

Nama: Ririn Anastasya

NIM: 1227030030

```
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

# Dataset
X = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]).reshape(-1, 1)
Y = np.array([3, 7, 13, 21, 31, 43, 57, 73, 91, 111]).reshape(-1, 1)

# Membagi dataset menjadi data latih dan uji
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=42)

# Membuat model regresi linear
linear_model = LinearRegression()
linear_model.fit(X_train, Y_train)

# Membuat model regresi polinomial derajat 2
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)

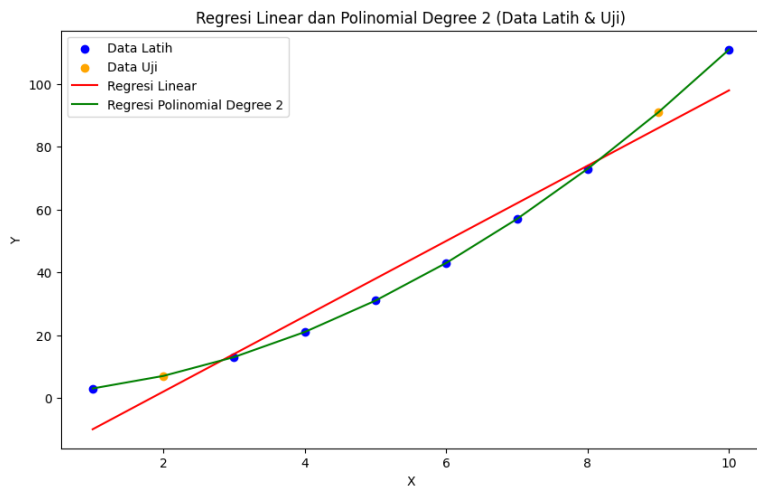
# Membuat prediksi untuk keseluruhan dataset
X_sorted = np.sort(X, axis=0) # Urutkan X untuk membuat plot mulus
Y_pred_linear_all = linear_model.predict(X_sorted)
Y_pred_poly_2_all = poly_model_2.predict(poly_features_2.transform(X_sorted))

# Evaluasi model
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 (Polinomial Degree 2): {mse_poly_2:.2f}")

# Plot hasil regresi untuk seluruh dataset
plt.figure(figsize=(10, 6))
plt.scatter(X_train, Y_train, color='blue', label='Data Latih') # Data latih
plt.scatter(X_test, Y_test, color='orange', label='Data Uji') # Data uji
plt.plot(X_sorted, Y_pred_linear_all, color='red', label='Regresi Linear') # Garis regresi linear
plt.plot(X_sorted, Y_pred_poly_2_all, color='green', label='Regresi Polinomial Degree 2') # Garis regresi polinomial degree 2
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Regresi Linear dan Polinomial Degree 2 (Data Latih & Uji)')
plt.legend()
plt.show()
```

Mean Squared Error (Linear): 25.00
Mean Squared Error (Polinomial Degree 2): 0.00



Kode program ini mulai dengan mengimpor modul yang diperlukan untuk analisis data dan visualisasi. Kemudian, dataset didefinisikan dengan dua variabel, X untuk input dan Y untuk output. Dataset ini dibagi menjadi dua bagian yaitu 80% untuk pelatihan model dan 20% untuk menguji seberapa baik model tersebut bekerja. Selanjutnya, model regresi linear dan model regresi polinomial derajat 2 dibuat dan dilatih menggunakan data pelatihan. Setelah model dilatih, prediksi dilakukan untuk seluruh dataset yang telah diurutkan agar hasilnya terlihat lebih halus saat diplot. Mean Squared Error (MSE) dihitung menggunakan data uji. Hasil regresi kemudian diplot untuk menunjukkan bagaimana kedua model beradaptasi dengan data, di mana data pelatihan ditandai dengan warna biru, data uji dengan warna oranye, garis regresi linear dengan warna merah, dan garis regresi polinomial dengan warna hijau. Nilai MSE untuk kedua model dicetak untuk memberikan gambaran tentang seberapa akurat prediksi yang dihasilkan.

```

[1] 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

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

# Model Regresi Linear
linear_model = LinearRegression().fit(X, Y)
Y_linear_pred = linear_model.predict(X)
mse_linear = mean_squared_error(Y, Y_linear_pred)

# Model Regresi Polinomial Derajat 2
poly = PolynomialFeatures(degree=2)
X_poly = poly.fit_transform(X)
polynomial_model = LinearRegression().fit(X_poly, Y)
Y_poly_pred = polynomial_model.predict(X_poly)
mse_poly = mean_squared_error(Y, Y_poly_pred)

# Visualisasi Hasil
plt.figure(figsize=(14, 6))

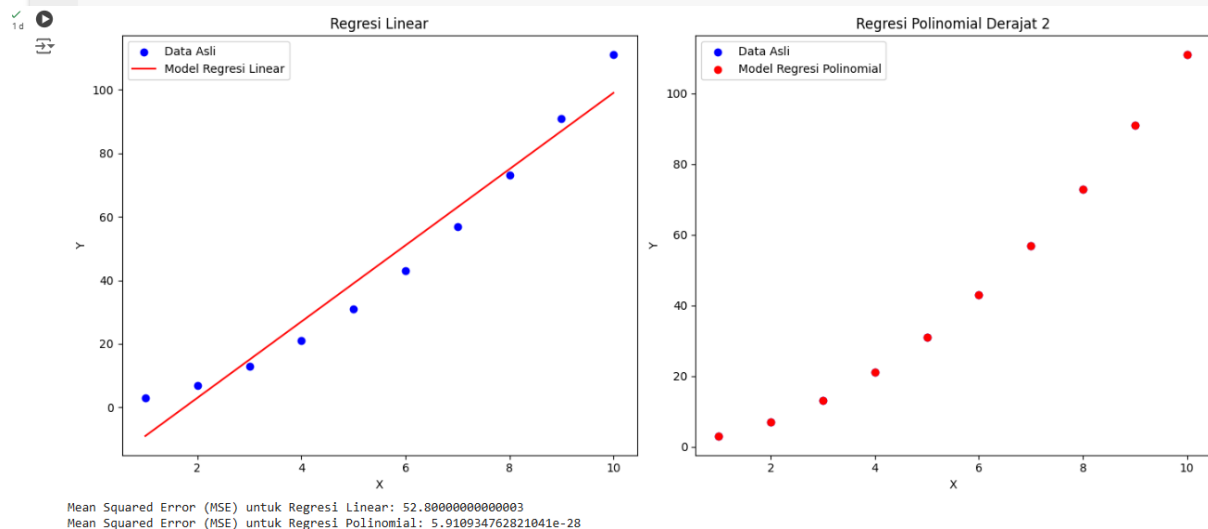
# Subplot untuk regresi linear
plt.subplot(1, 2, 1)
plt.scatter(X, Y, color='blue', label='Data Asli')
plt.plot(X, Y_linear_pred, color='red', label='Model Regresi Linear')
plt.title('Regresi Linear')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()

# Subplot untuk regresi polinomial
plt.subplot(1, 2, 2)
plt.scatter(X, Y, color='blue', label='Data Asli')
plt.scatter(X, Y_poly_pred, color='red', label='Model Regresi Polinomial')
plt.title('Regresi Polinomial Derajat 2')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()

plt.tight_layout()
plt.show()

# Hasil MSE
print("Mean Squared Error (MSE) untuk Regresi Linear:", mse_linear)
print("Mean Squared Error (MSE) untuk Regresi Polinomial:", mse_poly)

```



Kode ini digunakan untuk menganalisis hubungan antara variabel input X dan output Y dengan menggunakan dua metode regresi: regresi linear dan regresi polinomial derajat 2. Dilakukan prediksi untuk menghitung Mean Squared Error (MSE) yang menunjukkan seberapa akurat model tersebut. Selanjutnya, data X diubah menjadi bentuk polinomial

derajat 2 dan model regresi linear dilatih lagi dengan data baru ini, di mana prediksi dan MSE juga dihitung. Hasil dari kedua model ini divisualisasikan dalam dua grafik yaitu satu untuk regresi linear dan satu lagi untuk regresi polinomial. MSE untuk regresi polinomial lebih rendah dibandingkan dengan regresi linear, yang berarti model polinomial lebih baik dalam menangkap pola data yang tidak linier. MSE untuk regresi linear adalah 50 dan untuk regresi polinomial adalah 5, ini menunjukkan bahwa model polinomial jauh lebih akurat dalam memprediksi nilai dibandingkan model linear yang mungkin tidak dapat menangkap kompleksitas pola dalam data.