

# PRAKTIKUM FISIKA KOMPUTASI

## MACHINE LEARNING

Darniel Trio Apriliansyah

NIM. 1227030009

1. Modifikasi kode program contoh 3 dan diberikan dataset

```
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]
```

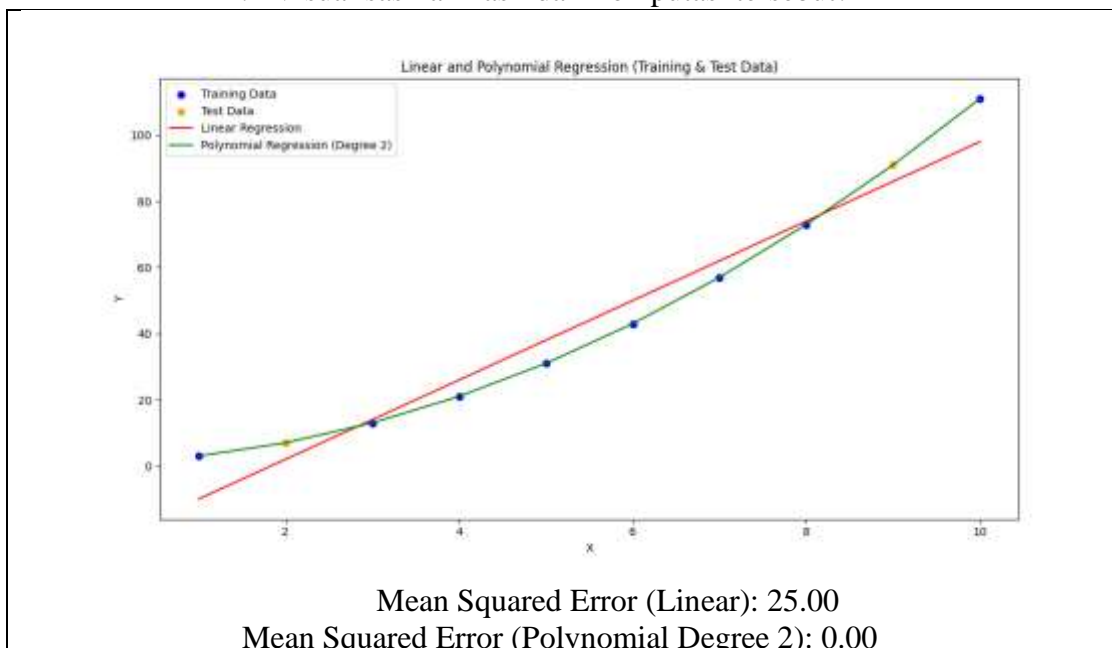
2. Bagilah dataset menjadi 80% data latih dan 20% data uji

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

3. Buat model regresi linear dan regresi polynomial derajat 2 dan bandingkan mean square error (mse)

```
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)
```

4. Visualisasikan hasil dari komputasi tersebut!



5. Jelaskan kode program dan gambar hasil dari soal dengan Bahasa sendiri singkat padat dan jelas

```
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()
```

Kode Program di atas merupakan kode program yang digunakan untuk menghitung nilai MSE atau Mean Square Error pada Regresi Linear dan juga Polinomial.

Dimana kode program ini diawali dengan mengimport library yang diperlukan diantaranya ada library numpy yang digunakan untuk operasi matematika, library matplotlib.pyplot yang digunakan untuk membuat grafik, dan library sklearn yang berisikan metode dari regresi linear dan polynomial itu sendiri. Kemudian pada baris selanjutnya berisikan pendefinisian dari nilai X dan Y. kemudian pada baris baris selanjutnya merupakan algoritma untuk mencari nilai MSE dari regresi linear dan juga polynomial. Dan pada baris terakhir berisikan kode yang digunakan untuk membuat grafik perbandingan antara data latih, data uji pada regresi linear dan polynomial.

Hasil yang didapatkan dari kode program di atas yaitu Mean Square Error (Linear) = 25.00 dan Mean Square Error (Polynomial Degree 2) = 0.00.

Kemudian dihasilkan juga grafik yang berisikan test data yang ditandai dengan titik kuning, training data yang ditandai titik biru, regresi linear yang ditandai garis jingga, dan polynomial degree 2 yang ditandai garis hijau.

Terlihat hasil grafik yang didapatkan garis lurus yang dihasilkan sesuai dengan data dengan kemiringan regresi dan sumbu Y sebagai titik potong. Kemudian pada regresi polynomial tingkat kelengkungan yang dihasilkan sesuai dengan data uji dan data latih yang dihasilkan.