**Міністерство освіти і науки України**

**Національний технічний університет України «КПІ» імені Ігоря Сікорського**

**Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ**

**з лабораторної роботи №6**

**з навчальної дисципліни «Вступ до технології Data Science»**

**Тема:**

**ДОСЛІДЖЕННЯ ПРОЦЕСІВ ЗАСТОСУВАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ЗАДАЧ АНАЛІЗУ ДАНИХ**

**Виконав:**

Студент X курсу кафедри ОТ ФІОТ,

Навчальної групи ІА-73

Петренко П.П.

**Перевірив:**

Професор кафедри ОТ ФІОТ

Писарчук О.О.

**Київ 2023**

**І. Мета:**

виявити дослідити та узагальнити особливості застосування штучного інтелекту для аналізу даних з використанням спеціалізованих пакетів мови програмування Python.

**ІІ. Завдання:**

Лабораторія провідної ІТ-компанії реалізує проект із розробки системи підтримки прийняття рішень для автоматизації процесів вироблення рішень у сфері задач електронної комерції.

Замовниками програмної системи – низка оптових компаній.

Вам, як Data Science [Engineer](https://jobs.dou.ua/companies/gravitum/vacancies/147764/) поставлене наступне завдання.

Розробити скрипт мовою Python що реалізує прогнозування динаміки зміни контрольованого процесу відповідно до технічних умов, заданих у таблиць Д1, Д2 додатку 1.

Завдання реалізувати у відповідності до пунктів:

* 1. Здійснити прогнозування динаміки зміни досліджуваного процесу;
  2. Здійснити розрахунок статистичних характеристик результатів прогнозування;
  3. Порівняти статистичні характеристики та прогнозовані оцінки з фактичними даними (модельними, або тими, що отримані в процесі спостереження) та зробити висновки.

Результат представити у формі:

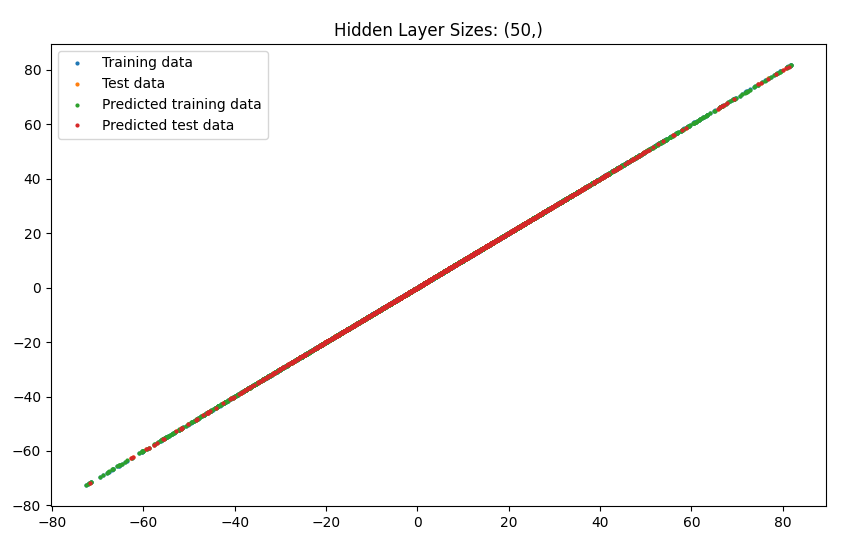
1.5. Результати архітектурного проектування скрипта, що реалізує технічні умови задачі.

1.6. Програмний скрипт, результати його функцілнування.

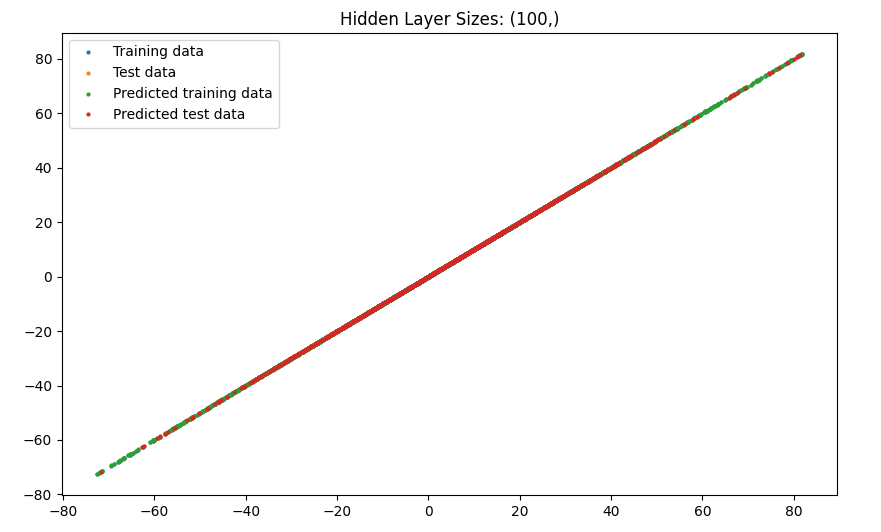
1.7. Результати дослідження ефективності та їх аналіз.

| 7, 22 | Розробити програмний скрипт, що реалізує:  1. Модель зміни досліджуваного процесу – dataset за лінійним законом, у 5000 дискретних вимірів, з нормальним шумом (нульове середнє та СКВ похибки 20) та 10% аномальних вимірів з рівномірним розподілом в межах вибірки;  2. Штучну нейронну мережу з прогнозування даних відповідно до параметрів заданого в п.1 dataset;  3. Відображення процесу прогнозування у формі графіків.  4. Дослідити залежність точності прогнозування числового ряду від структури мережі. |
| --- | --- |

| 7, 22 | Розробити програмний скрипт, що реалізує:  1. Отримання даних динаміки захворювання на обрану хворобу, обраного регіону з ресурсів Інтернет за обраний проміжок часу;  2. Прогнозування динаміки захворювання на 2 інтервали спостереження з використанням обраного статистичного методу прогнозування.  3. Прогнозування зміни динаміки захворювання на 2 інтервали спостереження з використанням штучних нейронних мереж.  4. Порівняння та відображення статистичних характеристик точності і абсолютних значень прогнозованих оцінок на останній точки часового інтервалу прогнозу.  Сформувати обґрунтовані висновки про вибір доцільного методу прогнозування. |
| --- | --- |

****

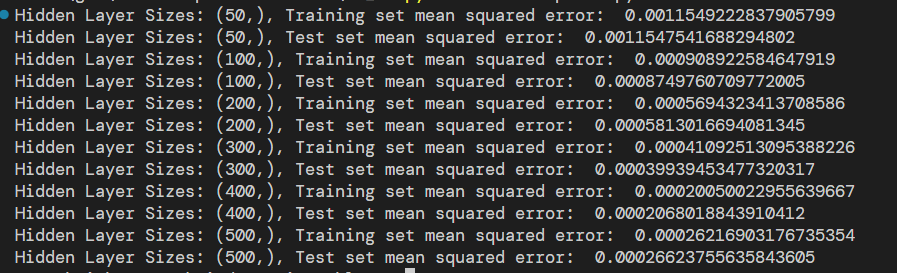
**Рисунок 1.**

****

**Рисунок 2.**

Зміна будь-якого параметру мережі ( кількість тренувальних епох, кількість нейронів у першому шарі, кількість шарів та нейронів у них, швидкість навчання та інше ) не призведе ні до яких змін при лінійному законі розподілу, оскільки такі дані завжди будуть лінією, і найкраще для цього завдання підходить лінійна регрессія.

Відповідно, і мережі не потрібно.

****

Як можемо бачити, Среденьоквадратична похибка змінюється дуже не суттєво.

**ІІІ. Довести адекватність сформованих моделей та працездатність розробленого скріпта.**

**Розроблений код повинен бути раціональним та відповідати вимогам до чистого коду.**

**Частина 1**

**import numpy as np**

**import pandas as pd**

**import matplotlib.pyplot as plt**

**from sklearn.neural\_network import MLPRegressor**

**from sklearn.model\_selection import train\_test\_split**

**from sklearn.metrics import mean\_squared\_error**

**# Step 1: Generate the dataset**

**x = np.linspace(0, 10, 5000)**

**y = x**

**noise = np.random.normal(0, 20, y.shape)**

**y += noise**

**anomalies = np.random.uniform(np.min(y), np.max(y), int(0.1 \* len(y)))**

**y[:len(anomalies)] = anomalies**

**data = pd.DataFrame({'x': x, 'y': y})**

**# Step 2: Create the neural network**

**X\_train, X\_test, y\_train, y\_test = train\_test\_split(data[['x']], data['y'], test\_size=0.2, random\_state=42)**

**# Step 3: Display the prediction process and Step 4: Investigate the dependence of the accuracy**

**layers = [(50,), (100,), (200,), (300,), (400,), (500,)]**

**for layer in layers:**

**model = MLPRegressor(hidden\_layer\_sizes=layer, activation='relu', solver='adam', max\_iter=500)**

**model.fit(X\_train, y\_train)**

**y\_pred\_train = model.predict(X\_train)**

**y\_pred\_test = model.predict(X\_test)**

**plt.figure(figsize=(10, 6))**

**plt.plot(X\_train, y\_train, 'o', label='Training data', markersize=2)**

**plt.plot(X\_test, y\_test, 'o', label='Test data', markersize=2)**

**plt.plot(X\_train, y\_pred\_train, 'o', label='Predicted training data', markersize=2)**

**plt.plot(X\_test, y\_pred\_test, 'o', label='Predicted test data', markersize=2)**

**plt.legend()**

**plt.title(f'Hidden Layer Sizes: {layer}')**

**plt.show()**

**print(f'Hidden Layer Sizes: {layer}, Training set mean squared error: ', mean\_squared\_error(y\_train, y\_pred\_train))**

**print(f'Hidden Layer Sizes: {layer}, Test set mean squared error: ', mean\_squared\_error(y\_test, y\_pred\_test))**

**III. Висновок:**

виявив дослідив та узагальнив особливості застосування штучного інтелекту для аналізу даних з використанням спеціалізованих пакетів мови програмування Python.

Виконав: студент Петренко П.П.