

Nama Dosen : Teguh Iman Hermanto, M.Kom
 Mata Kuliah : Machine Learning 1
 Pembahasan : Simple Linear Regresion
 Pokok Pemb : - Regresi Linier Berganda
 - Membangun Model Regresi Linier Berganda
 - Simulasi Algoritma Regresi Linier Berganda
 - Evaluasi Algoritma Regresi Linier Berganda

1. Regresi Linier Berganda

a. Konsep Regresi Linier Berganda

Regresi Linier Berganda adalah salah satu teknik statistik yang digunakan untuk memodelkan hubungan antara satu variabel dependen (output/label) dan lebih dari satu variabel independen (input/fitur). Regresi linier berganda memperluas konsep regresi linier sederhana, yang hanya melibatkan satu variabel independen, dengan memasukkan beberapa variabel independen dalam analisis.

Tujuan utama dari regresi linier berganda adalah untuk menemukan hubungan linear antara variabel-variabel independen dan variabel dependen, sehingga kita dapat memprediksi nilai variabel dependen berdasarkan nilai-nilai variabel independen.

b. Aplikasi Regresi Linier Berganda

Regresi linier berganda sering digunakan dalam berbagai bidang seperti:

- **Ekonomi:** untuk memprediksi pendapatan berdasarkan berbagai faktor seperti tingkat pendidikan, pengalaman kerja, dan usia.
- **Pemasaran:** untuk menganalisis dampak harga, iklan, dan lokasi terhadap penjualan produk.
- **Kesehatan:** untuk memprediksi jumlah kalori yang dibakar berdasarkan usia, berat badan, tinggi badan, dan durasi olahraga.

c. Rumus Regresi Linier Berganda

Secara matematis, persamaan regresi linier berganda dapat dinyatakan sebagai berikut:

$$Y = a + b_1 X_1 + b_2 X_2 + b_3 X_3 + \dots + b_n X_n$$

Di mana:

- Y = adalah variabel dependen (yang ingin diprediksi).
- $X_1, X_2, X_3, \dots, X_n$ = adalah variabel independen (faktor yang memengaruhi variabel dependen).
- a = adalah konstanta atau intercept, yaitu nilai Y saat semua variabel independen bernilai nol.
- $b_1, b_2, b_3, \dots, b_n$ adalah koefisien regresi, yang menunjukkan seberapa besar pengaruh setiap variabel independen terhadap variabel dependen.

2. Membangun Model Regresi Linier Berganda

```
1 import pandas as pd
2 import numpy as np
3
4 from sklearn.linear_model import LinearRegression
5 from sklearn.model_selection import train_test_split
6 from sklearn.metrics import mean_squared_error, r2_score
7
8 import matplotlib.pyplot as plt
9 import seaborn as sns
10 import plotly.express as px
11 import plotly.graph_objects as go
```

```
1 df = pd.read_csv('kalori_ganda.csv')
```

```
1 # 1. Scatter plot untuk melihat hubungan antara Umur dan Kalori
2 plt.figure(figsize=(7,5))
3 plt.scatter(x='Umur', y='Kalori', data=df, color='blue')
4 plt.title('Age vs Calorie Intake')
5 plt.xlabel('Age (years)')
6 plt.ylabel('Calorie Intake (kcal)')
7 plt.grid(True)
8 plt.show()
```



```
1 # 2. Berat Badan vs Konsumsi Kalori
2 plt.figure(figsize=(7,5))
3 plt.scatter(x='BB', y='Kalori', data=df, color='green')
4 plt.title('Weight vs Calorie Intake')
5 plt.xlabel('Weight (kg)')
6 plt.ylabel('Calorie Intake (kcal)')
7 plt.grid(True)
8 plt.show()
```



```
1 # Histogram untuk melihat distribusi BB
2 plt.figure(figsize=(8, 6))
3 sns.histplot(x='BB', data=df)
4 plt.title('Distribusi BB')
5 plt.show()
```



```
1 # menampilkan plot animasi 3d
2 fig = px.scatter_3d(df, x='Umur', y='BB', z='TB',
3                     color='Olahraga', size='Kalori',
4                     hover_name='Umur')
5 fig.show()
```

3. Simulasi Algoritma Regresi Linier Berganda



```
1 # Memisahkan fitur (X) dan target (y)
2 x = df[['Umur', 'BB', 'TB', 'Olahraga']]
3 y = df['Kalori']
```



```
1 # Membagi data menjadi data latih dan data uji
2 X_train, X_test, y_train, y_test = train_test_split(
3     x, y, test_size=0.2, random_state=42)
```



```
1 # Membuat model regresi linier
2 model = LinearRegression()
3 model.fit(x, y)
```



```
1 # Menampilkan intercept regresi dari model
2 intercept = model.intercept_
3 intercept
```



```
1 # Menampilkan koefisien regresi dari model
2 coefficients = model.coef_
3 coefficients
```



```
1 manual = intercept+(25*3.05138649)+(75*22.34269713)+(170*0.0466017)+(60*0.06622248)
2 manual
```



```
1 # Data input untuk prediksi
2 input_data = np.array([[25, 75, 170, 60]])
```



```
1 # Melakukan prediksi
2 predicted_calories = model.predict(input_data)
3 predicted_calories
```

4. Evaluasi Regresi Linier Berganda



```
1 # Memprediksi nilai BB dengan data uji
2 y_pred = model.predict(X_test)
```




```
1 # Menghitung MSE dan R2 score
2 mse = mean_squared_error(y_test, y_pred)
3 r2 = r2_score(y_test, y_pred)
```



```
1 # Menampilkan MSE dan R2 score
2 print(f'Mean Squared Error (MSE): {mse}')
3 print(f'R-squared (R2): {r2}')
```

5. Simpan Model



```
1 # Menyimpan model menggunakan pickle
2 import pickle
3
4 filename = 'regression_model.pkl'
5 pickle.dump(model, open(filename, 'wb'))
```

Dari perintah ini akan menghasilkan file baru yaitu **regression_model.pkl**. file ini akan otomatis muncul di direktori project kalian. Apabila berhasil, file model ini bisa kita gunakan untuk membuat aplikasi machine learning berbasis web menggunakan streamlit.

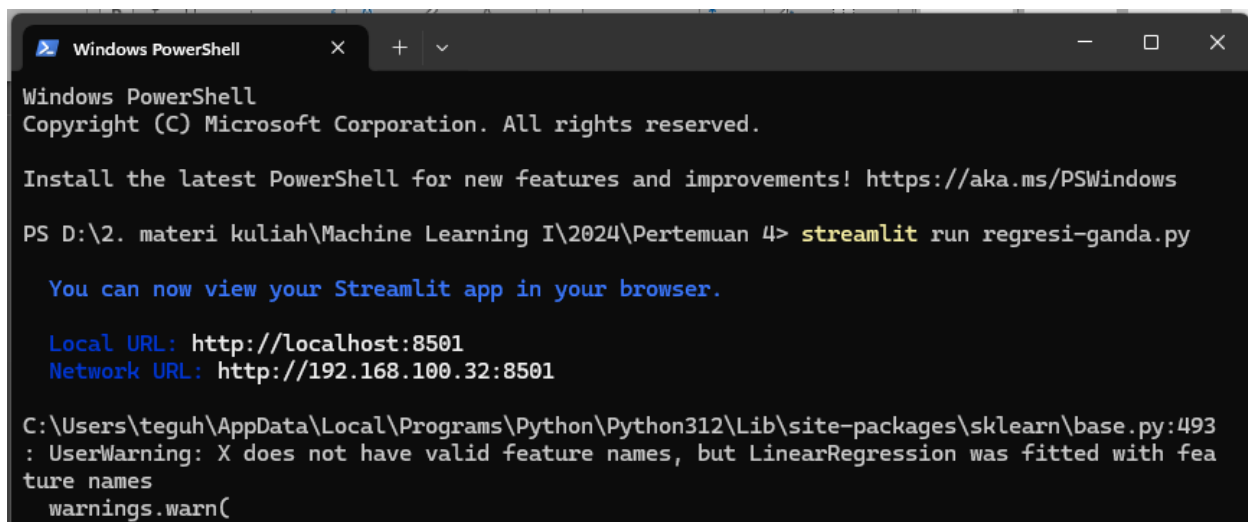
6. Membuat Aplikasi

Buat file **regresi-ganda.py** pada direktori yang sama.

```

1 import pickle
2 import streamlit as st
3
4 # Aplikasi Streamlit
5 st.title('Prediksi Kalori')
6
7 # Form input data
8 st.header('Masukkan Data')
9 umur = st.number_input('Umur', min_value=25, max_value=55)
10 bb = st.number_input('Berat Badan (BB)', min_value=60, max_value=95)
11 tb = st.number_input('Tinggi Badan (TB)', min_value=155, max_value=180)
12 olahraga = st.number_input('Durasi Olahraga (menit)', min_value=20, max_value=90)
13
14 # Tombol prediksi
15 if st.button('Prediksi'):
16     # Memuat model dari file pickle
17     loaded_model = pickle.load(open('regression_model.pkl', 'rb'))
18
19     # Melakukan prediksi
20     input_data = [[umur, bb, tb, olahraga]]
21     prediction = loaded_model.predict(input_data)
22
23     # Menampilkan hasil prediksi
24     st.header('Hasil Prediksi')
25     st.write(f'Kalori yang diperkirakan: {prediction[0]:.2f}')
```

Untuk menjalankan aplikasinya buka command prompt, lalu arahkan ke direktory project dan masukan perintah **streamlit run nama_aplikasi.py**



```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\2. materi kuliah\Machine Learning I\2024\Pertemuan 4> streamlit run regresi-ganda.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.100.32:8501

C:\Users\teguh\AppData\Local\Programs\Python\Python312\Lib\site-packages\sklearn\base.py:493
: UserWarning: X does not have valid feature names, but LinearRegression was fitted with fea
ture names
warnings.warn(
```