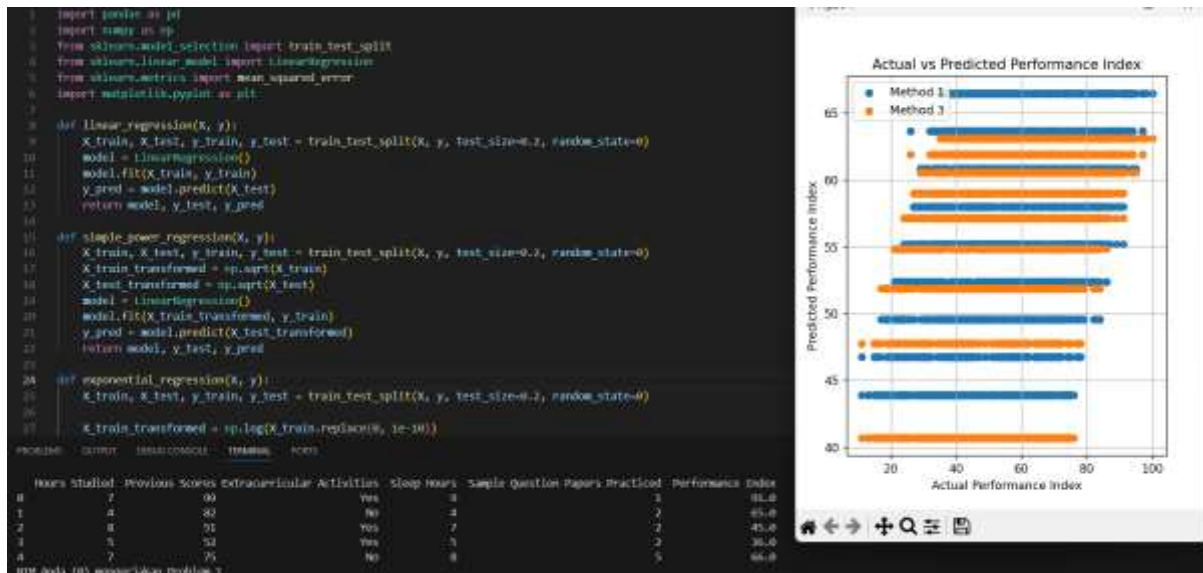


NAMA : Ridho Rabbani Refa Muhammad

NIM : 21120122140153

Kode Testing

NIM 0



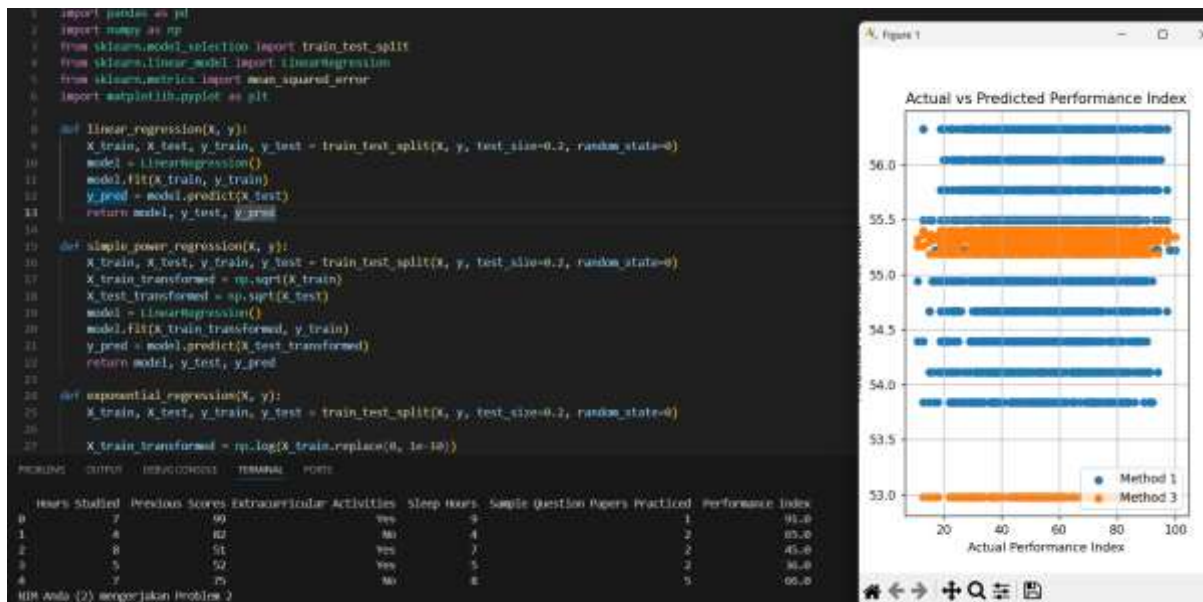
NIM Anda (0) mengerjakan Problem 1
Metode 1 RMSE: 17.321694540065916
Metode 3 RMSE: 17.38106941610259

NIM 1



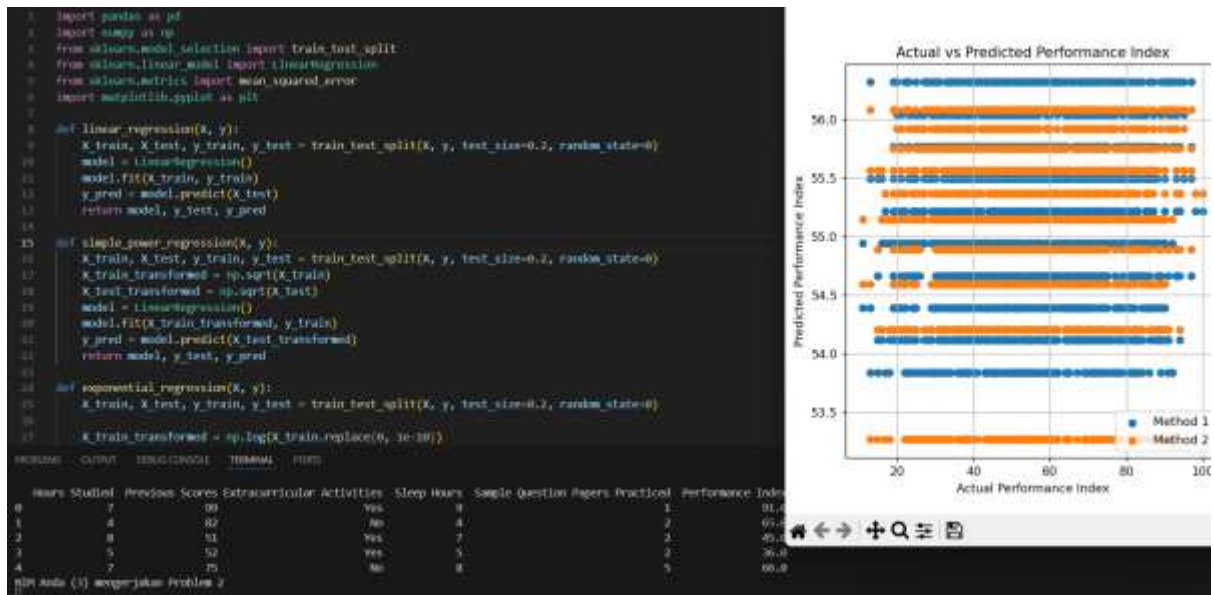
NIM Anda (1) mengerjakan Problem 1
 Metode 1 RMSE: 17.321694540065916
 Metode 2 RMSE: 17.318032149078263

NIM 2



NIM Anda (2) mengerjakan Problem 2
 Metode 1 RMSE: 18.534107625089508
 Metode 3 RMSE: 18.53063681833497

NIM 3



NIM Anda (3) mengerjakan Problem 2
 Metode 1 RMSE: 18.534107625089508
 Metode 2 RMSE: 18.52748673028731

Alur Kode

Kode ini adalah implementasi dari analisis regresi untuk memahami hubungan antara variabel input (durasi belajar atau jumlah latihan soal) dengan nilai ujian siswa. Metode regresi yang digunakan mencakup model linear, model pangkat sederhana, dan model eksponensial. Pengguna diminta untuk memasukkan digit terakhir dari NIM mereka, dan berdasarkan itu, program akan menentukan problem yang akan dikerjakan serta metode regresi yang akan digunakan.

Konsep:

- Persiapan Data:** Data siswa yang berisi durasi belajar, jumlah latihan soal, dan nilai ujian dimuat.
- Pemilihan Metode:** Berdasarkan digit terakhir NIM, program akan menentukan problem yang akan dikerjakan (Problem 1 atau Problem 2) serta metode regresi yang akan digunakan.
- Regresi dan Evaluasi:** Metode regresi (linear, pangkat sederhana, atau eksponensial) diterapkan pada data. Hasil regresi dievaluasi dengan menghitung Root Mean Squared Error (RMSE) untuk setiap metode.

4. **Visualisasi:** Hasil regresi divisualisasikan dengan plot yang menunjukkan hubungan antara nilai ujian yang diprediksi dan nilai ujian sebenarnya.

Implementasi Kode:

1. **Persiapan Data:** Menggunakan Pandas untuk memuat data dari file CSV.
2. **Pemilihan Metode:** Berdasarkan digit terakhir NIM, menentukan problem yang akan dikerjakan dan metode regresi yang akan digunakan.
3. **Regresi dan Evaluasi:** Menggunakan Scikit-learn untuk menerapkan regresi linear, regresi pangkat sederhana, dan regresi eksponensial. Menghitung RMSE untuk setiap metode.
4. **Visualisasi:** Menggunakan Matplotlib untuk membuat plot yang menampilkan nilai ujian yang diprediksi vs. nilai ujian sebenarnya.

Hasil Pengujian:

1. Data siswa dimuat dan ditampilkan.
2. Pengguna diminta untuk memasukkan digit terakhir NIM mereka.
3. Program menentukan problem yang akan dikerjakan dan metode regresi yang akan digunakan berdasarkan digit terakhir NIM.
4. Hasil regresi divisualisasikan dalam plot.
5. RMSE untuk setiap metode regresi ditampilkan.

Analisis Hasil:

- **Performa Metode Regresi:** RMSE digunakan untuk mengevaluasi performa metode regresi. Metode dengan RMSE yang lebih rendah dianggap lebih baik.
- **Interpretasi Grafik:** Grafik menunjukkan seberapa baik model regresi memprediksi nilai ujian siswa. Jika titik data tersebar dekat dengan garis regresi, itu menunjukkan model yang baik.
- **Penyesuaian Model:** Setiap metode regresi memiliki kelebihan dan kelemahan tersendiri tergantung pada hubungan antara variabel input dan output. Misalnya, model eksponensial cocok untuk data yang menunjukkan pertumbuhan eksponensial atau penurunan, sementara model linear cocok untuk hubungan linier.

Demikianlah, implementasi ini memberikan pemahaman tentang bagaimana metode regresi dapat diterapkan dalam analisis data dan bagaimana hasilnya dievaluasi untuk memahami hubungan antara variabel.

Penjelasan Kode

1. Persiapan Data:

```
import pandas as pd

data_path = "student_performance.csv"

data = pd.read_csv(data_path)

print("Loaded data:")

print(data.head())
```

Pada bagian ini, data siswa dimuat dari file CSV menggunakan library Pandas. Data kemudian ditampilkan untuk memberikan gambaran singkat tentang struktur dan konten data.

2. Input NIM dan Penentuan Tugas serta Metode:

```
# Input NIM pengguna

nim_last_digit = int(input("Masukkan digit terakhir NIM Anda: "))

if nim_last_digit % 4 == 0:

    problem_num = 1

    method_nums = [1, 3]

    print("NIM Anda mengerjakan Problem 1 dengan Metode 1 dan Metode 3")

elif nim_last_digit % 4 == 1:

    problem_num = 1

    method_nums = [1, 2]

    print("NIM Anda mengerjakan Problem 1 dengan Metode 1 dan Metode 2")

elif nim_last_digit % 4 == 2:
```

```

problem_num = 2

method_nums = [1, 3]

print("NIM Anda mengerjakan Problem 2 dengan Metode 1 dan Metode 3")

elif nim_last_digit % 4 == 3:

    problem_num = 2

    method_nums = [1, 2]

    print("NIM Anda mengerjakan Problem 2 dengan Metode 1 dan Metode 2")

else:

    raise ValueError("Invalid NIM.")

```

Pada bagian ini, pengguna diminta untuk memasukkan digit terakhir dari NIM mereka. Berdasarkan digit tersebut, program menentukan problem yang akan dikerjakan (Problem 1 atau Problem 2) dan metode regresi yang akan digunakan (Metode 1, Metode 2, atau Metode 3). Informasi tersebut kemudian ditampilkan kepada pengguna.

3. Regresi dan Evaluasi:

```

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_squared_error

import numpy as np

if problem_num == 1:

    X = data[['Hours Studied']]

elif problem_num == 2:

    X = data[['Sample Question Papers Practiced']]

else:

    raise ValueError("Invalid problem number. Must be 1 or 2.")

y = data['Performance Index']

```

```

for method_num in method_nums:

    if method_num == 1:

        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

        model = LinearRegression()

        model.fit(X_train, y_train)

        y_pred = model.predict(X_test)

    elif method_num == 2:

        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

        X_train_transformed = np.sqrt(X_train) # Simple power transformation

        X_test_transformed = np.sqrt(X_test)

        model = LinearRegression()

        model.fit(X_train_transformed, y_train)

        y_pred = model.predict(X_test_transformed)

    elif method_num == 3:

        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

        X_train_transformed = np.log(X_train.replace(0, 1e-10)) # Handle invalid values in X

        X_test_transformed = np.log(X_test.replace(0, 1e-10))

        model = LinearRegression()

        model.fit(X_train_transformed, y_train)

        y_pred = model.predict(X_test_transformed)

    else:

        raise ValueError("Invalid method number. Must be 1, 2, or 3.")

```

Pada bagian ini, regresi dilakukan untuk setiap metode yang telah ditentukan sebelumnya. Pertama, variabel X dan y dipilih berdasarkan problem yang akan dikerjakan. Selanjutnya, untuk setiap metode regresi, model regresi diterapkan pada data menggunakan library Scikit-learn. Untuk metode regresi eksponensial, nilai-nilai yang tidak valid dalam X (nilai nol) diatasi

terlebih dahulu dengan menggantinya dengan nilai yang sangat kecil sebelum melakukan transformasi logaritma.

4. Visualisasi Hasil:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(12, 6))

for method_num, result in results.items():

    plt.scatter(result['y_test'], result['y_pred'], label=f'Method {method_num}')

plt.xlabel('Actual Performance Index')

plt.ylabel('Predicted Performance Index')

plt.title('Actual vs Predicted Performance Index')

plt.legend()

plt.grid(True)

plt.show()
```

Pada bagian ini, hasil regresi divisualisasikan dalam plot yang menunjukkan hubungan antara nilai ujian yang diprediksi dan nilai ujian sebenarnya. Setiap titik pada plot mewakili satu data ujian siswa, sementara garis regresi menunjukkan prediksi dari model regresi. Grafik divisualisasikan dan ditampilkan kepada pengguna.

5. Tampilkan RMSE:

```
python

for method_num, rmse in rmse_values.items():

    print(f'Method {method_num} RMSE: {rmse}')
```

Pada bagian terakhir, program menampilkan nilai Root Mean Squared Error (RMSE) untuk setiap metode regresi yang digunakan. RMSE digunakan sebagai metrik evaluasi untuk menilai keakuratan model regresi. Semakin kecil nilai RMSE, semakin baik kinerja model.