

## Лабораторна робота №3 ІАД

### Тема: Регресійний аналіз

#### Мета:

Ознайомитися з методами регресійного аналізу, навчитися працювати з реальними і штучно створеними наборами даних для класифікації, застосовувати моделі для класифікації даних та аналізувати результати.

Підготував студент 3 курсу МІТ-31

Крижановський Олександр

#### Створення набору даних за допомогою DatasetGenerator:

- Використовуйте спеціальні бібліотеки для генерації даних, наприклад, `sklearn.datasets.make_regression` у Python.

Приклад коду:

```
from sklearn.datasets import make_regression
import pandas as pd

# Генеруємо набір даних для регресії
X, y = make_regression(n_samples=100, n_features=1, noise=0.1)

# Перетворимо в DataFrame для зручності
data = pd.DataFrame(data=X, columns=['Feature'])
data['Target'] = y
data.to_csv('dataset.csv', index=False)
```

#### Завантаження набору даних для регресії з сайту Kaggle:

- Виберіть унікальний набір даних для кожного студента. Наприклад, можна вибрати набір даних щодо ціни будинків чи інших числових параметрів.
- Використовуйте бібліотеку `kaggle` для завантаження даних:

```
!pip install kaggle
!kaggle datasets download -d [dataset-path]
```

#### Генерація набору даних для регресії за допомогою Numpy або random (опціонально):

- Використовуйте `numpy` для створення випадкових даних.

```
import numpy as np

# Генерація випадкових даних
X = np.random.rand(100, 1) * 100
y = 3 * X.flatten() + np.random.randn(100) * 10

# Перетворимо в DataFrame для зручності
data = pd.DataFrame(data=X, columns=['Feature'])
data['Target'] = y
data.to_csv('generated_dataset.csv', index=False)
```

## Проведення моделювання з використанням регресійного аналізу:

- Використовуйте бібліотеку scikit-learn для побудови регресійної моделі.

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Завантаження даних
data = pd.read_csv('dataset.csv')
X = data[['Feature']]
y = data['Target']

# Розподіл на тренувальний та тестовий набори
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_

# Побудова та навчання моделі
model = LinearRegression()
model.fit(X_train, y_train)

# Прогнозування та оцінка
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
print(f'MSE: {mse}')
```

## Висновок

У ході виконання лабораторної роботи було створено та використано кілька наборів даних для задач регресії. За допомогою бібліотек sklearn, numpy, та інших інструментів, ми генерували власні набори даних та аналізували реальні дані з сайту Kaggle. Потім було проведено моделювання, використовуючи методи регресійного аналізу, для оцінки ефективності наших моделей. Результати були представлені та завантажені у GitHub репозиторій, що дозволяє зберегти виконану роботу та полегшує її подальший перегляд та аналіз.

Цей процес не тільки допомагає зрозуміти основи регресійного аналізу та генерації даних, але й надає практичний досвід роботи з інструментами

аналізу даних та управління проектами. Виконання цього завдання є важливим кроком у розвитку навичок у галузі Data Science та машинного навчання.