

TUGAS METODE NUMERIK

APLIKASI REGRESI UNTUK PROBLEM SOLVING

Nama : Leonardus Deni Prabowo

NIM : 21120122120012

Metode Numerik - Kelas A

Soal

Diinginkan untuk mencari hubungan faktor yang mempengaruhi nilai ujian siswa (NT):

1. Durasi waktu belajar (TB) terhadap nilai ujian (Problem 1)
2. Jumlah latihan soal (NL) terhadap nilai ujian (Problem 2)

Data TB, NL, dan NT diperoleh dari <https://www.kaggle.com/datasets/nikhil7280/student-performance-multiple-linear-regression>, yaitu kolom *Hours Studied*, *Sample Question Papers Practiced*, dan *Performance Index*.

Implementasikan regresi untuk mencari hubungan tersebut menggunakan metode:

1. Model linear (Metode 1)
2. Model pangkat sederhana (Metode 2)
3. Model eksponensial (Metode 3)
4. Model lainnya di halaman 24 slide materi (Metode opsional)

Tugas Mahasiswa

1. Mahasiswa membuat kode sumber dengan bahasa pemrograman yang dikuasai untuk mengimplementasikan solusi di atas, dengan ketentuan:
 - NIM terakhir ganjil mengerjakan Problem 1 dengan Metode 1 dan Metode 2
 - NIM terakhir genap mengerjakan Problem 2 dengan Metode 1 dan Metode 3
 - Mahasiswa juga bisa menambah solusi dengan salah satu metode opsional
2. Sertakan kode testing untuk menguji kode sumber tersebut untuk menyelesaikan problem dalam gambar. Plot grafik titik data dan hasil regresinya masing-masing
3. Hitung galat RMS dari tiap metode yang digunakan
4. Mengunggah kode sumber tersebut ke Github dan setel sebagai publik. Berikan deskripsi yang memadai dari project tersebut. Masukkan juga dataset dan data hasil di repositori tersebut.

5. Buat dokumen docx dan pdf yang menjelaskan alur kode dari (1), analisis hasil, dan penjabarannya. Sistematika dokumen: Ringkasan, Konsep, Implementasi Kode, Hasil Pengujian, dan Analisis Hasil.

Source Code

Berikut ini merupakan source code lengkap. Dan di bawahnya merupakan penjelasan dari setiap bagian dari source code yang ada. Source code dibuat dengan menggunakan bahasa Python.

```
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
from scipy.optimize import curve_fit

# Nama : Leonardus Deni Prabowo
# NIM : 21120122120012
# Metode Numerik - Kelas A

# Pembaca File Yang Diunggah
file_path = 'D:\#Kuliah\Metode Numerik\Student_Performance.csv'
data = pd.read_csv(file_path)

# Menampilkan Nama Kolom Yang Sesuai
print(data.columns)
print(data.head())

# Memilih Kolom Yang Relevan Dengan Nama Yang Sesuai
NL = data['Sample Question Papers Practiced'].values
NT = data['Performance Index'].values

# Model Linear Untuk Problem 2
X_NL = NL.reshape(-1, 1)
linear_model_NL = LinearRegression()
linear_model_NL.fit(X_NL, NT)
NT_pred_linear_NL = linear_model_NL.predict(X_NL)

# Hasil Dengan Regresi Linear
plt.figure(figsize=(12, 6))
plt.scatter(NL, NT, color='red', label='Data Asli')
plt.plot(NL, NT_pred_linear_NL, color='blue', label='Regresi Linear')
plt.title('Regresi Linear (Jumlah Latihan Soal vs Nilai Ujian)')
plt.xlabel('Jumlah Latihan Soal (NL)')
plt.ylabel('Nilai Ujian (NT)')
plt.legend()
plt.grid(True)
plt.show()

# Menghitung RMS Model Linear (NL)
rms_linear_NL = np.sqrt(mean_squared_error(NT, NT_pred_linear_NL))
print(f'RMS untuk model linear (NL): {rms_linear_NL}')

# Model Eksponensial Untuk Problem 2
def exp_func(x, a, b, c):
    return a * np.exp(b * x) + c
```

```

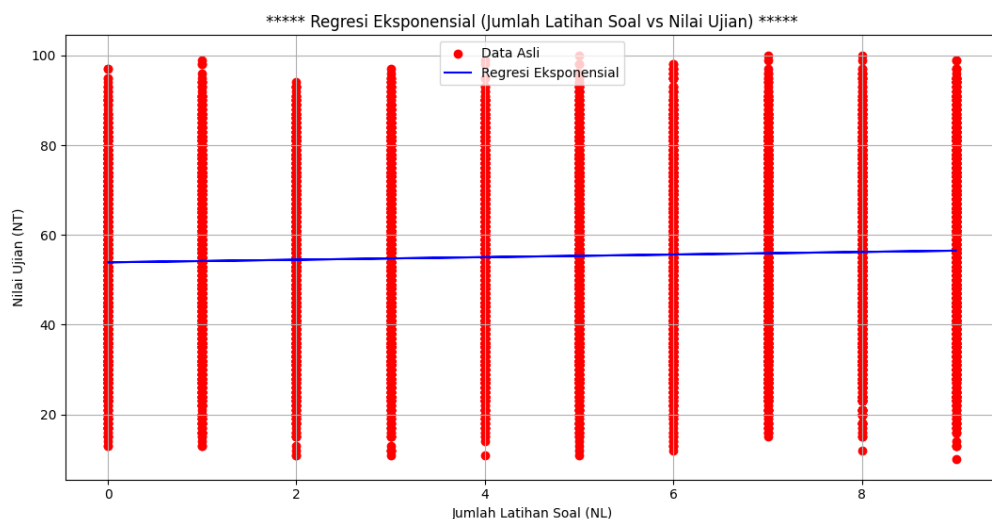
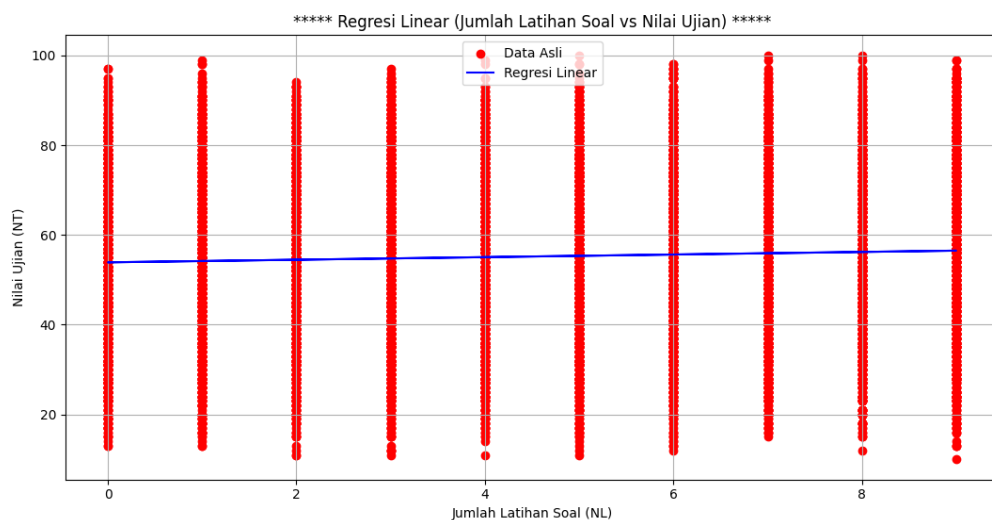
popt, _ = curve_fit(exp_func, NL, NT)
NT_pred_exp_NL = exp_func(NL, *popt)

# Hasil Dengan Regresi Eksponensial
plt.figure(figsize=(12, 6))
plt.scatter(NL, NT, color='red', label='Data Asli')
plt.plot(NL, NT_pred_exp_NL, color='blue', label='Regresi Eksponensial')
plt.title('Regresi Eksponensial (Jumlah Latihan Soal vs Nilai Ujian)')
plt.xlabel('Jumlah Latihan Soal (NL)')
plt.ylabel('Nilai Ujian (NT)')
plt.legend()
plt.grid(True)
plt.show()

# Menghitung RMS Model Eksponensial (NL)
rms_exp_NL = np.sqrt(mean_squared_error(NT, NT_pred_exp_NL))
print(f'RMS untuk model eksponensial (NL): {rms_exp_NL}')

```

Hasil Running



Analisis Program

Berikut ini merupakan analisis dari alur program yang dibuat. Analisis berupa penjelasan tiap-tiap bagian.

```
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
from scipy.optimize import curve_fit

# Nama : Leonardus Deni Prabowo
# NIM : 21120122120012
# Metode Numerik - Kelas A

# Pembaca File Yang Diunggah
file_path = 'D:\#Kuliah\Metode Numerik\Student_Performance.csv'
data = pd.read_csv(file_path)

# Menampilkan Nama Kolom Yang Sesuai
print(data.columns)
print(data.head())
```

Pada bagian di atas, merupakan bagian yang berfungsi untuk mengimpor library yang diperlukan seperti pandas yang berfungsi untuk pengolahan data yang dimasukkan, numpy untuk operasi numerik, matplotlib.pyplot untuk visualisasi, LinearRegression dari sklearn yang berfungsi untuk model regresi linear, mean_squared_error untuk melakukan pengukuran kesalahan prediksi, dan curve_fit dari scipy untuk melakukan regresi eksponensial. Data dibaca dari file .csv yang dimasukkan. File .csv dimasukkan menggunakan pd.read_csv dan nama kolom serta beberapa baris pertama dari data akan ditampilkan untuk proses verifikasi.

```
# Memilih Kolom Yang Relevan Dengan Nama Yang Sesuai
NL = data['Sample Question Papers Practiced'].values
NT = data['Performance Index'].values
```

Pada bagian ini, kita akan melakukan pemilihan kolom yang relevan dari data yaitu 'Sample Question Papers Practiced' yang akan mewakili jumlah dari latihan soal (NL) dan 'Performance Index' yang akan mewakili nilai dari ujian siswa (NT). Kolom ini kemudian akan dikonversi menjadi array numpy untuk mempermudah operasi selanjutnya.

```
# Model Linear Untuk Problem 2
X_NL = NL.reshape(-1, 1)
linear_model_NL = LinearRegression()
linear_model_NL.fit(X_NL, NT)
NT_pred_linear_NL = linear_model_NL.predict(X_NL)
```

Di bagian ini akan dibangun model regresi linear yang berfungsi untuk memprediksi nilai ujian berdasarkan jumlah latihan soal. Data NL akan diubah bentuk menjadi array 2D agar dapat sesuai dengan syarat masukan LinearRegression. Model regresi linear lalu diinisialisasi dan dilatih memakai data dari NL dan NT. Prediksi nilai ujian akan dibuat memakai model yang sudah dilatih.

```
# Hasil Dengan Regresi Linear
plt.figure(figsize=(12, 6))
plt.scatter(NL, NT, color='red', label='Data Asli')
plt.plot(NL, NT_pred_linear_NL, color='blue', label='Regresi Linear')
plt.title('Regresi Linear (Jumlah Latihan Soal vs Nilai Ujian)')
plt.xlabel('Jumlah Latihan Soal (NL)')
plt.ylabel('Nilai Ujian (NT)')
plt.legend()
plt.grid(True)
plt.show()
```

Pada bagian ini akan dibuat sebuah plot yang menghasilkan regresi linear. Scatter plot dibuat untuk menunjukkan data asli antara NL dan NT, dan garis biru menunjukkan sebuah prediksi dari model regresi linear. Grafik akan diberi judul dan juga label pada sumbu X dan Y nya untuk memberikan konteks yang lebih memperjelas grafiknya.

```
# Menghitung RMS Model Linear (NL)
rms_linear_NL = np.sqrt(mean_squared_error(NT, NT_pred_linear_NL))
print(f'RMS untuk model linear (NL): {rms_linear_NL}')
```

Di bagian ini, Root Mean Square (RMS) error akan dihitung untuk model dari regresi linear. RMKS error yang dipakai akan mengukur seberapa baik model ketika memprediksi nilai dari ujian siswa. Hasil dari RMS error akan dicetak untuk menganalisis lebih lanjut, dengan nilai RMS yang lebih rendah, menandakan model yang ada memiliki nilai keakuratan yang lebih baik.

```
# Model Eksponensial Untuk Problem 2
def exp_func(x, a, b, c):
    return a * np.exp(b * x) + c

popt, _ = curve_fit(exp_func, NL, NT)
NT_pred_exp_NL = exp_func(NL, *popt)
```

Pada bagian ini, akan dibangun model regresi eksponensial yang berfungsi untuk memprediksi nilai ujian berdasarkan jumlah latihan soal. Fungsi dari eksponensial didefinisi dengan $\text{exp_func}(x, a, b, c)$. Parameter yang ada untuk fungsi tersebut, diperkirakan memakai `curve_fit`, dan hasil dari prediksi akan dibuat memakai parameter optimal yang sudah ditemukan.

```
# Hasil Dengan Regresi Eksponensial
plt.figure(figsize=(12, 6))
plt.scatter(NL, NT, color='red', label='Data Asli')
plt.plot(NL, NT_pred_exp_NL, color='blue', label='Regresi Eksponensial')
plt.title('Regresi Eksponensial (Jumlah Latihan Soal vs Nilai Ujian)')
plt.xlabel('Jumlah Latihan Soal (NL)')
plt.ylabel('Nilai Ujian (NT)')
plt.legend()
plt.grid(True)
plt.show()
```

Di bagian ini akan membuat sebuah plot hasil dari regresi eksponensial. Seperti pada sebelumnya, yaitu regresi linear, scatter plot akan menunjukkan data hasil asli dan garis biru

akan menunjukkan prediksi dari model regresi eksponensial. Grafik akan diberi judul dan label pada bagian sumbu X dan Y untuk memberikan konteks yang lebih jelas.

```
# Menghitung RMS Model Eksponensial (NL)
rms_exp_NL = np.sqrt(mean_squared_error(NT, NT_pred_exp_NL))
print(f'RMS untuk model eksponensial (NL): {rms_exp_NL}')
```

Di bagian ini, RMS error juga akan dihitung untuk regresi eksponensial. RMS error akan digunakan untuk mengevaluasi seberapa baik model dapat memprediksi nilai ujian dari siswa. Hasil RMS error kemudian akan dicetak untuk melakukan analisis yang lebih lanjut, dengan nilai RMS yang lebih rendah, maka akan menunjukkan model yang lebih akurat.

Kode ini secara komprehensif akan melakukan analisis terhadap hubungan antara jumlah latihan soal dan nilai ujian siswa memakai dua buah metode regresi yang berbeda, yaitu regresi linear dan regresi eksponensial. Regresi linear akan memberikan model sederhana yang akan mudah untuk diinterpretasi, sementara regresi eksponensial menawarkan model yang kompleks dan lebih sesuai dengan data yang menunjukkan pola pertumbuhan eksponensial. Dengan menghitung dan membandingkan RMS error, akan bisa diketahui model mana yang lebih akurat dalam memprediksi nilai ujian berdasarkan jumlah latihan soal.

Link Github Project

<https://github.com/LeonDeniP/Tugas-Metnum-Aplikasi-Regresi-Leonardus-Deni-Prabowo-21120122120012>