

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №1
за III семестр
по дисциплине: "Метады и алгоритмы принятия решений"
Тема: "Линейная искусственная нейронная сеть. Правило Видроу-Хоффа"

Выполнил:
студент 2 курса
группы ПО-4 (1)
Галанин П. И.

Проверил:
ст. преподаватель
Крощенко А. А.

Лабораторная работа №1

Тема: "Линейная искусственная нейронная сеть. Правило Видроу-Хоффа".

Цель: "Изучить обучение и функционирование линейной ИНС при решении задач прогнозирования".

Ход работы:

Вариант 5

Листинг: "main.py"

```
1 import math
2 import random
3
4 def print_headTable():
5     print("| %20s | %20s | %20s | %20s |" % (
6         "Итерация",
7         "Эталонное значение",
8         "Полученное значение",
9         "Отклонение"
10    ))
11    print("| %16s | %16s | %16s | %16s |" % (
12        "-----",
13        "-----",
14        "-----",
15        "-----"
16    ))
17
18 a = 1
19 b = 9
20 d = 0.5
21 L = 4
22
23 alpha = 0.5
24 Em = 1e-6
25
```

					ЛР.ПО4.190333-....						
Изм	Лист	№ докум.	Подп.	Дата							
Разраб.	Галанин				Лабораторная работа №1 Линейная искусственная нейронная сеть. Правило Видроу-Хоффа			Лит.	Лист	Листов	
Пров.	Крощенко							Л	2	6	
Н. контр.	Крощенко							БрГТУ			
Утв.											

```

26 w = []
27 for i in range(L):
28     w.append(random.random() * 0.02 - 0.01)
29     print("w[%d] = %lf" % (i, w[i]))
30
31 T = 1
32
33 m = 30
34 m2 = 15
35 e = []
36 for i in range(m + m2):
37     step = 0.1
38     x = step * i
39     e.append(a * math.sin(b * x) + d)
40
41 while 1:
42     E = 0
43     for i in range (m - L):
44         y1 = 0
45         for j in range(L):
46             y1 += w[j] * e[i + j]
47         y1 -= T
48
49         for j in range(L):
50             w[j] -= alpha * ( y1 - e[i + L] ) * e[i + j]
51
52         T += alpha * (y1 - e[i + L])
53
54         E += 0.5 * math.pow( (y1 - e[i + L]), 2)
55
56     if E < Em:
57         break
58
59 print("Результаты обучение:")
60 print_headTable()
61
62 trainingSample = []
63
64 for i in range(m):

```

```

65 trainingSample.append(0)
66
67 if i % L == 0:
68     print("%d эпоха" % (i / 4 + 1))
69
70 for j in range(L):
71     trainingSample[i] += w[j] * e[j + i - L]
72
73 trainingSample[i] -= T
74
75 print("| %20d | %20lf | %20lf | %20lf |" % (
76     i,
77     e[i],
78     trainingSample[i],
79     e[i] - trainingSample[i]
80 ))
81
82 print("Результаты прогнозирования:")
83 print_headTable()
84
85 for i in range(m2):
86     trainingSample.append(0)
87
88 if i % L == 0:
89     print("%d эпоха" % (i / 4 + 1))
90
91 for j in range(L):
92     trainingSample[i + m] += w[j] * e[m - L + j + i]
93
94 trainingSample[i + m] -= T
95
96 print("| %20d | %20lf | %20lf | %20lf |" % (
97     i + m,
98     e[i + m],
99     trainingSample[i + m],
100    e[i + m] - trainingSample[i + m]
101 ))

```

Листинг: "Консольный вывод"

```

1 w[0] = -0.000906
2 w[1] = 0.003552
3 w[2] = 0.006427
4 Результаты обучение:
5 |           Итерация |      Эталонное значение |      Полученное значение |           Отклонение |
6 | ----- | ----- | ----- | ----- |
7 1 эпоха
8 |           0 |      0.400000 |      1.068464 |      -0.668464 |
9 |           1 |      1.966654 |     -1.091010 |       3.057664 |
10 |          2 |      2.347695 |      0.273130 |       2.074565 |
11 1 эпоха
12 |           3 |      1.254760 |      1.254918 |      -0.000158 |
13 |           4 |     -0.485041 |     -0.484884 |      -0.000157 |
14 |           5 |     -1.555060 |     -1.554977 |      -0.000083 |
15 2 эпоха
16 |           6 |     -1.145529 |     -1.145536 |       0.000007 |
17 |           7 |      0.433628 |      0.433582 |       0.000045 |
18 |           8 |      1.987336 |      1.987332 |       0.000003 |
19 3 эпоха
20 |           9 |      2.339780 |      2.339867 |      -0.000088 |
21 |          10 |      1.224237 |      1.224396 |      -0.000159 |
22 |          11 |     -0.515072 |     -0.514916 |      -0.000156 |
23 4 эпоха
24 |          12 |     -1.561872 |     -1.561791 |      -0.000082 |
25 |          13 |     -1.123967 |     -1.123975 |       0.000008 |
26 |          14 |      0.467246 |      0.467201 |       0.000045 |
27 4 эпоха
28 |          15 |      2.007569 |      2.007567 |       0.000002 |
29 |          16 |      2.331316 |      2.331405 |      -0.000089 |
30 |          17 |      1.193481 |      1.193640 |      -0.000159 |
31 5 эпоха
32 |          18 |     -0.544844 |     -0.544689 |      -0.000155 |
33 |          19 |     -1.568130 |     -1.568050 |      -0.000080 |
34 |          20 |     -1.101974 |     -1.101984 |       0.000009 |
35 6 эпоха
36 |          21 |      0.500845 |      0.500800 |       0.000045 |
37 |          22 |      2.027347 |      2.027347 |       0.000000 |
38 |          23 |      2.322305 |      2.322397 |      -0.000091 |
39 7 эпоха
40 |          24 |      1.162501 |      1.162661 |      -0.000160 |
41 |          25 |     -0.574349 |     -0.574195 |      -0.000154 |
42 |          26 |     -1.573831 |     -1.573753 |      -0.000078 |
43 7 эпоха
44 |          27 |     -1.079557 |     -1.079568 |       0.000011 |
45 |          28 |      0.534416 |      0.534371 |       0.000045 |
46 |          29 |      2.046666 |      2.046667 |      -0.000001 |
47 Результаты прогнозирование:
48 |           Итерация |      Эталонное значение |      Полученное значение |           Отклонение |
49 | ----- | ----- | ----- | ----- |
50 1 эпоха
51 |          30 |      2.312752 |      2.312845 |      -0.000093 |
52 |          31 |      1.131305 |      1.131466 |      -0.000161 |
53 |          32 |     -0.603579 |     -0.603425 |      -0.000153 |
54 1 эпоха
55 |          33 |     -1.578974 |     -1.578898 |      -0.000076 |
56 |          34 |     -1.056722 |     -1.056734 |       0.000012 |
57 |          35 |      0.567949 |      0.567904 |       0.000045 |
58 2 эпоха

```

59		36		2.065519		2.065522		-0.000003	
60		37		2.302657		2.302752		-0.000095	
61		38		1.099903		1.100064		-0.000161	
62	3 эпоха								
63		39		-0.632524		-0.632372		-0.000152	
64		40		-1.583558		-1.583483		-0.000075	
65		41		-1.033474		-1.033487		0.000013	
66	4 эпоха								
67		42		0.601434		0.601389		0.000045	
68		43		2.083901		2.083905		-0.000004	
69		44		2.292025		2.292121		-0.000096	

Вывод: "Изучили обучение и функционирование линейной ИНС при решении задач прогнозирования".