

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

Условие

1. Написать на любом ЯВУ программу моделирования прогнозирующей линейной ИНС. Для тестирования использовать функцию:

$$y = a * \sin(b * x) + d$$

Варианты заданий приведены в следующей таблице:

№ варианта	a	b	d	Кол-во входов ИНС
1	1	5	0.1	3
2	2	6	0.2	4
3	3	7	0.3	5
4	4	8	0.4	3
5	1	9	0.5	4
6	2	5	0.6	5
7	3	6	0.1	3
8	4	7	0.2	4
9	1	8	0.3	5
10	2	9	0.4	3
11	3	5	0.5	4

Обучение и прогнозирование производить на 30 и 15 значениях соответственно табулируя функцию с шагом 0.1. Скорость обучения выбирается студентом самостоятельно, для чего моделирование проводится несколько раз для разных α . Результаты

					ЛР.ПО4.190333-....				
Изм	Лист	№ докум.	Подп.	Дата	Лабораторная работа №1 Линейная искусственная нейронная сеть. Правило Видроу-Хоффа		Лит.	Лист	Листов
Разраб.	Галанин						Л	2	8
Пров.	Крощенко						БрГТУ		
Н. контр.	Крощенко								
Утв.									

оцениваются по двум критериям - скорости обучения и минимальной достигнутой ошибке. Необходимо заметить, что эти критерии в общем случае являются взаимоисключающими, и оптимальные значения для каждого критерия достигаются при разных α .

2. Результаты представить в виде отчета содержащего:

- 1) Титульный лист
- 2) Цель работы
- 3) Задание
- 4) Результаты обучения: таблицу со столбцами: эталонное значение, полученное значение, отклонение; график изменения ошибки в зависимости от итерации.
- 5) Результаты прогнозирования: таблицу со столбцами: эталонное значение, полученное значение, отклонение.
- 6) Выводы по лабораторной работе.

Результаты для пунктов 3 и 4 приводятся для значения α , при котором достигается минимальная ошибка. В выводах анализируются все полученные результаты.

Контрольные вопросы

- а) ИНС какой архитектуры Вы использовали в данной работе? Опишите принцип построения этой ИНС.
- б) Как функционирует используемая Вами ИНС?
- в) Опишите (в общих чертах) алгоритм обучения Вашей ИНС.
- г) Как формируется обучающая выборка для решения задачи прогнозирования?
- д) Как выполняется многошаговое прогнозирование временного ряда?
- е) Предложите критерий оценки качества результатов прогноза.

Исходный код

Листинг: Исходный код

```
1 import math
2
3 def print_headTable():
4     print("| %16s | %16s | %16s | %16s |" % (
5         "y[]",
6         "etalonoe zn",
7         "Polychenoe zn",
8         "Otklonenie"
9     ))
10    print("| %16s | %16s | %16s | %16s |" % (
11        "-----",
12        "-----",
13        "-----",
14        "-----"
15    ))
16
17 a = 1
18 b = 9
19 d = 0.5
20 L = 4
21
22 alpha = 10e-6
23 Em = 10e-6
24
25 w = []
26 for i in range(L):
27     w.append(0)
28
29 T = 1
30
31 m = 30
32 m2 = 15
33 etalon = []
34 for i in range(m + m2):
35     etalon.append(0)
```

```

36
37 for i in range(m + m2):
38     step = 0.1
39     x = step * i
40     etalon[i] = alpha * math.sin(b * x) + d
41
42 while 1:
43     E = 0
44
45     for i in range (m - L):
46         y1 = 0
47         for j in range(L):
48             y1 += w[j] * etalon[i + j]
49         y1 -= T
50
51         for j in range(L):
52             w[j] -= alpha * ( y1 - etalon[i + L] ) * etalon[i + j]
53
54         T += alpha * (y1 - etalon[i + L])
55
56         E += 0.5 * math.pow( (y1 - etalon[i + L]), 2)
57
58     if E < Em:
59         break;
60
61 print("Training sample:")
62 print_headTable()
63
64 trainingSample = []
65
66 for i in range(m + m2):
67     trainingSample.append(0)
68
69 for i in range(m):
70     trainingSample[i] = 0
71     for j in range(L):
72         trainingSample[i] += w[j] * etalon[j + i - L]
73
74     trainingSample[i] -= T

```

```

75
76     print("| %16d | %16lf | %16lf | %16lf |" % (
77         i,
78         etalon[i],
79         trainingSample[i],
80         etalon[i] - trainingSample[i]
81     ))
82
83     print("Forecasting the future:")
84     print_headTable()
85
86     for i in range(m2):
87         trainingSample[i + m] = 0
88
89         for j in range(L):
90             trainingSample[i + m] += w[j] * etalon[m - L + j + i]
91
92         trainingSample[i] += T
93
94     print("| %16d | %16lf | %16lf | %16lf |" % (
95         i + m,
96         etalon[i],
97         trainingSample[i],
98         etalon[i] - trainingSample[i]
99     ))

```

Вывод в консоль

Листинг: Вывод в консоль

Training sample :

y []	etalonoe zn	Polychenoe zn	Otklonenie
0	0.500000	0.499127	0.000873
1	0.500008	0.499130	0.000878
2	0.500010	0.499133	0.000877
3	0.500004	0.499133	0.000871
4	0.499996	0.499131	0.000864
5	0.499990	0.499129	0.000861
6	0.499992	0.499123	0.000869
7	0.500000	0.499116	0.000884
8	0.500008	0.499115	0.000893
9	0.500010	0.499119	0.000890
10	0.500004	0.499127	0.000877
11	0.499995	0.499131	0.000864
12	0.499990	0.499129	0.000861
13	0.499992	0.499123	0.000870
14	0.500000	0.499116	0.000884
15	0.500008	0.499115	0.000893
16	0.500010	0.499120	0.000890
17	0.500004	0.499127	0.000877
18	0.499995	0.499131	0.000864
19	0.499990	0.499129	0.000861
20	0.499992	0.499123	0.000870
21	0.500001	0.499116	0.000884
22	0.500008	0.499115	0.000893
23	0.500010	0.499120	0.000890
24	0.500004	0.499127	0.000877
25	0.499995	0.499131	0.000864
26	0.499990	0.499129	0.000861
27	0.499993	0.499122	0.000870
28	0.500001	0.499116	0.000885
29	0.500008	0.499115	0.000893

Forecasting the future :

y []	etalonoe zn	Polychenoe zn	Otklonenie
30	0.500000	0.749566	-0.249566

38		31		0.500008		0.749569		−0.249561	
39		32		0.500010		0.749571		−0.249561	
40		33		0.500004		0.749572		−0.249567	
41		34		0.499996		0.749570		−0.249574	
42		35		0.499990		0.749568		−0.249578	
43		36		0.499992		0.749561		−0.249569	
44		37		0.500000		0.749555		−0.249555	
45		38		0.500008		0.749553		−0.249545	
46		39		0.500010		0.749558		−0.249548	
47		40		0.500004		0.749565		−0.249561	
48		41		0.499995		0.749570		−0.249574	
49		42		0.499990		0.749568		−0.249578	
50		43		0.499992		0.749561		−0.249569	
51		44		0.500000		0.749555		−0.249554	

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