

PRAKTIKUM FISIKA KOMPUTASI

MODUL 10

MACHINE LEARNING REGRESI LINEAR DAN POLINOMIAL

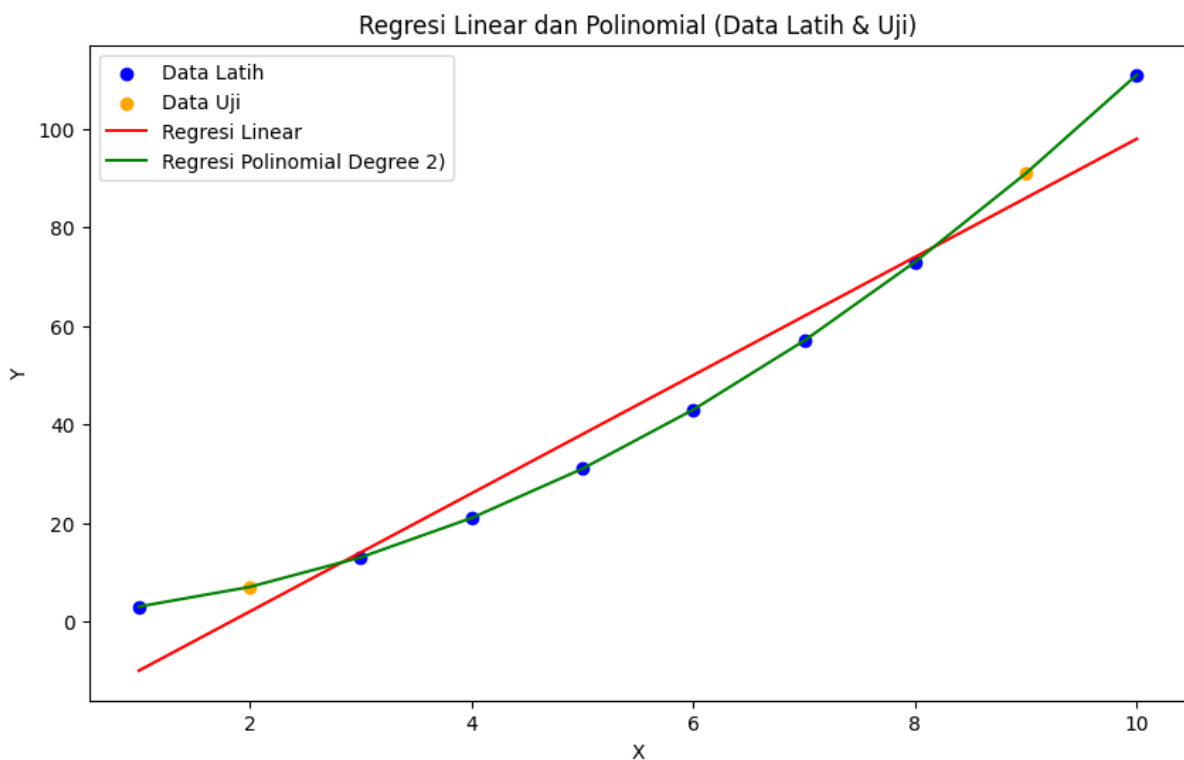
Nama : Muhammad Raza Naufal

NIM : 1227030023

- Mengganti dataset menjadi $X = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$; $Y = [3, 7, 13, 21, 31, 43, 57, 73, 91, 111]$ serta membagi dataset menjadi 80% data latih dan 20% data uji

```
X = [[1], [2], [3], [4], [5], [6], [7], [8], [9], [10]]
Y = [3, 7, 13, 21, 31, 43, 57, 73, 91, 111]
```

Hasil :



- Regresi linear dan regresi polynomial derajat 2 dan bandingkan Mean Squared Error (MSE)

```
Mean Squared Error (Linear): 25.00
Mean Squared Error (Polynomial Degree 2): 0.00
```

- **Penjelasan kode program dan gambar hasil**

- a. Kode Program**

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split

X = [[1], [2], [3], [4], [5], [6], [7], [8], [9], [10]]
Y = [3, 7, 13, 21, 31, 43, 57, 73, 91, 111]

X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
test_size=0.2, random_state=42)

linear_model = LinearRegression()
linear_model.fit(X_train, Y_train)

poly_features_2 = PolynomialFeatures(degree=2)
X_train_poly_2 = poly_features_2.fit_transform(X_train)
poly_model_2 = LinearRegression()
poly_model_2.fit(X_train_poly_2, Y_train)

X_sorted = np.sort(X, axis=0)
Y_pred_linear_all = linear_model.predict(X_sorted)
Y_pred_poly_2_all =
poly_model_2.predict(poly_features_2.transform(X_sorted))

mse_linear = mean_squared_error(Y_test, linear_model.predict(X_test))
mse_poly_2 = mean_squared_error(Y_test,
poly_model_2.predict(poly_features_2.transform(X_test)))

print(f"Mean Squared Error (Linear): {mse_linear:.2f}")
print(f"Mean Squared Error (Polynomial Degree 2): {mse_poly_2:.2f}")

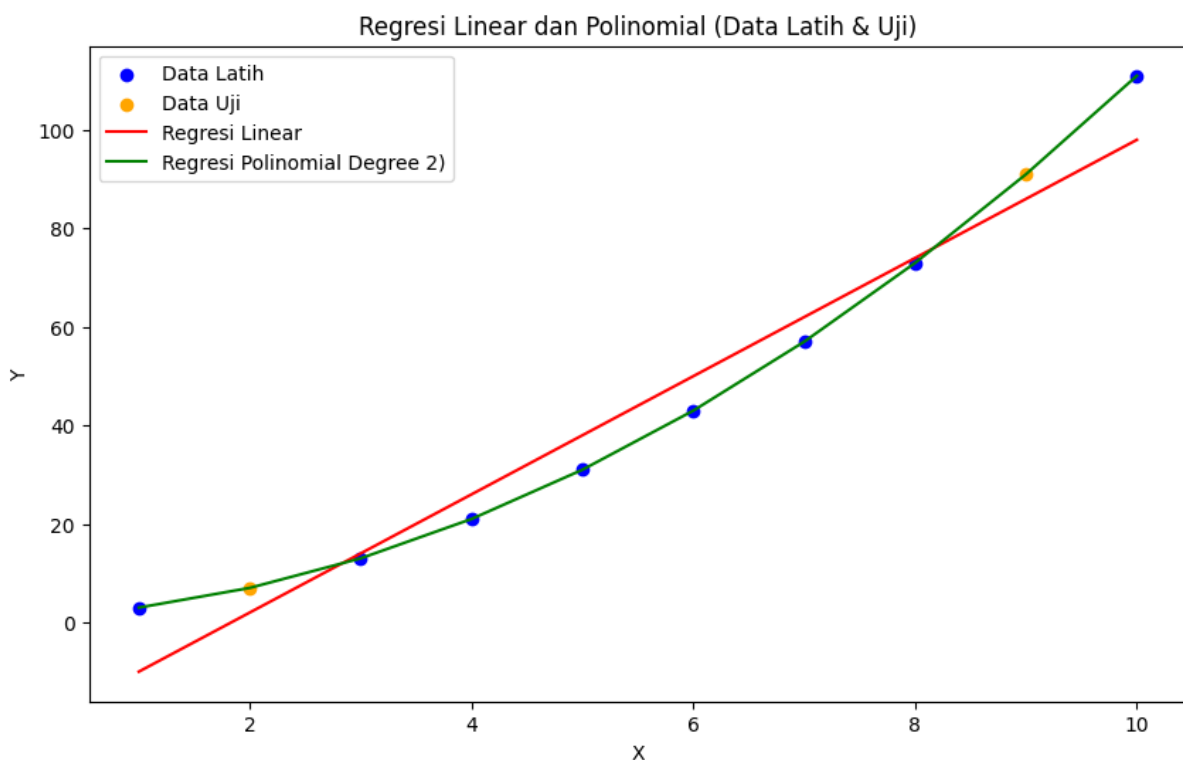
plt.figure(figsize=(10, 6))
plt.scatter(X_train, Y_train, color='blue', label='Data Latih')
plt.scatter(X_test, Y_test, color='orange', label='Data Uji')
plt.plot(X_sorted, Y_pred_linear_all, color='red', label='Regresi
Linear')
plt.plot(X_sorted, Y_pred_poly_2_all, color='green', label='Regresi
Polinomial Degree 2')

plt.xlabel('X')
plt.ylabel('Y')
plt.title('Regresi Linear dan Polinomial (Data Latih & Uji)')
plt.legend()
```

```
plt.show()
```

Kode program ini membandingkan kinerja regresi linear dan regresi polinomial derajat 2 dalam memprediksi hubungan antara dua variabel, X (input) dan Y (output). Data input yang berbentuk sederhana menunjukkan pola kuadratik antara X dan Y. Data tersebut dibagi menjadi data latih (80%) untuk melatih model dan data uji (20%) untuk mengevaluasi performa model menggunakan fungsi `train_test_split`. Model regresi linear dilatih dengan data latih untuk mencari hubungan linear terbaik antara X dan Y. Sebagai pembandingan, regresi polinomial derajat 2 dilakukan dengan terlebih dahulu mengubah data X menjadi fitur polinomial menggunakan `PolynomialFeatures(degree=2)`, sehingga memungkinkan model menangkap pola hubungan non-linear. Kedua model kemudian digunakan untuk memprediksi seluruh data X, dan performa dievaluasi menggunakan Mean Squared Error (MSE) pada data uji. Hasil visualisasi menunjukkan bahwa regresi linear (garis merah) menghasilkan garis lurus yang tidak sesuai dengan pola data, sementara regresi polinomial (garis hijau) lebih baik dalam menangkap pola kuadratik.

b. Hasil Gambar



Gambar tersebut menggambarkan perbandingan antara hasil prediksi regresi linear dan regresi polinomial derajat 2 terhadap data yang diberikan. Titik-titik biru mewakili data latih, yang digunakan untuk melatih kedua model, sedangkan titik-titik oranye menunjukkan data uji, yang digunakan untuk mengevaluasi performa model. Garis merah, yang merepresentasikan hasil prediksi dari regresi linear, terlihat membentuk garis lurus yang tidak mampu mengikuti pola kurva data dengan baik, terutama pada bagian data dengan nilai X yang lebih tinggi. Sementara itu, garis hijau, yang merupakan hasil prediksi dari regresi polinomial

derajat 2, terlihat lebih fleksibel dan mengikuti pola kurva data secara mendekati sempurna. Hal ini menunjukkan bahwa regresi polinomial lebih cocok untuk menangkap hubungan non-linear antara variabel X dan Y. Visualisasi ini dengan jelas menggambarkan perbedaan kemampuan kedua model dalam mencocokkan pola data.