SIMPLE DAN MULTIPLE LINEAR REGRESSION

Penyusun :
Andika Ainur Wibowo (2141720238)
Ibnu Hajar Askholani (2141720170)
Diwa Oliq Atthoriq (2141720170)
Muhammad Farhan R. P. (2141720197)
Zaky Muhammad I. (2141720131)
Trisinus Gulo (2141720035)

Kelas : TI 3C
Mata Kuliah : Mesin Learning
Dosen Pengampu :
Ely Setyo Astuti ST., MT.

JURUSAN TEKNOLOGI INFORMASI

PRODI TEKNIK INFORMATIKA

POLITEKNIK NEGERI MALANG

2023

Tugas Praktikum

1. Identifikasi variabel-variabel yang akan digunakan sebagai variabel bebas (fitur) dan variabel target (biaya medis personal).

Keterangan: Import library yang diperkukan

```
# Import library yang diperlukan
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
```

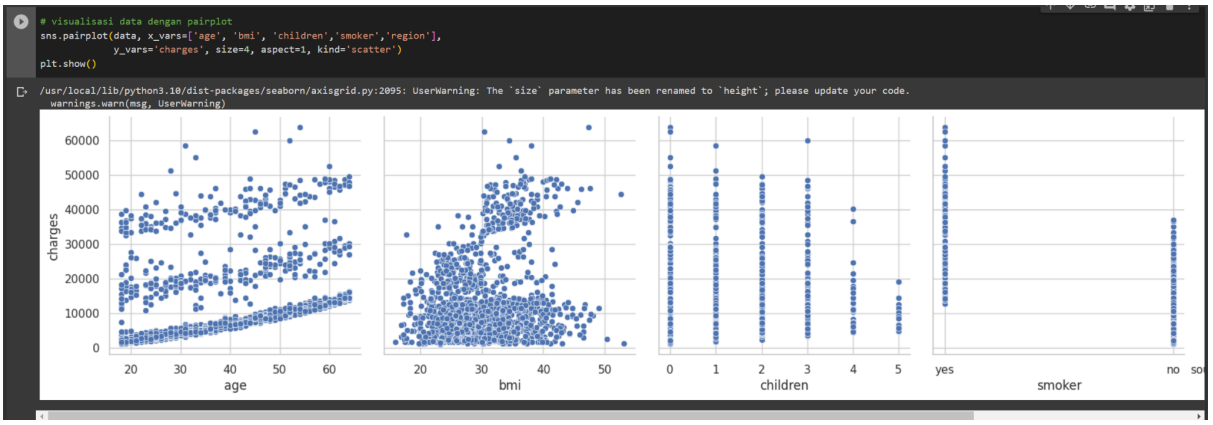
Keterangan : Membaca file CSV dari google drive serta menampilkan isi datanya

```
[6] # baca data dari file CSV
    data = pd.read_csv('/content/drive/MyDrive/Machine-Learning/Minggu3/Tugas Praktikum/insurance.csv')

# melihat beberapa data awal
data.head()

# mengecek ukuran data
data.shape

# informasi tentang data
data.info()

# deskripsi data
data.describe()
```

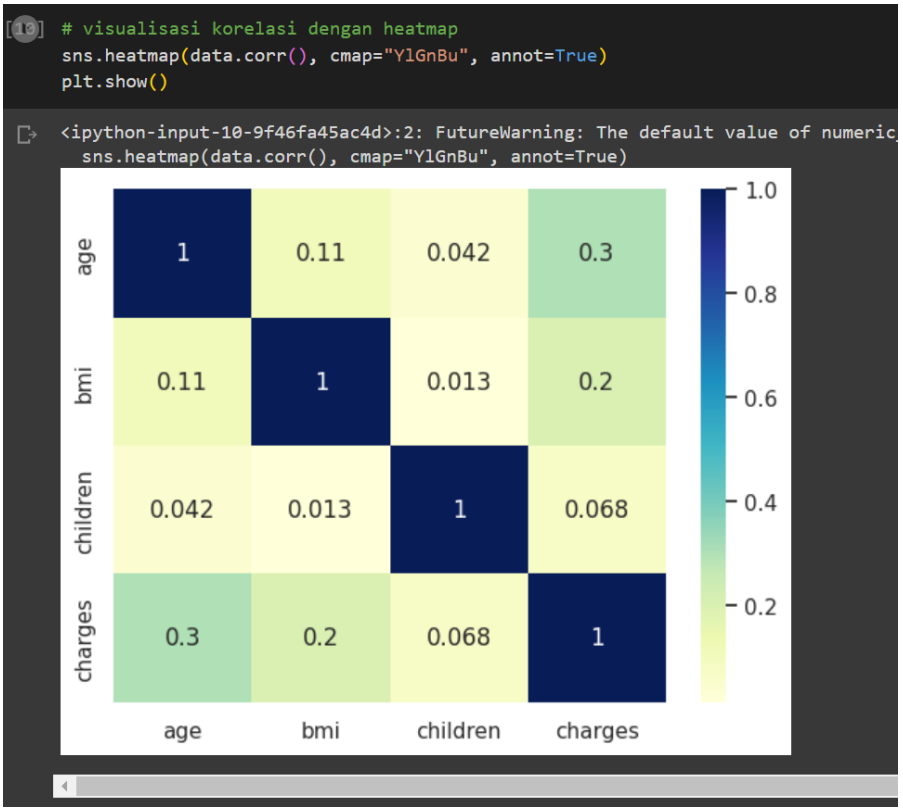
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

Keterangan: Memvisualisasikan data setiap column dengan pairplot

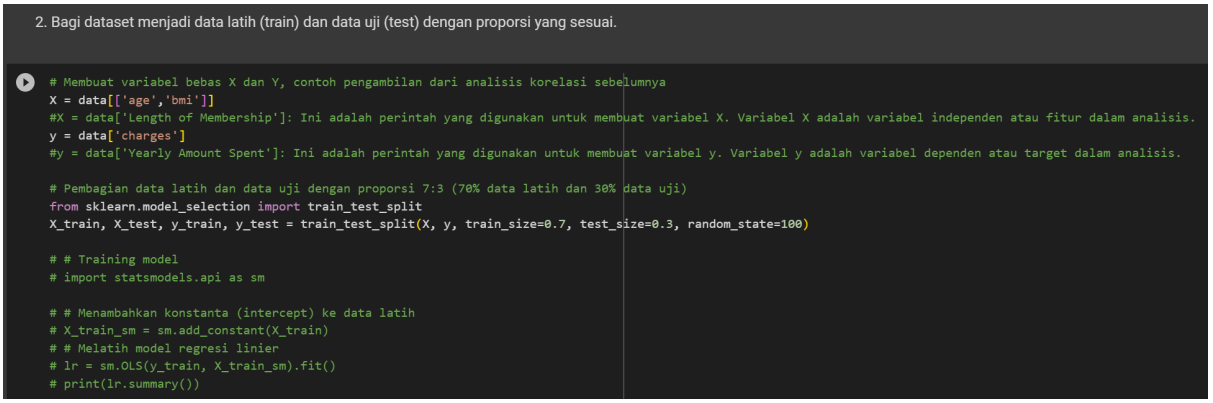


Keterangan: Visulisasi korelasi dengan hatmap



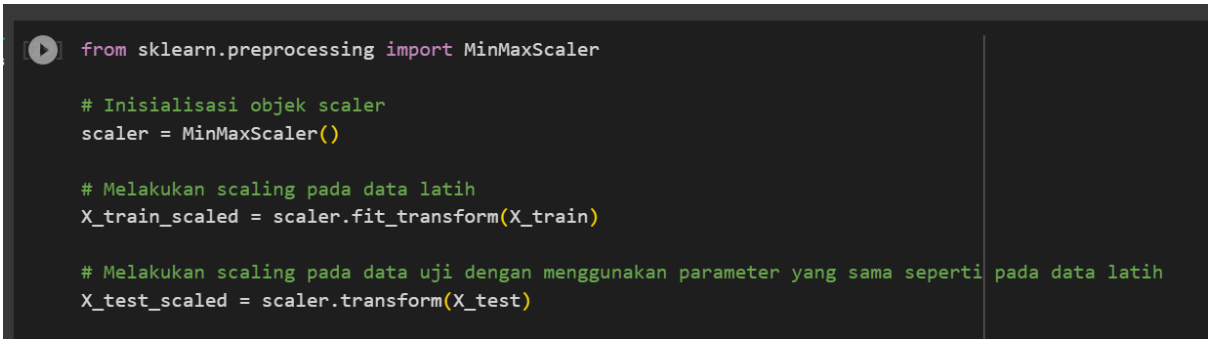
2. Bagi dataset menjadi data latih (train) dan data uji (test) dengan proporsi yang sesuai.

Keterangan: Kelompok kami menggunakan data independen (X) “age” dan “bmi” untuk data dependen (y) yaitu “charges”. Pembagian data latih dan data uji dengan proporsi 7:3

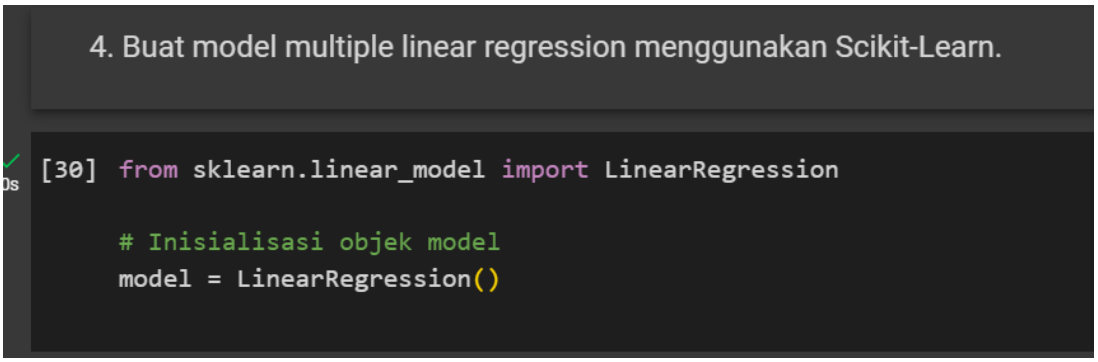


3. Lakukan feature scaling jika diperlukan.

Keterangan: Kelompok kami menggunakan minMaxScaler dalam proses scaling



4. Buat model multiple linear regression menggunakan Scikit-Learn.



5. Latih model pada data latih dan lakukan prediksi pada data uji.

Keterangan: Di bawah ini merupakan code untuk melatih model data latih dan melakukan prediksi pada data uji yang telah di scaling

```
[31] # # Melatih model dengan data latih
     # model.fit(X_train, y_train)

     # # Membuat prediksi menggunakan data uji
     # y_pred = model.predict(X_test)

     # Melatih model dengan data latih yang telah di-scaled
     model.fit(X_train_scaled, y_train)

     # Melakukan prediksi pada data uji yang telah di-scaled
     y_pred = model.predict(X_test_scaled)
```

6. Evaluasi model dengan menghitung metrik seperti R-squared, MSE, dan MAE. Tampilkan hasil evaluasi.

Keterangan: Berikut adalah code untuk menghitung metrik evaluasi dan visualisasi multiple linier regression

```
# Menghitung metrik evaluasi
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Menghitung metrik evaluasi (misalnya, RMSE dan R-squared)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))

# Menampilkan hasil evaluasi
print("Mean Squared Error (MSE):", mse)
print("Mean Absolute Error (MAE):", mae)
print("R-squared (R2):", r2)
print("Root Mean Squared Error (RMSE):", rmse)

# Membuat plot multiple linier regression untuk age
plt.scatter(X_test['age'], y_test, color='blue', label='Data Asli')
plt.scatter(X_test['age'], y_pred, color='red', label='Prediksi')
plt.xlabel("Umur (age)")
plt.ylabel("Biaya (charges)")
plt.title("Multiple Linier Regression untuk Umur (age)")
plt.legend()
plt.show()

# Membuat plot multiple linier regression untuk bmi
plt.scatter(X_test['bmi'], y_test, color='blue', label='Data Asli')
plt.scatter(X_test['bmi'], y_pred, color='red', label='Prediksi')
plt.xlabel("Indeks Massa Tubuh (bmi)")
plt.ylabel("Biaya (charges)")
plt.title("Multiple Linier Regression untuk Indeks Massa Tubuh (bmi)")
plt.legend()
plt.show()
```

```
Mean Squared Error (MSE): 129696311.34332742
Mean Absolute Error (MAE): 8919.311771816696
R-squared (R2): 0.10676050395208692
Root Mean Squared Error (RMSE): 11388.4288355913
```