МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №5

Специальность АС-66

Выполнил
А. С. Рогожин,
студент группы АС-66

Проверил
А. А. Крощенко,
ст. преп. кафедры ИИТ,
«___» _____2025 г.

Брест 2025

Цель: выполнить моделирование прогнозирующей нелинейной ИНС. Для генерации обучающих и тестовых данных использовать функцию
Вариант 10:

$$y = a\cos(bx) + c\sin(dx) \ .$$

Варианты заданий приведены в следующей таблице:

№ варианта	a	b	c	d	Кол-во входов ИНС	Кол-во НЭ в скрытом слое
1	0.1	0.1	0.05	0.1	6	2
2	0.2	0.2	0.06	0.2	8	3
3	0.3	0.3	0.07	0.3	10	4
4	0.4	0.4	0.08	0.4	6	2
5	0.1	0.5	0.09	0.5	8	3
6	0.2	0.6	0.05	0.6	10	4
7	0.3	0.1	0.06	0.1	6	2
8	0.4	0.2	0.07	0.2	8	3
9	0.1	0.3	0.08	0.3	10	4
10	0.2	0.4	0.09	0.4	6	2
11	0.3	0.5	0.05	0.5	8	3

Для прогнозирования использовать многослойную ИНС с одним скрытым слоем. В качестве функций активации для скрытого слоя использовать сигмоидную функцию, для выходного - линейную.

```
import numpy as np
import matplotlib.pyplot as plt

a, b, c, d = 0.2, 0.4, 0.09, 0.4

def func(x):
    return a * np.cos(b * x) + c * np.sin(d * x)

x = np.linspace(0, 20, 400)
y = func(x)

window = 6
X, Y = [], []

for i in range(len(y) - window):
    X.append(y[i:i + window])
    Y.append(y[i + window])

X = np.array(X)
Y = np.array(Y)

train_size = 300
X_train, X_test = X[:train_size], X[train_size:]
Y_train, Y_test = Y[:train_size], Y[train_size:]

np.random.seed(0)
hidden = 2
```

```
W1 = np.random.randn(window, hidden) * 0.5
b1 = np.zeros(hidden)
W2 = np.random.randn(hidden) * 0.5
b2 = 0.0

def sigmoid(z):
    return 1 / (1 + np.exp(-z))

lr = 0.05
epochs = 800
errors = []

for epoch in range(epochs):
    z1 = X_train @ W1 + b1
    a1 = sigmoid(z1)
    y_pred = a1 @ W2 + b2

    err = y_pred - Y_train
    mse = np.mean(err ** 2)
    errors.append(mse)

    dW2 = a1.T @ err * (2 / len(X_train))
    db2 = np.mean(err) * 2

    da1 = err[:, None] * W2
    dz1 = da1 * (a1 * (1 - a1))

    dW1 = X_train.T @ dz1 * (2 / len(X_train))
    db1 = np.mean(dz1, axis=0) * 2

    W1 -= lr * dW1
    b1 -= lr * db1
    W2 -= lr * dW2
    b2 -= lr * db2

def mlp_predict(X):
    return sigmoid(X @ W1 + b1) @ W2 + b2

train_pred = mlp_predict(X_train)
test_pred = mlp_predict(X_test)

plt.figure(figsize=(8, 4))
plt.plot(Y_train, label="Эталон")
plt.plot(train_pred, label="Прогноз")
plt.title("Прогноз на обучающем участке")
plt.legend()
plt.grid()
plt.show()

plt.figure(figsize=(8, 4))
plt.plot(errors)
plt.title("Ошибка обучения (MSE)")
plt.xlabel("Итерация")
plt.ylabel("Ошибка")
plt.grid()
plt.show()

def print_table(name, Y_true, Y_pred, limit=20):
    print("\n-----", name, "-----")
    print("Эталон\t\tПрогноз\t\tОтклонение")
    for t, p in list(zip(Y_true, Y_pred))[:limit]:
        print(f"{t:.6f}\t{p:.6f}\t{t - p:.6f}")

print_table("ОБУЧЕНИЕ", Y_train, train_pred)
print_table("ПРОГНОЗ", Y_test, test_pred)
```

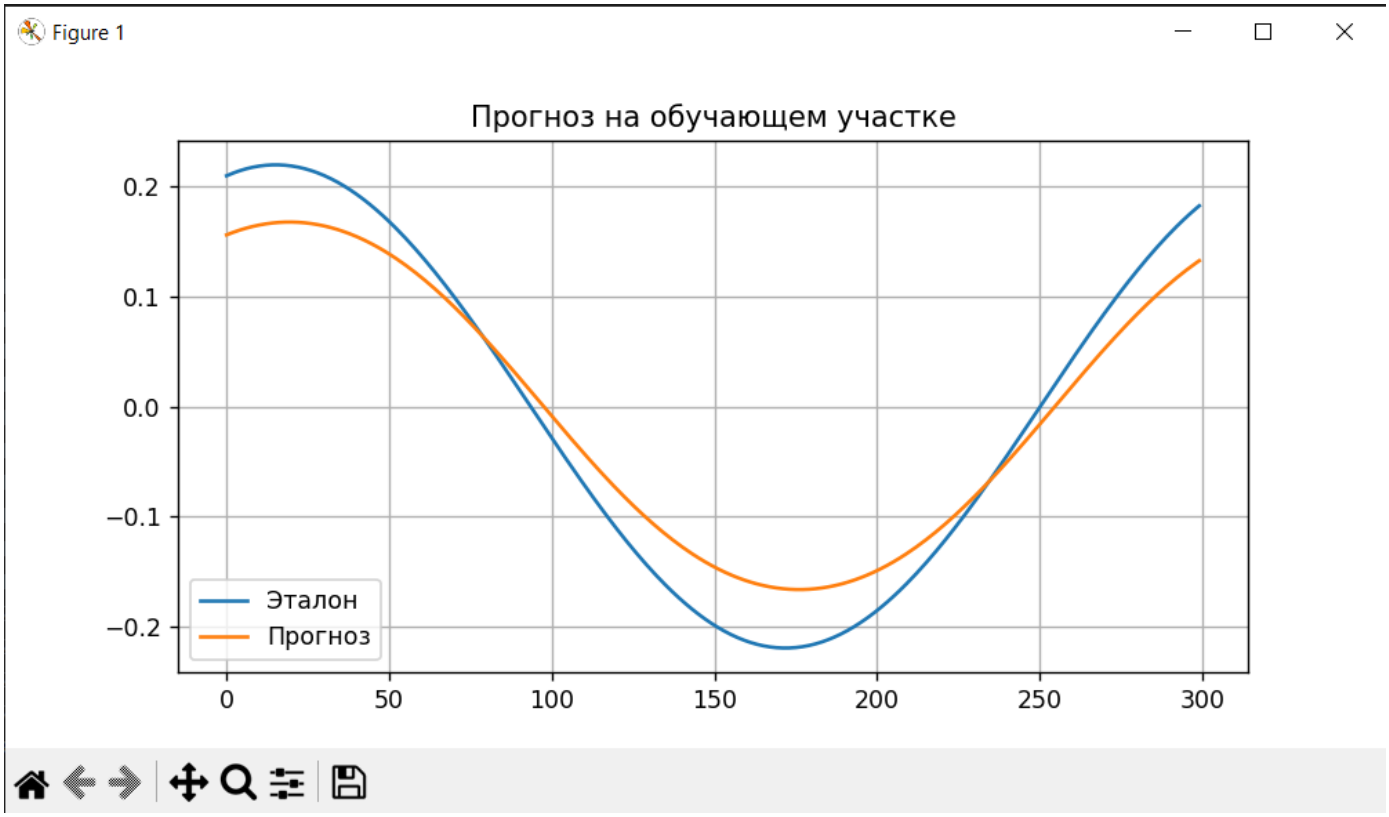
Figure 1
Прогноз на обучающем участке
0.2
0.1
0.0
−0.1
−0.2
0
50
100
150
200
250
300
Эталон
Прогноз

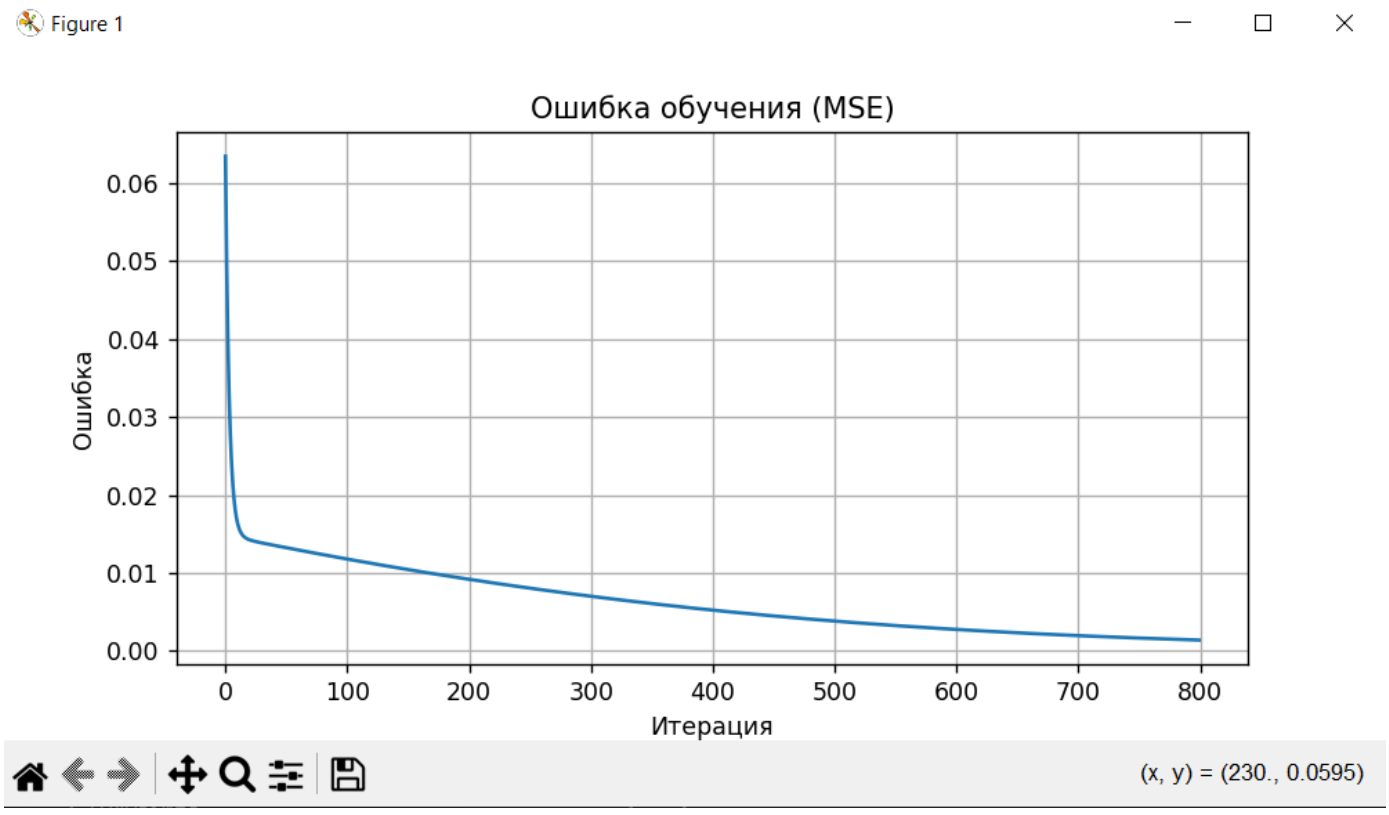
Figure 1
Ошибка обучения (MSE)
Ошибка
0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
100
200
300
400
500
600
700
800
Итерация
(x, y) = (230., 0.0595)

```
PS D:\Uni\OMO\ml_as66\reports> & C:/Python

----- ОБУЧЕНИЕ -----
Эталон          Прогноз         Отклонение
0.209355        0.155717        0.053639
0.210624        0.156886        0.053738
0.211807        0.157994        0.053813
0.212905        0.159041        0.053864
0.213918        0.160027        0.053891
0.214844        0.160951        0.053893
0.215685        0.161813        0.053871
0.216438        0.162614        0.053824
0.217105        0.163353        0.053752
0.217684        0.164029        0.053655
0.218176        0.164644        0.053532
0.218580        0.165196        0.053384
0.218896        0.165686        0.053210
0.219125        0.166114        0.053010
0.219265        0.166480        0.052785
0.219317        0.166783        0.052534
0.219281        0.167023        0.052257
0.219156        0.167202        0.051955
0.218944        0.167317        0.051627
0.218643        0.167370        0.051273
----- ПРОГНОЗ -----
Эталон          Прогноз         Отклонение
0.184560        0.134330        0.050230
0.186899        0.136291        0.050608
0.189162        0.138194        0.050968
0.191349        0.140040        0.051309
0.193459        0.141828        0.051631
0.195491        0.143558        0.051933
0.197445        0.145230        0.052216
0.199320        0.146842        0.052478
0.201114        0.148396        0.052719
0.202828        0.149889        0.052938
0.204460        0.151323        0.053136
0.206009        0.152697        0.053312
0.207476        0.154011        0.053465
0.208860        0.155264        0.053596
0.210159        0.156456        0.053703
0.211374        0.157587        0.053787
0.212505        0.158657        0.053848
0.213549        0.159666        0.053884
0.214508        0.160613        0.053895
0.215381        0.161498        0.053883
PS D:\Uni\OMO\ml_as66\reports>
```

Вывод: выполнили моделирование прогнозирующей нелинейной ИНС для прогнозирования функции a*cosbx+c*sindx.