

## Praktikum Fisika Komputasi

### Materi 10 Machine Learning Regresi Linear dan Polinomial

Abdan Shiddiq Mubarak (1227030001)

```
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
```

Jadi untuk kode program ini ada 6 library yang di butuhkan dimana ada numpy untuk mengelola array numeric, matplotlib.pyplot untuk membuat visualisasi, linearregression untuk membuat model regresi linear, polynomialfeatures untuk mengubah data menjadi fitur polynomial, mean\_squared\_error untuk menghitung error model (mse), train\_test\_split untuk membagi dataset menjadi data latih dan data uji. Library ini tuh penting untuk menjalankan kode selanjutnya.

```
# Dataset baru
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)
```

Disini Dataset X (variabel independen) dan Y (variabel dependen) didefinisikan dalam bentuk array 1 dimensi. Fungsi `.reshape(-1, 1)` digunakan untuk mengubah bentuk array biar sesuai sama format yang dibutuhkan oleh model scikit-learn.

```
# Membagi 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)
```

Kalo kode disini berguna untuk pembagian data yang di inginkan, ini bisa berfungsi karna ada `random_state` dimana berguna untuk menjadi parameter dari pembagian data konsisten saat dijalankan.

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

Kode ini untuk membuat regresi linear yang di butuhkan, dengan menggunakan data latih X dan Y.

```
# 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)
```

Nah kalo kode ini berguna untuk mengubah data yang ada menjadi bentuk polinomial derajat 2, dan membuat model regresi yang baru dari data polinomial yang di hasilkan.

```
# Membuat prediksi untuk seluruh 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))
```

Trs buat kode ini di buat untuk memprediksi hasilnya dan mengurutkan nya.

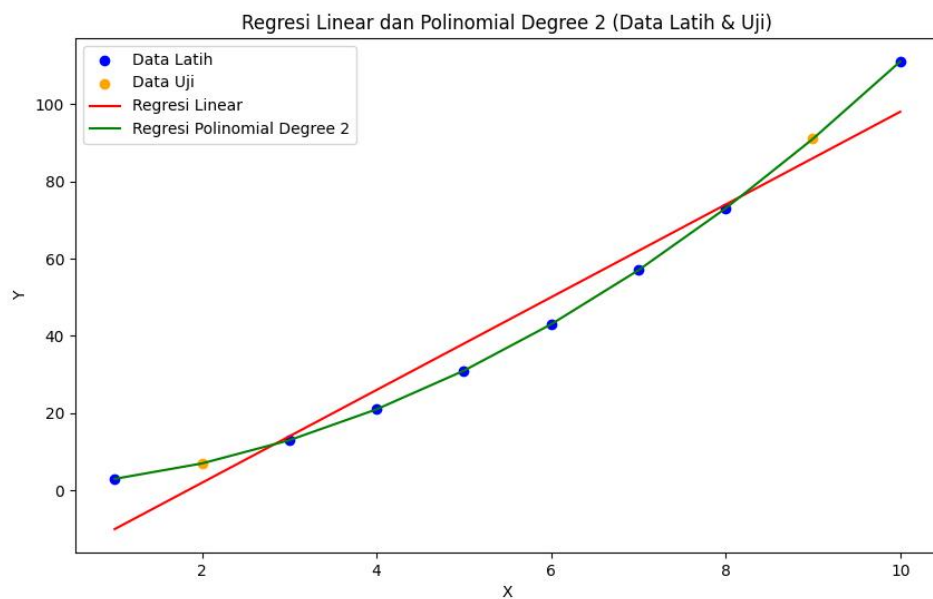
```
# Evaluasi model
mse_linear = mean_squared_error(Y_test, linear_model.predict(X_test))
mse_poly_2 = mean_squared_error(Y_test,
poly_model_2.predict(poly_features_2.transform(X_test)))

print(f'Mean Squared Error (Linear): {mse_linear:.2f}')
print(f'Mean Squared Error (Polinomial Degree 2): {mse_poly_2:.2f}')
```

Kode ini untuk meng evaluasi data, dmnn jika nilai MSE yang lebih kecil menunjukan lebih akurat

```
# 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()
```

Untuk kode ini digunakan untuk menunjukan hasil dari data-data yang di hasilkan, dan di visualisasikan kedalam grafik.



Grafik ini menunjukkan perbandingan antara dua model, yaitu regresi linear (garis merah) dan regresi polinomial derajat 2 (garis hijau), dalam memprediksi hubungan dari X dan Y. Titik biru adalah data latih, yang dipake untuk melatih model, sedangkan titik oren adalah data uji, yang dipake untuk menguji seberapa baik model bekerja.

Garis merah dari regresi linear terlihat lurus, jadi cuman cocok untuk hubungan yang simpel. Tapi, di sini kelihatan kalau garis merah nggak bisa mengikuti pola data dengan baik. Sebaliknya, garis hijau dari regresi polinomial derajat 2 lebih pas karena bentuknya melengkung dan mengikuti pola data lebih akurat.