Praktikum Fisika Komputasi

Materi 10 Machine Learning Regresi Linear dan Polinomial

Abdan Shiddiq Mubarok (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 disni berguna unruk pembagian data yang di inginkan, ini bisa berfungsi karna ada rando\_mstate 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 regrensi linear yang du 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 polinmial derajat 2, dan membuat model regrensi yang baru dari data polynomial 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)) |

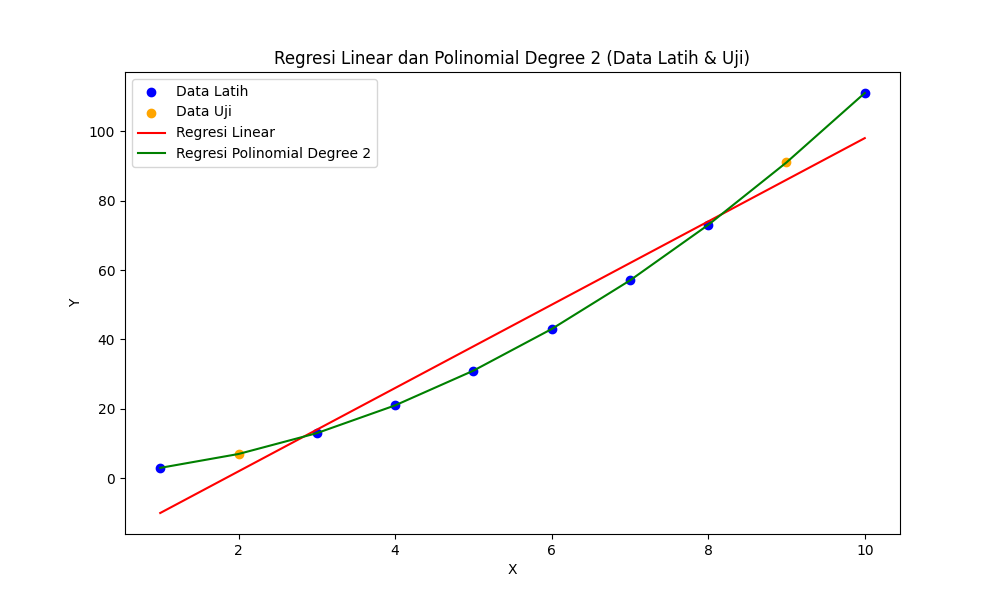
Trs buat kode ini di buat untuk memprediksi hasilnya dan mengurutkan nya.

|  |
| --- |
| # 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, dmn 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 keliatan 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.