

Министерство образования Республики Беларусь
Учреждение образования
“Брестский государственный университет”
Факультет электронно-информационных систем
Кафедра ИИТ

Отчет по лабораторной работе №1
По дисциплине “Методы и алгоритмы принятия решений”
Тема: “Правило обучения Видроу-Хоффа”
Вариант №7

Выполнил:
Кравцевич Г.А.
студент группы ПО-7

Проверил:
А. А. Крощенко,
ст.преп.кафедры ИИТ
7 сентября 2017 г.

Брест 2021

Цель работы:

Изучить обучение и функционирование линейной ИНС при решении задач прогнозирования.

Задание:

Написать программу моделирования прогнозирующей линейной ИНС. Для тестирования использовать функцию $y = a * \sin(b * x) + d$. ($a = 3$, $b = 6$, $d = 0.1$, количество входов ИНС: 3)

Выполнение:**Код программы:**

```
import math
import random
```

```
STEP = 0.1
```

```
VECTOR_SIZE = 3
```

```
EXPECTED_ERROR = 0.000001
```

```
weights = [random.uniform(-1, 1) for _ in range(VECTOR_SIZE)]
```

```
T = random.uniform(-1, 1)
```

```
def get_func_value(x: float) -> float:
```

```
    a, b, d = 3, 6, 0.1
```

```
    return a * math.sin(b * x) + d
```

```
def get_training_data() -> list:
```

```
    data_size = 10
```

```
    training_data = []
```

```
    current_x = 1
```

```
    for _ in range(data_size):
```

```
        line = []
```

```
        vector = []
```

```
        for _ in range(VECTOR_SIZE):
```

```
vector.append(get_func_value(current_x))
current_x += STEP
```

```
line.append(vector)
line.append(get_func_value(current_x))
current_x -= STEP * 2
```

```
training_data.append(line)
```

```
return training_data
```

```
def train_model() -> None:
```

```
    common_error = 0
```

```
    while True:
```

```
        for line in get_training_data():
```

```
            vector = line[0]
```

```
            expected_result = line[-1]
```

```
            obtained_result = predict(vector)
```

```
            error = obtained_result - expected_result
```

```
            common_error += error ** 2
```

```
            change_weights(error, vector)
```

```
        if math.fabs(common_error) <= EXPECTED_ERROR:
```

```
            break
```

```
    common_error = 0
```

```
def predict(vector: list) -> float:
```

```
    output = 0
```

```
    for index, value in enumerate(vector):
```

```
        output += value * weights[index]
```

```
    output -= T
```

return output

```
def change_weights(error: float, vector: list) -> None:
    global T, weights

    for w_index in range(len(weights)):
        weights[w_index] -= STEP * error * vector[w_index]

    T += STEP * error

if __name__ == '__main__':
    train_model()
    print('The training is complete!')

    test_vector = [get_func_value(1 + STEP * n) for n in range(3)]
    right_answer = get_func_value(1 + STEP * 3)
    print('Test vector:', *test_vector, sep=' ')
    print('Obtained result: ', predict(test_vector))
    print('Right result: ', right_answer)
```

Спецификация вывода:

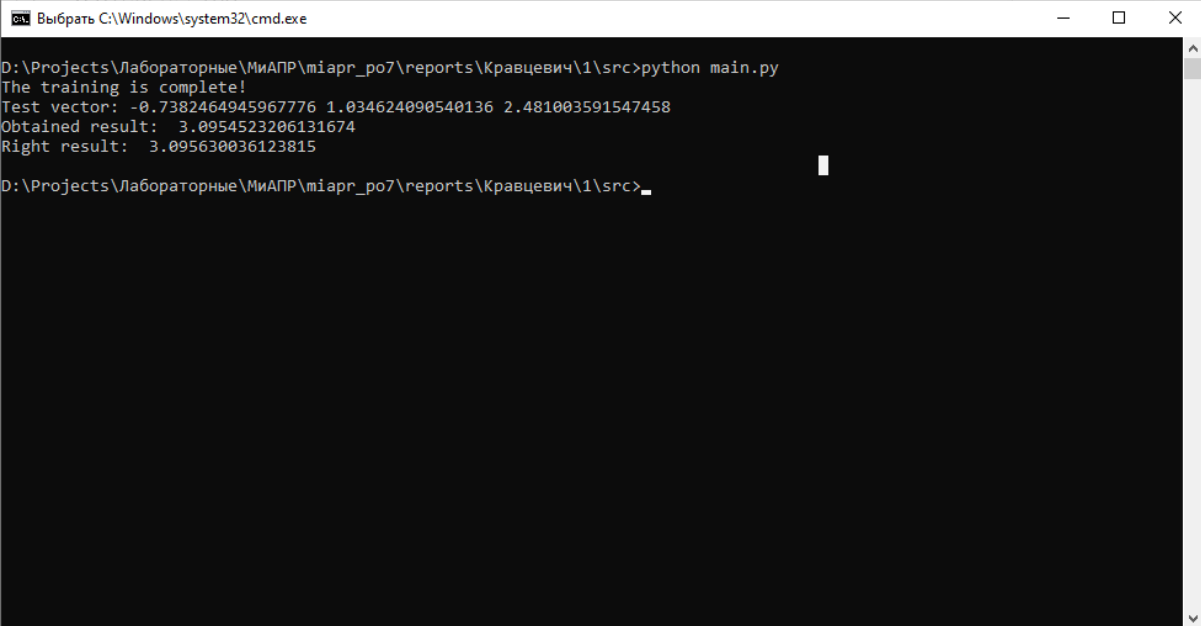
The training is complete!

Test vector: <значение вектора для проверки>

Obtained result: <полученный результат>

Right result: <правильный ответ>

Рисунки с результатами работы программы:



```
Выбрать C:\Windows\system32\cmd.exe
D:\Projects\Лабораторные\МиАПР\miapr_po7\reports\Кравцевич\1\src>python main.py
The training is complete!
Test vector: -0.7382464945967776 1.034624090540136 2.481003591547458
Obtained result: 3.0954523206131674
Right result: 3.095630036123815
D:\Projects\Лабораторные\МиАПР\miapr_po7\reports\Кравцевич\1\src>_
```

Вывод:

Изучил обучение и функционирование линейной ИНС при решении задач прогнозирования. Написал программу на языке Python