

Praktikum Fisika Komputasi

Attala Muflih Gumilang

1227030007

Machine Learning Regresi Linear dan Polinomial

1. Modifikasi kode program contoh 3 dan diberikan Dataset $X = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$ $Y = [3, 7, 13, 21, 31, 43, 57, 73, 91, 111]$!

```
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

np.random.seed(0)
X = [[1], [2], [3], [4], [5], [6], [7], [8], [9], [10]]
Y = [3, 7, 13, 21, 31, 43, 57, 73, 91, 111]
```

2. Bagilah dataset menjadi 80% data latih dan 20% data uji!

```
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)
```

3. Buat model regresi linear dan regresi polinomial derajat 2 dan bandingkan Mean Squared Error (MSE)!

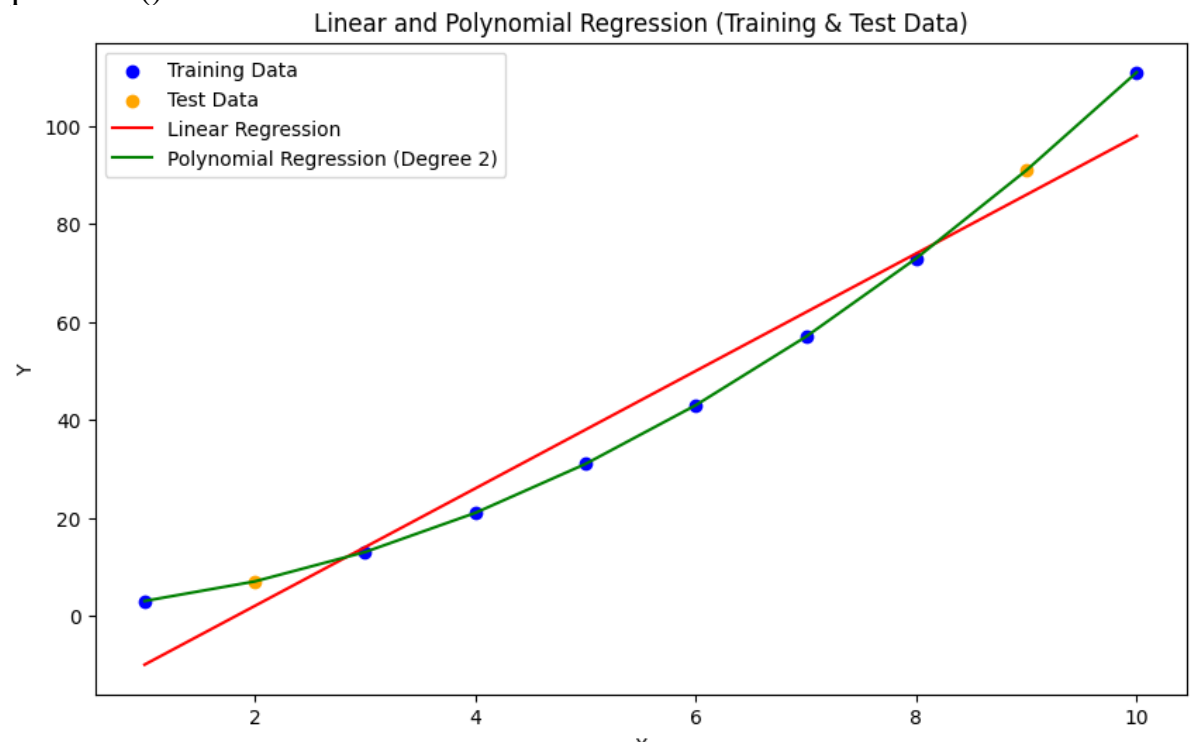
```
Mean Squared Error (Linear): 25.00
Mean Squared Error (Polynomial Degree 2): 0.00
```

Hasil evaluasi menggunakan Mean Squared Error (MSE) menunjukkan regresi linear memiliki MSE sebesar 25, sedangkan regresi polinomial derajat 2 mencapai MSE 0 karena mampu menangkap pola data dengan sempurna. Visualisasi memperlihatkan prediksi linear berupa garis lurus, sedangkan regresi

polinomial menghasilkan kurva sesuai data. Hal ini menegaskan bahwa regresi polinomial lebih cocok untuk dataset dengan pola non-linear.

4. Visualisasikan hasil dari komputasi tersebut!

```
plt.figure(figsize=(10, 6))
plt.scatter(X_train, Y_train, color='blue', label='Training Data')
plt.scatter(X_test, Y_test, color='orange', label='Test Data')
plt.plot(X_sorted, Y_pred_linear_all, color='red', label='Linear Regression')
plt.plot(X_sorted, Y_pred_poly_2_all, color='green', label='Polynomial
Regression (Degree 2)')
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Linear and Polynomial Regression (Training & Test Data)')
plt.legend()
plt.show()
```



5. Jelaskan kode program dan gambar hasil dari soal dengan Bahasa sendiri siingkat, padat dan jelas!

Kode ini membuat dua model untuk memprediksi hubungan X dan Y regresi linear dan polinomial derajat 2. Data dibagi menjadi pelatihan dan pengujian, lalu dievaluasi menggunakan Mean Squared Error (MSE). Regresi linear menghasilkan MSE 25 karena kurang sesuai untuk pola non-linear, sedangkan regresi polinomial yang membentuk kurva sesuai data menghasilkan MSE 0. Grafik menunjukkan data pelatihan (biru) dan pengujian (oranye), dengan garis

merah (linear) dan hijau (polinomial). Model polinomial terlihat lebih akurat dalam mencocokkan pola data.

6. Berikut kodingan yang di gunakan :

```
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

np.random.seed(0)
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='Training Data')
plt.scatter(X_test, Y_test, color='orange', label='Test Data')
plt.plot(X_sorted, Y_pred_linear_all, color='red', label='Linear Regression')
plt.plot(X_sorted, Y_pred_poly_2_all, color='green', label='Polynomial
Regression (Degree 2)')
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Linear and Polynomial Regression (Training & Test Data)')
plt.legend()
plt.show()
```