|---------------|
| 1         | 1                 | 1             |
| 2         | 1                 | 1             |
| <b>3</b>  | <b>1</b>          | <b>0</b>      |
| 4         | 0                 | 0             |
| 5         | 0                 | 0             |
| 6         | 0                 | 0             |
| <b>7</b>  | <b>1</b>          | <b>0</b>      |
| 8         | 0                 | 0             |
| 9         | 0                 | 0             |
| 10        | 0                 | 0             |
| 11        | 1                 | 1             |
| 12        | 0                 | 0             |
| 13        | 0                 | 0             |
| 14        | 0                 | 0             |
| 15        | 0                 | 0             |
| 16        | 0                 | 0             |
| 17        | 0                 | 0             |
| 18        | 1                 | 1             |
| 19        | 0                 | 0             |
| 20        | 0                 | 0             |
| <b>21</b> | <b>1</b>          | <b>0</b>      |
| 22        | 1                 | 1             |
| 23        | 0                 | 0             |
| 24        | 0                 | 0             |
| 25        | 0                 | 0             |
| 26        | 1                 | 1             |
| <b>27</b> | <b>1</b>          | <b>0</b>      |
| 28        | 0                 | 0             |
| 29        | 0                 | 0             |
| 30        | 0                 | 0             |
| 31        | 1                 | 1             |
| 32        | 0                 | 0             |
| 33        | 0                 | 0             |
| 34        | 0                 | 0             |
| 35        | 0                 | 0             |
| 36        | 0                 | 0             |
| 37        | 1                 | 1             |
| <b>38</b> | <b>0</b>          | <b>1</b>      |
| 39        | 1                 | 1             |
| 40        | 0                 | 0             |
| 41        | 1                 | 1             |
| 42        | 0                 | 0             |



|           |          |          |
|-----------|----------|----------|
| 43        | 1        | 1        |
| 44        | 0        | 0        |
| 45        | 1        | 1        |
| 46        | 0        | 0        |
| 47        | 0        | 0        |
| 48        | 1        | 1        |
| 49        | 1        | 1        |
| 50        | 1        | 1        |
| 51        | 1        | 1        |
| 52        | 1        | 1        |
| <b>53</b> | <b>1</b> | <b>0</b> |
| 54        | 0        | 0        |
| 55        | 1        | 1        |
| 56        | 0        | 0        |
| 57        | 0        | 0        |
| 58        | 0        | 0        |
| 59        | 1        | 1        |
| 60        | 1        | 1        |
| 61        | 0        | 0        |
| 62        | 1        | 1        |
| 63        | 0        | 0        |
| 64        | 0        | 0        |
| 65        | 0        | 0        |
| 66        | 0        | 0        |
| <b>67</b> | <b>1</b> | <b>0</b> |
| 68        | 1        | 1        |
| 69        | 0        | 0        |
| 70        | 0        | 0        |
| 71        | 0        | 0        |
| 72        | 0        | 0        |
| 73        | 0        | 0        |
| 74        | 1        | 1        |
| 75        | 0        | 0        |
| <b>76</b> | <b>1</b> | <b>0</b> |
| 77        | 0        | 0        |
| 78        | 1        | 1        |
| 79        | 1        | 1        |
| 80        | 1        | 1        |
| 81        | 0        | 0        |
| 82        | 0        | 0        |
| 83        | 1        | 1        |
| 84        | 0        | 0        |
| 85        | 1        | 1        |
| 86        | 1        | 1        |
| 87        | 0        | 0        |
| 88        | 1        | 1        |

|            |          |          |
|------------|----------|----------|
| 89         | 1        | 1        |
| 90         | 1        | 1        |
| 91         | 0        | 0        |
| 92         | 1        | 1        |
| 93         | 0        | 0        |
| 94         | 1        | 1        |
| 95         | 0        | 0        |
| 96         | 0        | 0        |
| 97         | 0        | 0        |
| 98         | 0        | 0        |
| 99         | 0        | 0        |
| <b>100</b> | <b>1</b> | <b>0</b> |
| 101        | 0        | 0        |
| 102        | 0        | 0        |
| 103        | 1        | 1        |
| <b>104</b> | <b>1</b> | <b>0</b> |
| 105        | 0        | 0        |
| 106        | 1        | 1        |
| 107        | 1        | 1        |
| 108        | 0        | 0        |
| 109        | 1        | 1        |
| 110        | 0        | 0        |
| 111        | 0        | 0        |
| 112        | 0        | 0        |
| 113        | 0        | 0        |
| 114        | 1        | 1        |
| <b>115</b> | <b>1</b> | <b>0</b> |
| 116        | 1        | 1        |
| 117        | 0        | 0        |
| 118        | 0        | 0        |
| 119        | 1        | 1        |
| 120        | 0        | 0        |
| 121        | 0        | 0        |
| 122        | 1        | 1        |
| 123        | 0        | 0        |
| <b>124</b> | <b>1</b> | <b>0</b> |
| 125        | 0        | 0        |
| 126        | 1        | 1        |
| 127        | 0        | 0        |
| 128        | 1        | 1        |
| 129        | 1        | 1        |
| 130        | 0        | 0        |
| 131        | 0        | 0        |
| 132        | 1        | 1        |
| 133        | 1        | 1        |
| 134        | 0        | 0        |

|            |          |          |
|------------|----------|----------|
| <b>135</b> | <b>1</b> | <b>0</b> |
| <b>136</b> | <b>1</b> | <b>0</b> |
| 137        | 0        | 0        |
| 138        | 1        | 1        |
| <b>139</b> | <b>1</b> | <b>0</b> |
| 140        | 1        | 1        |
| 141        | 0        | 0        |
| 142        | 0        | 0        |
| 143        | 1        | 1        |
| 144        | 0        | 0        |
| 145        | 0        | 0        |
| 146        | 1        | 1        |
| 147        | 0        | 0        |
| 148        | 1        | 1        |
| <b>149</b> | <b>0</b> | <b>1</b> |
| 150        | 0        | 0        |
| 151        | 0        | 0        |
| 152        | 0        | 0        |
| 153        | 1        | 1        |
| 154        | 0        | 0        |
| 155        | 1        | 1        |
| 156        | 1        | 1        |
| 157        | 0        | 0        |
| 158        | 0        | 0        |
| 159        | 1        | 1        |
| 160        | 0        | 0        |
| 161        | 1        | 1        |
| 162        | 0        | 0        |
| 163        | 1        | 1        |
| 164        | 0        | 0        |
| 165        | 0        | 0        |
| 166        | 0        | 0        |
| 167        | 1        | 1        |
| 168        | 0        | 0        |
| <b>169</b> | <b>1</b> | <b>0</b> |
| 170        | 0        | 0        |
| 171        | 0        | 0        |

## 5 Išvados

- Norint pasiekti geriausius klasifikavimo rezultatus su krūties vėžio duomenimis bei vienu neuronu rekomenduojamas epochų skaičius yra didesnis nei 10, o aktyvacijos funkcijos ir mokymosi greičio parinkimas nesukelia didelių rezultatų pokyčių.

- Norint pasiekti geriausias klasifikavimo rezultatus su irisų duomenimis bei vienu neuronu rekomenduojama vykdyti 40 apmokymo epochų, parinkti 0,3 mokymosi greitį ir naudoti slenkstinę aktyvacijos funkciją.