

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

Лабораторная работа №5  
По дисциплине: «ОМО»  
Тема:” Нелинейные ИНС в задачах регрессии”

Выполнил:  
Студент 3-го курса  
Группы АС-66  
Куган Н.Л.  
Проверил:  
Крощенко А.А.

Брест 2025

Цель: Выполнить моделирование прогнозирующей нелинейной ИНС.

## Вариант 6

### Задание:

- Выполнить моделирование прогнозирующей нелинейной ИНС. Для генерации обучающих и тестовых данных использовать функцию

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

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

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.neural_network import MLPRegressor

a = 0.2
b = 0.6
c = 0.05
d = 0.6
INPUTS = 10
HIDDEN = 4

def f(t):
    return a * np.cos(b * t) + c * np.sin(d * t)
N = 500
t = np.linspace(0, 50, N)
y = f(t)

X_train = []
y_train = []

for i in range(N - INPUTS):
    X_train.append(y[i:i + INPUTS])
    y_train.append(y[i + INPUTS])

X_train = np.array(X_train)
y_train = np.array(y_train)
```

```

model = MLPRegressor(
    hidden_layer_sizes=(HIDDEN,),
    activation='logistic',      # сигмоида
    solver='adam',
    max_iter=5000,
    learning_rate_init=0.01
)

model.fit(X_train, y_train)

y_pred_train = model.predict(X_train)

# многошаговый прогноз
forecast_steps = 50
start_seq = list(y[-INPUTS:])
forecast = []

cur = start_seq.copy()

for _ in range(forecast_steps):
    pred = model.predict([cur])[0]
    forecast.append(pred)
    cur = cur[1:] + [pred]

plt.figure(figsize=(12, 5))
plt.plot(y, label="Истинная функция")
plt.plot(np.arange(INPUTS, N), y_pred_train, label="Прогноз (обучение)")
plt.title("График функции и результата обучения")
plt.legend()
plt.grid()
plt.show()

plt.figure(figsize=(12, 5))
plt.plot(model.loss_curve_)
plt.title("Изменение ошибки (loss) от итерации")
plt.xlabel("итерация")
plt.ylabel("loss")
plt.grid()
plt.show()

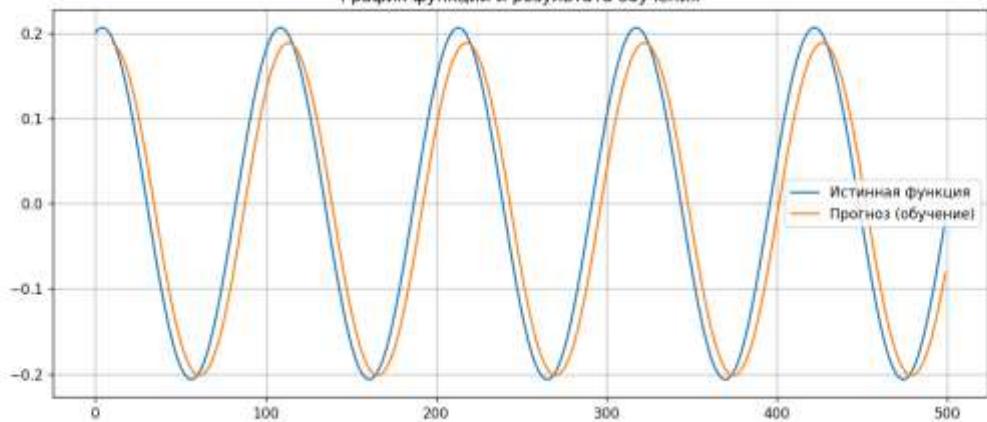
print("\nТаблица (первые 20 значений – обучение):")
print("Эталон\tПолучено\tОтклонение")
for i in range(20):
    print(f"{y_train[i]: .6f}\t{y_pred_train[i]: .6f}\t{(y_pred_train[i]-y_train[i]): .6f}")

print("\nТаблица прогнозирования (50 шагов):")
print("Эталон отсутствует (будущее) – выводим только предсказания:")
for i, val in enumerate(forecast):
    print(f"Шаг {i+1}: {val:.6f}")

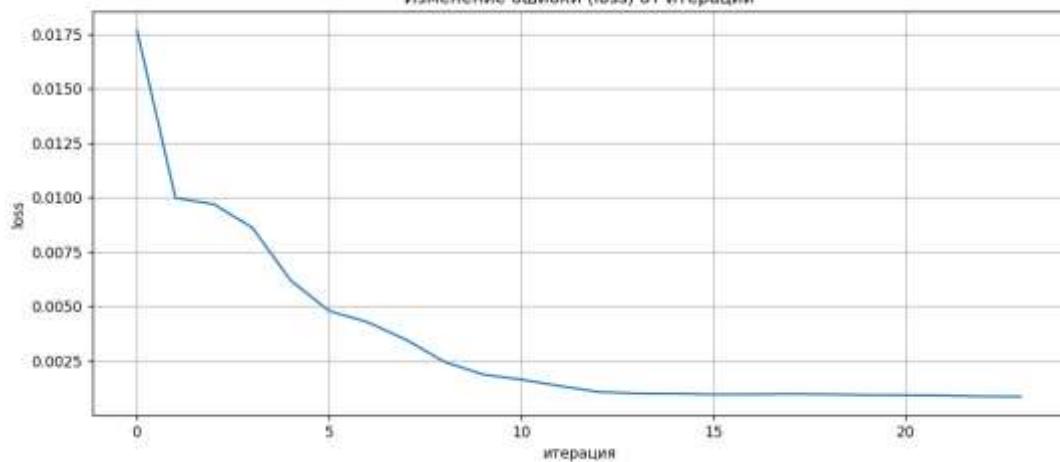
```

Результаты:

График функции и результата обучения



Изменение ошибки (loss) от итерации



Вывод: На практике выполнили моделирование прогнозирующей нелинейной ИНС.