Implementasi Regresi Linear dan Pangkat Sederhana pada Python

Nama : Muhammad Nio Hastungkoro
NIM : 21120122140155
Matkul : Metode Numerik (A)

1. Ringkasan

Dokumen ini menjelaskan dua metode regresi yang digunakan untuk memodelkan hubungan antara waktu belajar dan nilai ujian mahasiswa dengan data dari https://www.kaggle.com/datasets/nikhil7280/student-performance-multiple-linear-regression. Metode pertama adalah regresi linear sederhana, sedangkan metode kedua adalah regresi pangkat sederhana. Pada dokumen ini, akan dibahas konsep masing-masing metode, implementasi kode, hasil pengujian, dan analisis hasilnya.

2. Konsep

a. Regresi Linear

Regresi linear adalah metode statistik yang digunakan untuk memodelkan hubungan antara variabel dependen (y) dan satu atau lebih variabel independen (X). Tujuannya adalah untuk menemukan garis yang paling sesuai dengan titik data. Rumus dasar untuk regresi linear adalah y = mx +c.

- y adalah variabel dependen (nilai ujian).
- x adalah variabel independen (waktu belajar).
- m adalah kemiringan garis.
- c adalah intercept (titik di mana garis memotong sumbu y).

b. Regresi Pangkat

Regresi pangkat adalah metode statistik di mana hubungan antara variabel independen dan dependen dimodelkan sebagai fungsi pangkat. Ini berguna untuk memodelkan hubungan non-linear. Rumus dasar untuk regresi pangkat adalah: y=Cx^b.

- y adalah variabel dependen (nilai ujian).
- x adalah variabel independen (waktu belajar).
- C dan bbb adalah parameter yang harus ditentukan dari data.

3. Implementasi Kode

a. Regresi Linear

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Membaca data CSV dan mengambil kolom "Hours Studied" dan "Performance
Index"
data = pd.read_csv('Student_Performance.csv')
X = data[['Hours Studied']]
y = data['Performance Index']

# Membuat model regresi linear
linear_model = LinearRegression()
linear_model.fit(X, y)

# Memprediksi menggunakan model regresi linear
y_predict_linear = linear_model.predict(X)

# Menghitung galat RMS untuk model regresi linear
rms_linear = np.sqrt(mean_squared_error(y, y_predict_linear))

# Plot grafik titik data dan hasil regresi linear
plt.figure(figsize=(10, 6))
plt.scatter(X, y, color='blue', label='Titik Data', s=1)
plt.plot(X, y_predict_linear, color='red', label= 'Hasil Regresi Linear')
plt.xlabel('Waktu Belajar (x)')
plt.ylabel('Nilai Ujian (y)')
plt.suptitle('Hasil Regresi Linear', fontsize=16)
plt.title(f'Galat RMS: {rms_linear}', fontsize=12, x=0.15)
plt.legend()
plt.show()

print(f'Galat RMS: {rms_linear}')
```

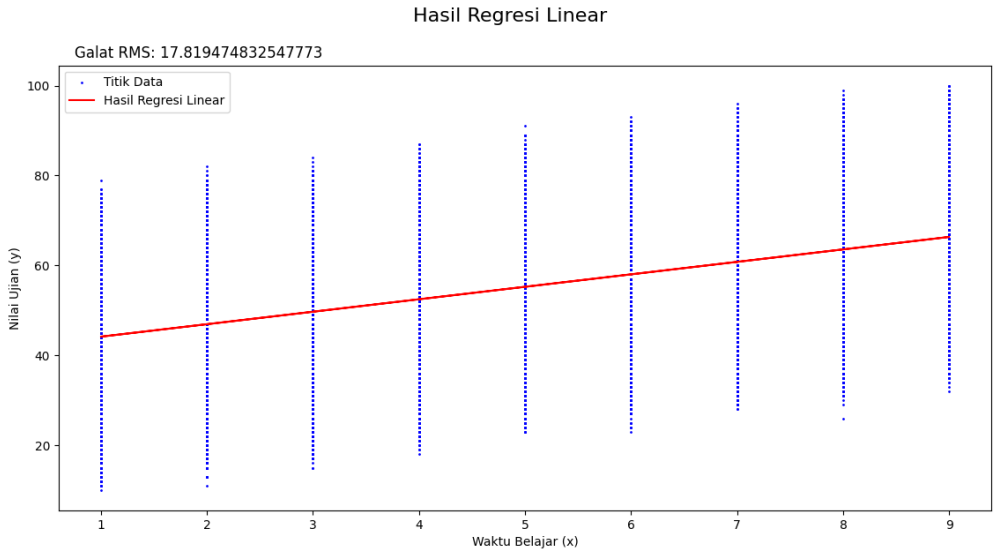
b. Regresi Pangkat Sederhana

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from sklearn.metrics import mean_squared_error

# Membaca data CSV dan mengambil kolom "Hours Studied" dan "Performance
Index"
data = pd.read_csv('Student_Performance.csv')
X = data['Hours Studied']
y = data['Performance Index']

# Urutkan data berdasarkan nilai X
sorted_indices = np.argsort(X)
X = X[sorted_indices]
y = y[sorted_indices]

# Definisikan model pangkat
def power_law(x, C, b):
    return C * np.power(x, b)

# Gunakan curve_fit untuk menemukan parameter a dan b
params, covariance_matrix = curve_fit(power_law, X, y)    # variabel
"covariance_matrix" tidak digunakan
C, b = params

# Prediksi menggunakan model pangkat
y_predict = power_law(X, C, b)

# Menghitung galat RMS
rms_error = np.sqrt(mean_squared_error(y, y_predict))

# Plot hasil
plt.figure(figsize=(10, 6))
plt.scatter(X, y, color='blue', label='Titik Data', s=1)
plt.plot(X, y_predict, color='red', label= 'Hasil Regresi Pangkat
Sederhana')
plt.xlabel('Waktu Belajar (x)')
plt.ylabel('Nilai Ujian (y)')
```

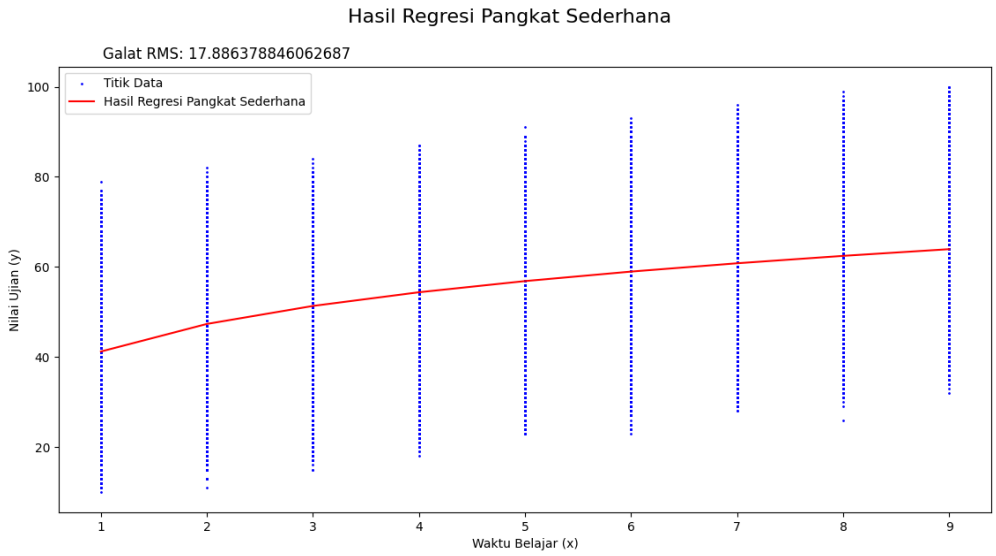
```
plt.suptitle('Hasil Regresi Pangkat Sederhana', fontsize=16)
plt.title(f'Galat RMS: {rms_error}', fontsize=12, x=0.18)
plt.legend()
plt.show()

print(f'Parameter C: {C}')
print(f'Parameter b: {b}')
print(f'Galat RMS: {rms_error}')
```

4. Hasil Pengujian

a. Regresi Linear



b. Regresi Pangkat Sederhana



5. Analisis Hasil

a. Regresi Linear

Regresi linear adalah salah satu metode paling sederhana dan umum digunakan dalam analisis regresi. Model ini bekerja dengan baik untuk data yang menunjukkan hubungan linear, yaitu ketika kenaikan atau penurunan pada variabel independen (X) berbanding lurus dengan kenaikan atau penurunan pada variabel dependen (y).

Pada kasus ini, model regresi linear memodelkan hubungan antara waktu belajar dan nilai ujian. Implementasi kode menunjukkan proses berikut:

1. Mengambil dan membaca data dari file CSV dan kolom "Hours Studied" serta "Performance Index".
2. Membuat Model regresi linear dibuat menggunakan `LinearRegression` dari `sklearn`.
3. Melatih Model dilatih menggunakan data.
4. Menggunakan Model untuk memprediksi nilai ujian berdasarkan waktu belajar.
5. Menghitung Galat RMS (Root Mean Square Error) dihitung untuk menilai performa model.
6. Menggambar Grafik scatter plot dari titik data dan garis hasil regresi linear.

Hasil dari regresi linear menunjukkan garis lurus yang mencoba untuk meminimalkan jarak vertikal antara titik data aktual dan garis regresi.

Kelebihan:

- Sederhana dan mudah diimplementasikan.
- Mudah diinterpretasikan.
- Memberikan estimasi yang baik jika data memiliki hubungan linear.

Kekurangan:

- Tidak cocok untuk data yang menunjukkan hubungan non-linear.
- Rentan terhadap outlier yang bisa mempengaruhi kemiringan garis secara signifikan.

Hasil dari regresi linear di kasus ini menunjukkan nilai galat RMS yang digunakan sebagai metrik untuk mengevaluasi seberapa baik model memprediksi data. Semakin kecil nilai galat RMS, semakin baik model dalam memprediksi data. Namun, dari hasil galat RMS yang didapat menunjukkan bahwa data yang digunakan terlalu tersebar rata sehingga banyak titik data yang tidak dilalui garis linear.

‘

b. Regresi Pangkat Sederhana

Regresi pangkat digunakan ketika hubungan antara variabel independen dan dependen lebih kompleks dan menunjukkan pola non-linear. Pada kasus ini, model pangkat digunakan untuk memodelkan hubungan antara waktu belajar dan nilai ujian.

Implementasi kode menunjukkan proses berikut:

1. Memngambil dan membaca data dari file CSV dan kolom "Hours Studied" serta "Performance Index".
2. Data diurutkan berdasarkan nilai "Hours Studied".
3. Model pangkat didefinisikan dengan rumus y=Cxby = C x^by=Cxb.
4. Gunakan `curve_fit` dari `scipy.optimize` untuk menemukan parameter C dan b.
5. Model digunakan untuk memprediksi nilai ujian berdasarkan waktu belajar.
6. Menghitung Galat RMS untuk menilai performa model.
7. Menggambar Grafik scatter plot dari titik data dan kurva hasil regresi pangkat sederhana

Hasil dari regresi pangkat menunjukkan kurva yang mencoba untuk meminimalkan jarak vertikal antara titik data aktual dan kurva regresi.

Kelebihan:

- Lebih fleksibel dan dapat memodelkan hubungan non-linear.
- Dapat menangkap pola yang lebih kompleks dalam data.

Kekurangan:

- Lebih kompleks dibandingkan regresi linear.
- Memerlukan lebih banyak usaha dalam interpretasi parameter model.
- Sensitif terhadap pemilihan model yang tepat.

Hasil dari regresi pangkat di kasus ini menunjukkan nilai galat RMS yang digunakan sebagai metrik untuk mengevaluasi seberapa baik model memprediksi data. Semakin kecil nilai galat RMS, semakin baik model dalam memprediksi data. Selain itu, parameter C dan b memberikan informasi tambahan tentang skala dan bentuk hubungan non-linear antara waktu belajar dan nilai ujian. Namun, dari hasil galat RMS yang didapat menunjukkan bahwa data yang digunakan terlalu tersebar rata sehingga banyak titik data yang tidak dilalui garis regresi pangkat sederhana.