

Nama : Gina Nabila

NIM : 21120122130055

Mata Kuliah : Metode Numerik – Kelas D

Github Link :

https://github.com/gin-na/Aplikasi-Regresi_Metode-Numerik_Gina-Nabila

Aplikasi Regresi Metode Linear dan Metode Eksponensial

1. Metode Linear

1.1 Ringkasan Regresi Metode Linear

Analisis regresi linear adalah sebuah metode pendekatan untuk pemodelan hubungan antara satu variabel dependen dan satu variabel independen. Dalam regresi, variabel independen menerangkan variabel dependennya. Dalam analisis regresi sederhana, hubungan antar variabel bersifat linear, dimana perubahan pada variabel X akan diikuti oleh perubahan Y secara tetap. Sementara pada hubungan non-linear, perubahan variabel X tidak diikuti variabel Y secara proposional.

$$Y = a + bx$$

Dimana:

Y = variabel

a = konstanta

b = koefisien regresi

Persamaan untuk mendapatkan nilai a dan b adalah

$$\begin{bmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \end{bmatrix}$$

$$b = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$a = \bar{y} - b\bar{x}$$

Diinginkan implementasi regresi linear untuk mencari hubungan durasi waktu belajar (TB) terhadap nilai ujian siswa (NT).

1.2 Implementasi Kode

```
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

# Memuat data dari file CSV
file_path = '/content/drive/MyDrive/Colab
Notebooks/Student_Performance.csv'
data = pd.read_csv(file_path)

# Mempersiapkan data
# Menggunakan kolom 'hours studied' sebagai durasi belajar
(TB) dan 'performance index' sebagai nilai ujian (NT)
data = data.dropna() # Menghapus baris yang memiliki nilai
NaN
X = data[['Hours Studied']].values # Mengambil kolom hours
studied sebagai variabel independen
y = data['Performance Index'].values # Mengambil kolom
performance index sebagai variabel dependen

# Melakukan regresi linear menggunakan Scikit-Learn
model = LinearRegression()
model.fit(X, y)
y_pred = model.predict(X)

# Membuat plot hasil regresi
plt.scatter(X, y, color='royalblue', label='Data Sebenarnya')
plt.plot(X, y_pred, color='hotpink', linewidth=2,
label='Garis Regresi')
plt.xlabel('Waktu Belajar')
plt.ylabel('Nilai Ujian')
plt.title('Hubungan waktu belajar terhadap nilai ujian')
plt.legend()
plt.show()
```

```
# Menghitung dan menampilkan nilai galat RMS (Root Mean
Square Error)

rms_error = np.sqrt(mean_squared_error(y, y_pred))

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

Penjelasan alur kode yaitu sebagai berikut:

- Mengambil data dari file .csv

```
# Memuat data dari file CSV

file_path = '/content/drive/MyDrive/Colab
Notebooks/Student_Performance.csv'

data = pd.read_csv(file_path)
```

Data yang dimasukkan berasal dari website Kaggle: <https://www.kaggle.com/datasets/nikhil7280/student-performance-multiple-linear-regression?resource=download> Dataset berisi 10.000 data siswa dengan informasi tentang berbagai predictor dan *performance index*.

- Memilah kolom yang akan digunakan

```
# Mempersiapkan data

# Menggunakan kolom 'hours studied' sebagai durasi belajar
(TB) dan 'performance index' sebagai nilai ujian (NT)

data = data.dropna() # Menghapus baris yang memiliki nilai
NaN

X = data[['Hours Studied']].values # Mengambil kolom hours
studied sebagai variabel independen

y = data['Performance Index'].values # Mengambil kolom
performance index sebagai variabel dependen
```

Data yang akan digunakan yaitu data 'Hours Studied' dan 'Performance Index' karena hubungan yang dicari ialah hubungan antara waktu belajar terhadap nilai siswa.

- Melakukan regresi linear

```
# Melakukan regresi linear menggunakan Scikit-Learn

model = LinearRegression()

model.fit(X, y)

y_pred = model.predict(X)
```

Model regresi linear didefinisikan dengan `LinearRegression` yang diambil dari Scikit-Learn. Model ini kemudian dilatih menggunakan data yang telah disiapkan, di mana variabel independen (X) adalah durasi waktu belajar (Hours

Studied) dan variabel dependen (y) adalah nilai ujian (Performance Index). Proses pelatihan dilakukan dengan memanggil metode `fit(X, y)` pada model, yang menyesuaikan garis regresi terbaik ke data. Setelah model dilatih, prediksi nilai ujian dihasilkan untuk setiap nilai durasi waktu belajar dengan memanggil metode `predict(X)`, menghasilkan `y_pred` yang merupakan nilai prediksi dari model.

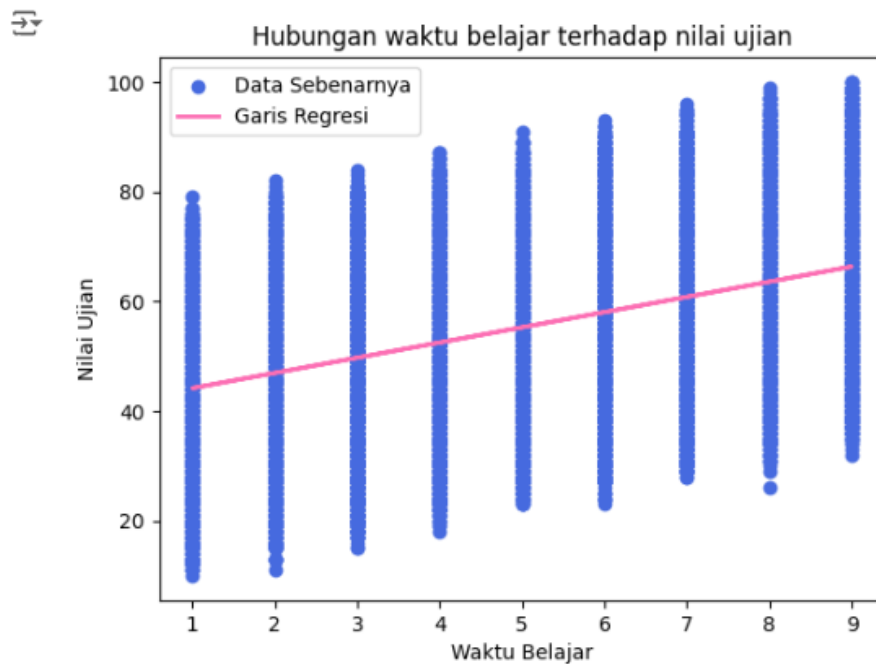
- Mencetak hasil menjadi sebuah plot

```
plt.scatter(X, y, color='royalblue', label='Data Sebenarnya')
plt.plot(X, y_pred, color='hotpink', linewidth=2,
label='Garis Regresi')
plt.xlabel('Waktu Belajar')
plt.ylabel('Nilai Ujian')
plt.title('Hubungan waktu belajar terhadap nilai ujian')
plt.legend()
plt.show()

# Menghitung dan menampilkan nilai galat RMS (Root Mean
Square Error)
rms_error = np.sqrt(mean_squared_error(y, y_pred))
print(f'Galat RMS: {rms_error}')
```

Hasil kemudian akan dibuat menjadi sebuah plot dengan kode di atas. Nilai galat RMS juga dicetak dengan `print(f'Galat RMS: {rms_error}')` yang menunjukkan seberapa besar kesalahan prediksi rata-rata dari model regresi linear.

1.3 Hasil Pengujian



Galat RMS: 17.819474832547773

1.4 Analisis Hasil

Plot hasil regresi menunjukkan bahwa ada hubungan positif antara waktu belajar dan nilai ujian, yang berarti semakin banyak waktu yang dihabiskan untuk belajar, semakin tinggi nilai ujian yang diperoleh. Garis regresi menggambarkan tren umum dari data ini.

Galat RMS yang dihasilkan bernilai 17.819474832547773. Ini mengindikasikan rata-rata kesalahan prediksi model regresi linear dalam satuan yang sama dengan nilai ujian. Nilai galat RMS ini memperlihatkan sejauh mana model prediksi menyimpang dari nilai sebenarnya. Jika nilai ujian berada pada skala 0-100, galat RMS senilai 17.819474832547773 dikategorikan cukup besar. Hal ini menunjukkan bahwa meskipun ada tren positif, prediksi model masih memiliki variasi yang cukup besar dari data sebenarnya. Hal ini bisa disebabkan oleh faktor-faktor lain yang mempengaruhi nilai ujian yang tidak terukur dalam variabel waktu belajar saja.

2. Metode Eksponensial

2.1 Ringkasan Regresi Metode Eksponensial

Regresi eksponensial adalah metode statistik yang digunakan untuk memodelkan hubungan antara variabel independen dan dependen yang bersifat eksponensial. Dalam regresi eksponensial, hubungan antara variabel independen (x) dan variabel dependen (y) dimodelkan dengan fungsi eksponensial, yaitu $y = a \cdot e^{bx}$, di mana a adalah konstanta, b adalah koefisien yang mengontrol tingkat pertumbuhan atau penurunan, dan e adalah basis logaritma natural.

2.2 Implementasi Kode

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

# Memuat data dari file CSV
file_path = '/content/drive/MyDrive/Colab
Notebooks/Student_Performance.csv'
data = pd.read_csv(file_path)

# Mempersiapkan data
data = data.dropna() # Menghapus baris yang memiliki nilai
NaN
X = data['Hours Studied'].values # Mengambil kolom hours
studied sebagai variabel independen
y = data['Performance Index'].values # Mengambil kolom
performance index sebagai variabel dependen

# Mendefinisikan fungsi eksponensial
def exponential_model(x, a, b):
    return a * np.exp(b * x)

# Melakukan regresi eksponensial menggunakan curve_fit dari
scipy
```

```

popt, pcov = curve_fit(exponential_model, X, y)
y_pred = exponential_model(X, *popt)

# Membuat plot hasil regresi
plt.scatter(X, y, color='royalblue', label='Data Sebenarnya')
plt.plot(X, y_pred, color='hotpink', linewidth=2,
label='Garis Regresi Eksponensial')
plt.xlabel('Waktu Belajar')
plt.ylabel('Nilai Ujian')
plt.title('Hubungan Waktu Belajar terhadap Nilai Ujian')
plt.legend()
plt.show()

# Menghitung dan menampilkan nilai galat RMS (Root Mean
Square Error)
rms_error = np.sqrt(mean_squared_error(y, y_pred))
print(f'Galat RMS: {rms_error}')

```

Penjelasan alur kode yaitu sebagai berikut:

- Mengambil data dari file .csv

```

# Memuat data dari file CSV
file_path = '/content/drive/MyDrive/Colab
Notebooks/Student_Performance.csv'
data = pd.read_csv(file_path)

```

Data yang dimasukkan berasal dari website Kaggle: <https://www.kaggle.com/datasets/nikhil7280/student-performance-multiple-linear-regression?resource=download> Dataset berisi 10.000 data siswa dengan informasi tentang berbagai predictor dan *performance index*.

- Memilah kolom yang akan digunakan

```

# Mempersiapkan data
data = data.dropna() # Menghapus baris yang memiliki nilai
NaN
X = data['Hours Studied'].values # Mengambil kolom hours
studied sebagai variabel independen
y = data['Performance Index'].values # Mengambil kolom
performance index sebagai variabel dependen

```

Data yang akan digunakan yaitu data 'Hours Studied' dan 'Performance Index' karena hubungan yang dicari ialah hubungan antara waktu belajar terhadap nilai siswa.

- Melakukan regresi eksponensial

```
# Mendefinisikan fungsi eksponensial
def exponential_model(x, a, b):
    return a * np.exp(b * x)

# Melakukan regresi eksponensial menggunakan curve_fit dari
# scipy
popt, pcov = curve_fit(exponential_model, X, y)
y_pred = exponential_model(X, *popt)
```

Model eksponensial $y = a \cdot e^{bx}$ didefinisikan dengan fungsi `exponential_model`. Kemudian, fungsi `curve_fit` dari pustaka `scipy.optimize` digunakan untuk menyesuaikan model eksponensial ke data yang diberikan, ini menghasilkan parameter optimal `popt` yang meminimalkan perbedaan antara data sebenarnya dan model. Prediksi nilai y dihitung dengan fungsi model eksponensial menggunakan parameter yang telah dioptimalkan.

- Mencetak hasil menjadi sebuah plot

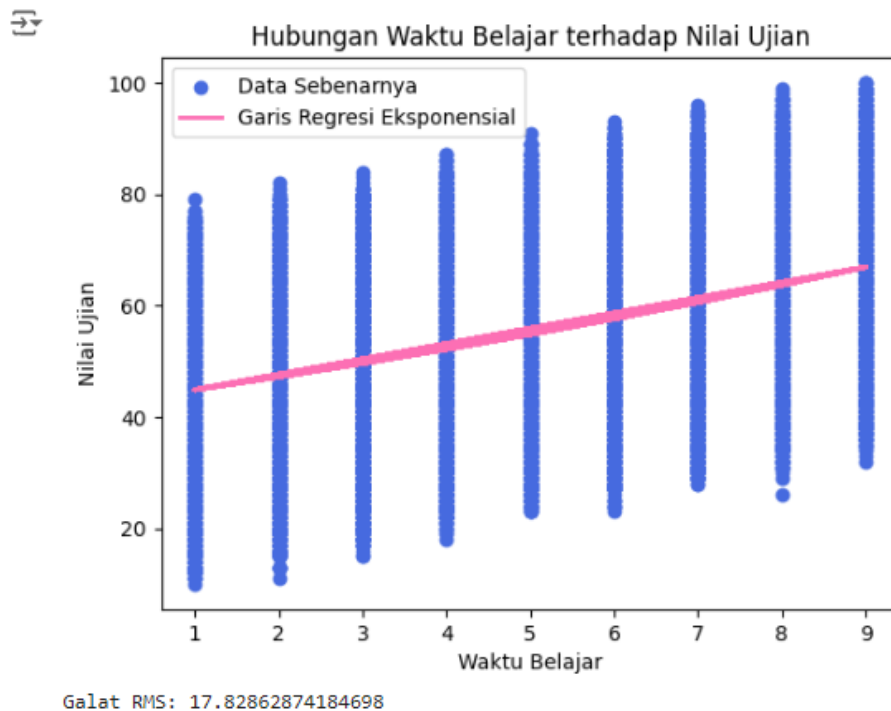
```
# Membuat plot hasil regresi
plt.scatter(X, y, color='royalblue', label='Data Sebenarnya')
plt.plot(X, y_pred, color='hotpink', linewidth=2,
         label='Garis Regresi Eksponensial')
plt.xlabel('Waktu Belajar')
plt.ylabel('Nilai Ujian')
plt.title('Hubungan Waktu Belajar terhadap Nilai Ujian')
plt.legend()
plt.show()

# Menghitung dan menampilkan nilai galat RMS (Root Mean
# Square Error)
rms_error = np.sqrt(mean_squared_error(y, y_pred))
print(f'Galat RMS: {rms_error}')
```

Hasil kemudian akan dibuat menjadi sebuah plot dengan kode di atas. Nilai galat RMS juga dicetak dengan `print(f'Galat RMS: {rms_error}')` yang

menunjukkan seberapa besar kesalahan prediksi rata-rata dari model regresi eksponensial.

2.3 Hasil Pengujian



2.4 Analisis Hasil

Plot menunjukkan tren umum di mana *performance index* atau nilai ujian cenderung meningkat seiring dengan bertambahnya jam belajar. Hal ini menunjukkan adanya korelasi positif antara waktu belajar dan nilai ujian. Garis regresi eksponensial yang diperoleh dari model menunjukkan bahwa hubungan tersebut dapat dimodelkan dengan fungsi eksponensial. Model eksponensial yang digunakan adalah $y = ae^{bx}$ di mana x adalah waktu belajar dan y adalah nilai ujian.

Galat RMS yang dihasilkan bernilai 17.82862874184698. Ini mengindikasikan rata-rata kesalahan prediksi model regresi linear dalam satuan yang sama dengan nilai ujian. Nilai galat RMS ini memperlihatkan sejauh mana model prediksi menyimpang dari nilai sebenarnya.