Laporan Tugas Praktikum3



**Syahrul Giga Wahyudi - 0110224085**

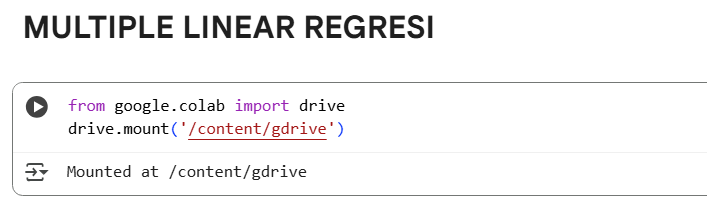
Teknik Informatika, STT Terpadu Nurul Fikri, Depok

[0110224085@student.nurulfikri.ac.id](mailto:0110224085@student.nurulfikri.ac.id)

**Multiple Linear Regression**

Pada bagian ini dilakukan penerapan regresi linear berganda untuk memprediksi berat badan balita berdasarkan dua variabel bebas, yaitu umur (dalam bulan) dan tinggi badan (dalam sentimeter).

*1.1 Load data dari Google drive*

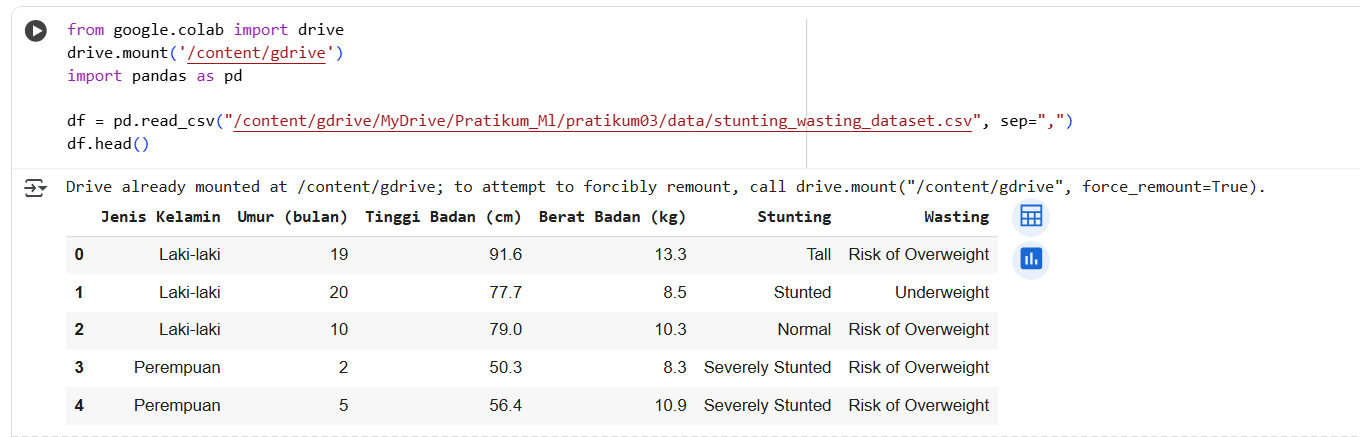
****

Gambar 1.1 kode dan output

Penjelasan

Bagian kode ini digunakan untuk **menghubungkan Google Colab dengan akun Google Drive mu**.. Setelah kode ini dijalankan, akan muncul tautan otorisasi. Kamu harus mengklik tautan tersebut, memilih akun Google, dan memberikan izin agar Colab bisa mengakses file-file yang tersimpan di Google Drive-mu. Proses ini hanya perlu dilakukan satu kali per sesi.

1.2 membaca file csv

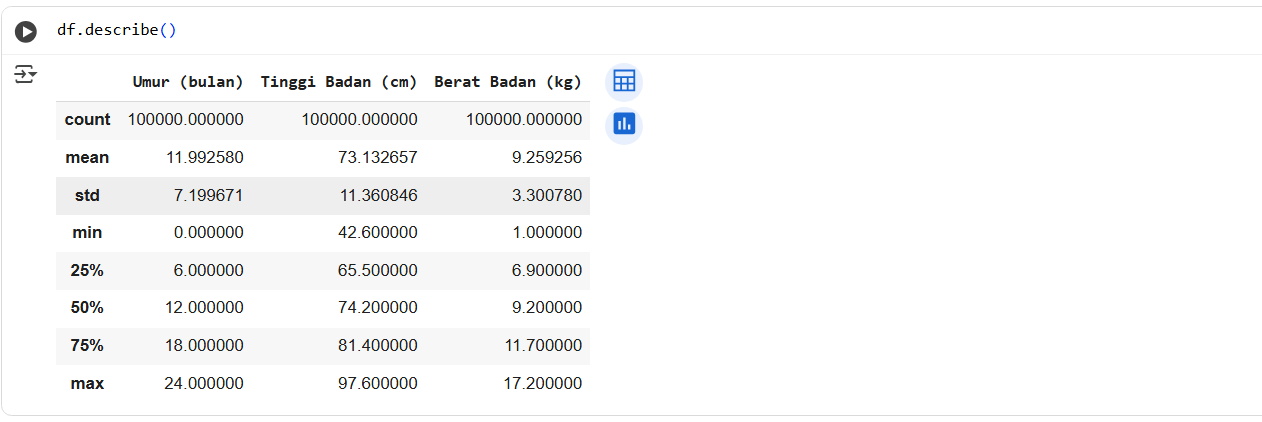
****

Gambar 1.2 kode dan output

Penjelasan

Data diambil menggunakan pustaka pandas dengan fungsi read\_csv(). Dataset berisi informasi mengenai jenis kelamin, umur, tinggi, dan berat badan balita. Setelah data dimuat ke dalam DataFrame, proses analisis dapat dilakukan secara lebih mudah dan terstruktur.

1.3 informasi data set

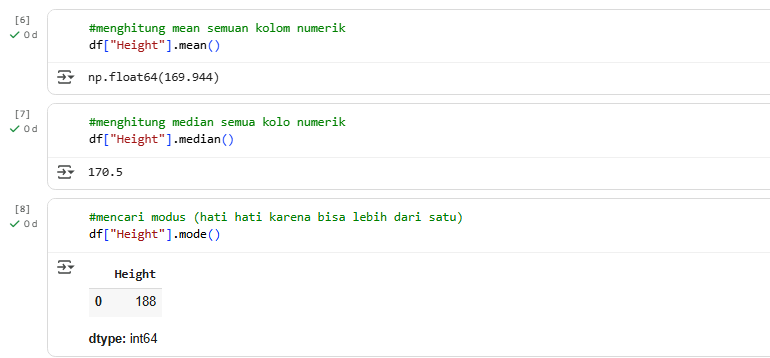
****

Gambar 1.3 kode dan output

Penjelasan

Metode .info() digunakan untuk melihat gambaran umum struktur data. Fungsi ini menampilkan jumlah baris dan kolom, nama kolom, jumlah data yang terisi, serta tipe data pada setiap kolom, seperti int64 untuk angka dan object untuk teks.

1.4 Pra-pemrosesan Data

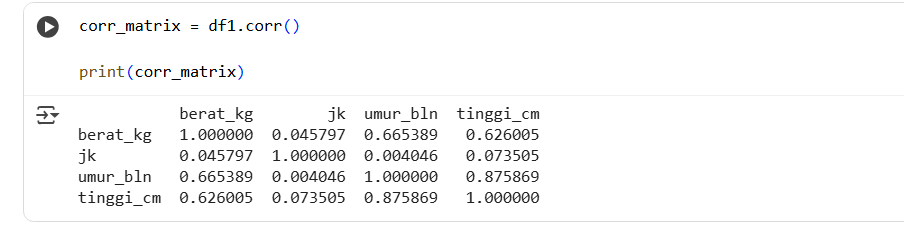
****

Gambar 1.4 kode dan output

Penjelasan

Dari data asli, hanya kolom yang relevan digunakan dalam model, yaitu umur\_bln, tinggi\_cm, dan berat\_kg. Data kemudian disalin ke DataFrame baru agar tidak mengubah data mentah. Tahap ini memastikan hanya data penting yang dianalisis dan siap digunakan pada pemodelan regresi.

1.5 Analisis Korelasi

****

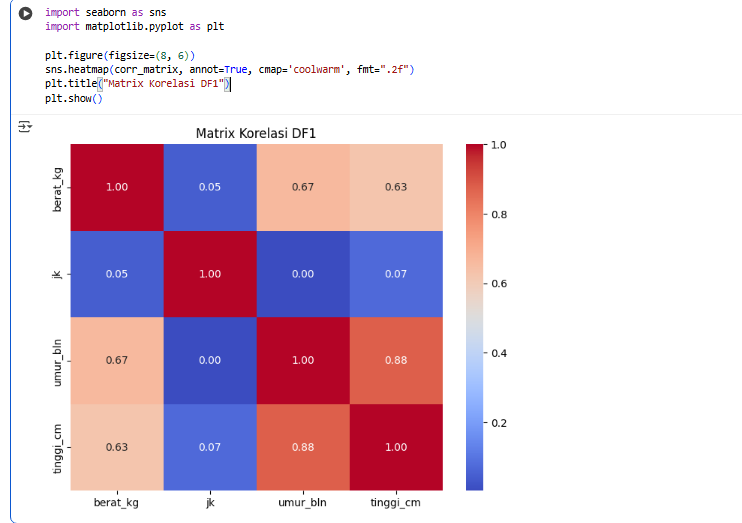
Gambar 1.5 kode dan output

Penjelasan

Hubungan antarvariabel diuji dengan metode korelasi menggunakan fungsi .corr().  
Hasilnya menunjukkan bahwa:

* Variabel **umur** memiliki korelasi 0.67 terhadap berat badan,
* Variabel **tinggi badan** memiliki korelasi 0.63 terhadap berat badan,
* Sementara **jenis kelamin** hanya berkorelasi 0.05 dan tidak signifikan.  
  Berdasarkan hasil tersebut, variabel **umur** dan **tinggi badan** digunakan sebagai prediktor (X), sedangkan **berat badan** menjadi variabel target (Y).

1.6 visualisasi heat map

****

Gambar 1.6 kode dan output

Penjelasan

Kode di atas digunakan untuk menampilkan dari matriks korelasi antarvariabel numerik.  
Baris import seaborn as sns dan import matplotlib.pyplot as plt berfungsi untuk memanggil pustaka visualisasi data yang digunakan dalam pembuatan grafik.

Perintah plt.figure(figsize=(8, 6)) mengatur ukuran tampilan grafik agar hasil visualisasi lebih jelas dan proporsional.  
Fungsi sns.heatmap(corr\_matrix, annot=True, cmap='coolwarm', fmt=".2f") digunakan untuk menampilkan nilai korelasi antarvariabel dalam bentuk warna.  
Parameter:

* annot=True menampilkan nilai korelasi pada setiap sel,
* cmap='coolwarm' memberikan gradasi warna merah–biru untuk menunjukkan tingkat korelasi,
* fmt=".2f" menampilkan angka korelasi dengan dua digit desimal.

Terakhir, plt.title("Matrix Korelasi DF1") memberikan judul pada grafik, dan plt.show() menampilkan hasil visualisasi ke layar.

1.7 Membagi dataset untuk Training dan Test

****

Gambar 1.7 kode dan output

Penjelasan

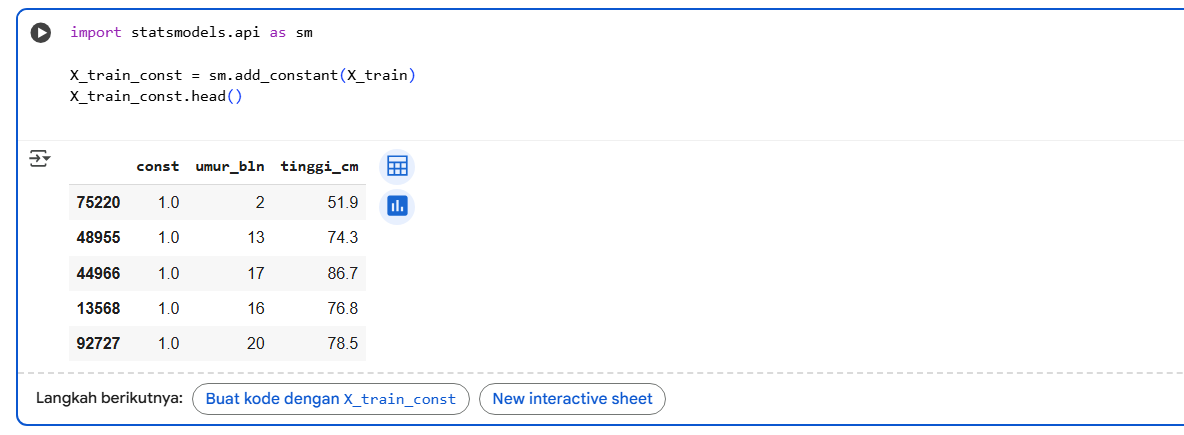
Kode di atas digunakan untuk **membagi dataset menjadi data latih (training) dan data uji (testing)** sebelum dilakukan pemodelan regresi linear berganda.  
Pertama, variabel **y** ditetapkan sebagai variabel target, yaitu *berat badan balita*, sedangkan variabel **X** berisi dua fitur prediktor yaitu *umur dalam bulan* dan *tinggi badan*.

Fungsi train\_test\_split() dari pustaka scikit-learn digunakan untuk memisahkan data menjadi dua bagian:

* **Data training (80%)** digunakan untuk melatih model,
* **Data testing (20%)** digunakan untuk menguji performa model.

Parameter random\_state=42 digunakan agar proses pembagian data selalu menghasilkan hasil yang sama setiap kali kode dijalankan (reproducible).  
Terakhir, fungsi len() digunakan untuk menampilkan jumlah data pada masing-masing bagian, yaitu data latih dan data uji.

1.8 kolom konstanta (intercept)

****

Gambar 1.8 kode dan output

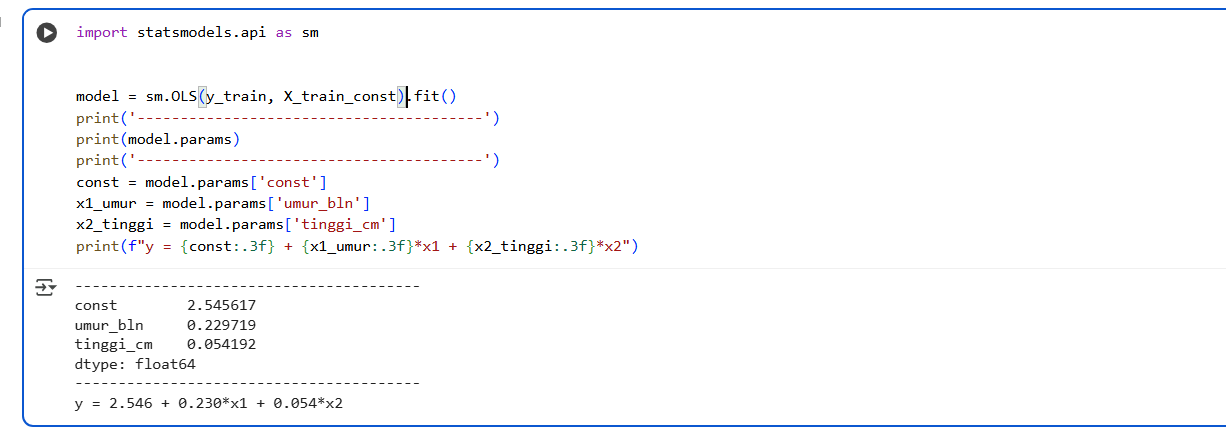
Penjelasan

Kode ini digunakan untuk **menambahkan kolom konstanta (intercept)** ke dalam data training sebelum membangun model regresi linear berganda menggunakan pustaka statsmodels.

Baris import statsmodels.api as sm berfungsi untuk memanggil pustaka *Statsmodels*, yang digunakan dalam analisis statistik dan pembuatan model regresi.

Selanjutnya, perintah sm.add\_constant(X\_train) menambahkan nilai konstanta ke setiap baris data agar model regresi dapat menghitung *intercept* (titik potong) selain koefisien variabel independen.  
Langkah ini penting karena tanpa konstanta, model regresi akan diasumsikan melewati titik nol, yang bisa menyebabkan hasil prediksi menjadi kurang akurat.

2.1 membangun dan menampilkan model regresi linear multiple

****

Gambar 2.1 kode dan output

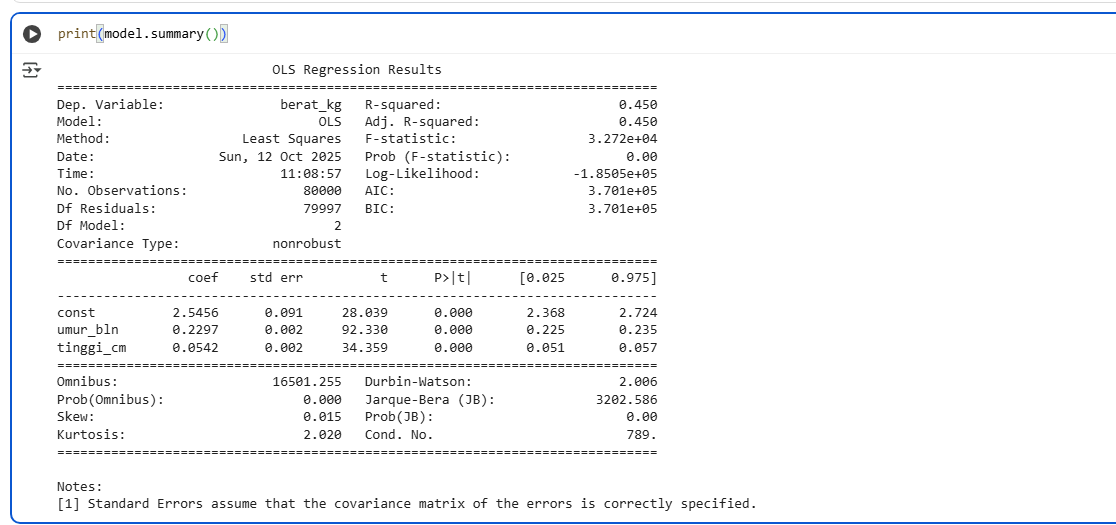
Penjelasan

Kode di atas digunakan untuk **membangun dan menampilkan model regresi linear multiple** menggunakan metode *Ordinary Least Squares (OLS)* dari pustaka statsmodels.

Baris model = sm.OLS(y\_train, X\_train\_const).fit() berfungsi untuk melatih model regresi dengan data training yang telah ditambahkan konstanta sebelumnya.  
Metode fit() digunakan untuk menghitung nilai **koefisien regresi** dan **intercept** yang paling sesuai dengan data.

Perintah model.params menampilkan parameter hasil pelatihan, yaitu nilai konstanta (const) serta koefisien untuk masing-masing variabel independen (umur\_bln dan tinggi\_cm).  
Kemudian, setiap parameter disimpan ke dalam variabel terpisah (const, x1\_umur, dan x2\_tinggi) agar lebih mudah digunakan atau ditampilkan Kembali

2.2 menampilkan ringkasan hasil analisis regresi linear multiple

****

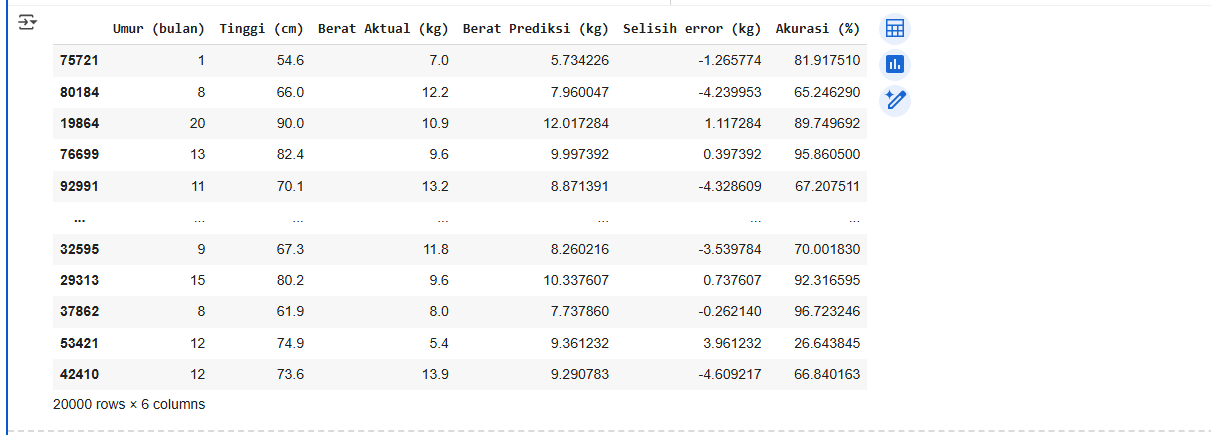
Gambar 2.2 kode dan output

Penjelasan

Kode di atas berfungsi untuk **menampilkan ringkasan hasil analisis regresi linear multiple** yang telah dibangun menggunakan metode *Ordinary Least Squares (OLS)*.  
Perintah model.summary() menghasilkan tabel yang berisi informasi statistik penting mengenai performa model, hubungan antarvariabel, serta tingkat signifikansi masing-masing parameter.

2.3 Scatter Plot (Korelasi Positif)

****

****

Gambar 2.3 kode dan output

Penjelasan

Kode ini digunakan untuk **menghasilkan hasil prediksi dan menghitung akurasi model regresi**.  
Pertama, sm.add\_constant() menambahkan kolom konstanta ke data uji agar sesuai dengan struktur model.  
Kemudian, model.predict() digunakan untuk memprediksi berat badan berdasarkan variabel umur dan tinggi.  
Hasilnya disimpan dalam DataFrame yang memuat nilai aktual, prediksi, dan selisih error.

*tugas pratikum mandiri*

**Prediksi Jumlah Penyewaan Sepeda Harian Menggunakan Model Regresi Linear pada Dataset Bike Sharing**

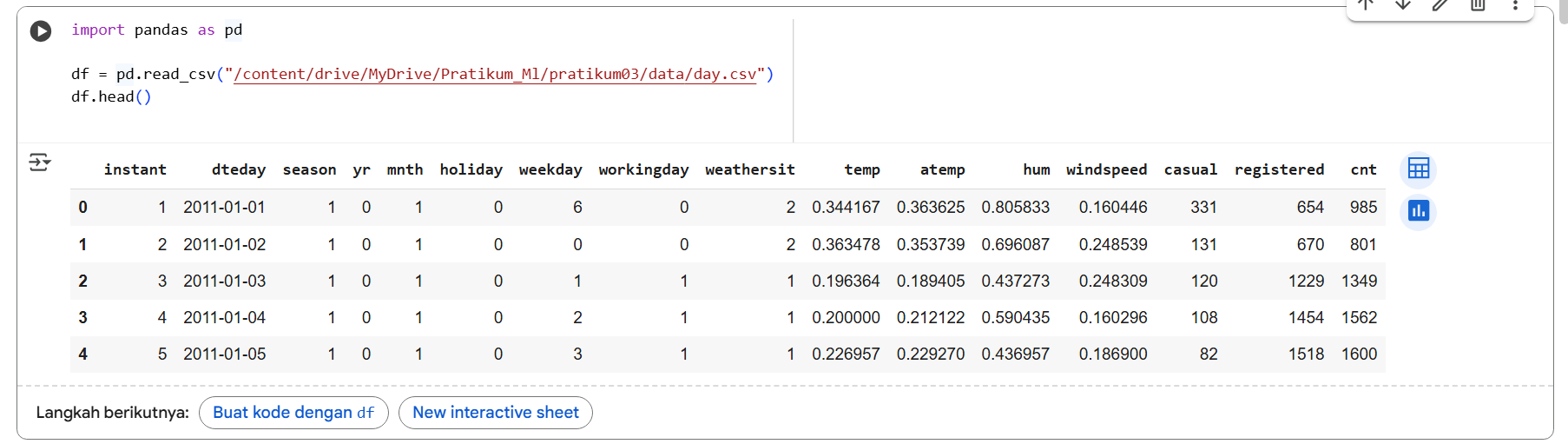
*3.1 Load data dari Google drive*

****

Pada bagian awal program, dilakukan proses *mount* Google Drive menggunakan perintah drive.mount('/content/gdrive'). Langkah ini bertujuan agar file dataset yang tersimpan di Google Drive dapat diakses langsung dari lingkungan Google Colab. Setelah proses otorisasi berhasil, direktori /content/gdrive akan berfungsi sebagai penghubung antara Colab dan akun Google Drive

Gambar 3.1 kode tugas mandiri

*3.2 membaca file csv*

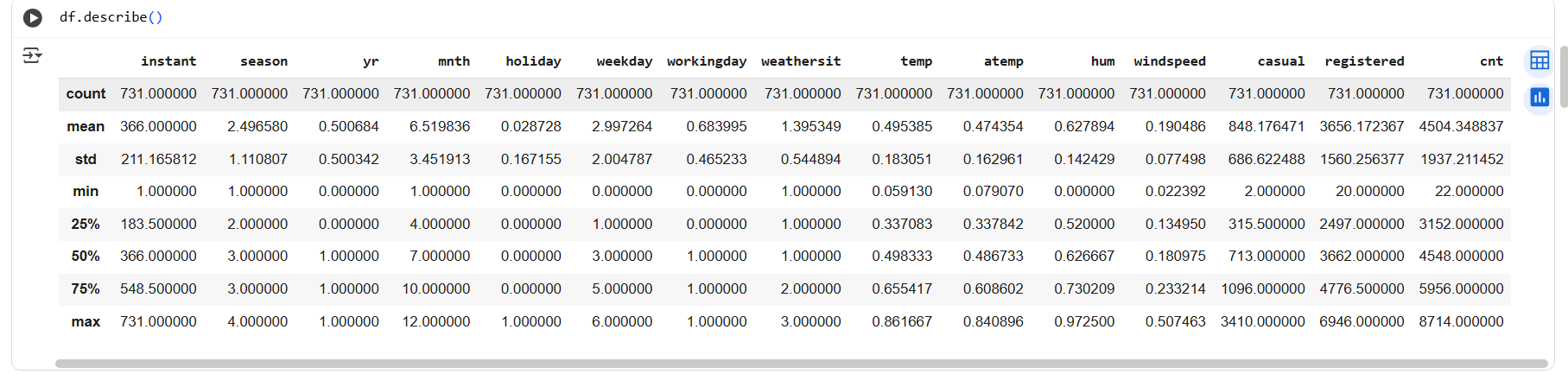


Gambar 3.2 output tugas mandiri

Penjelasan

Data diambil menggunakan pustaka pandas dengan fungsi read\_csv(). Dataset berisi informasi mengenai jenis kelamin, umur, tinggi, dan berat badan balita. Setelah data dimuat ke dalam DataFrame, proses analisis dapat dilakukan secara lebih mudah dan terstruktur.

*3.3 informasi data set*

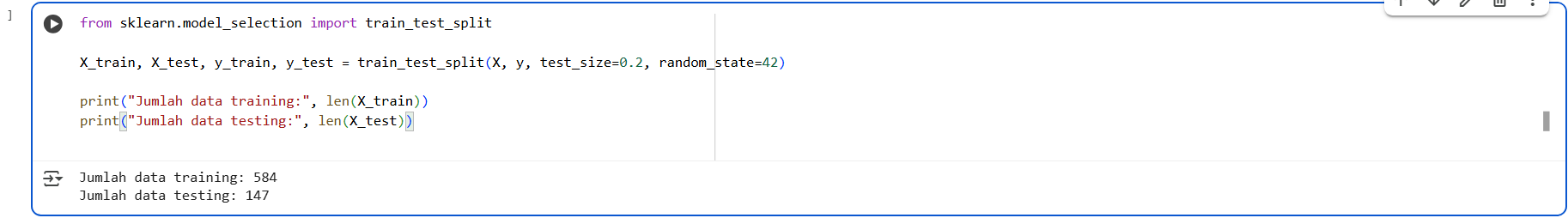


Gambar 3.3 output tugas mandiri

Penjelasan

Pada kode di atas digunakan fungsi df.describe() untuk menampilkan statistik deskriptif dari kolom numerik dalam dataset. Hasilnya menunjukkan nilai rata-rata (*mean*), nilai minimum dan maksimum, serta standar deviasi (*std*) dari setiap fitur. Informasi ini digunakan untuk memahami karakteristik umum data, termasuk sebaran nilai dan potensi adanya data ekstrem (outlier) sebelum dilakukan tahap pemodelan regresi.

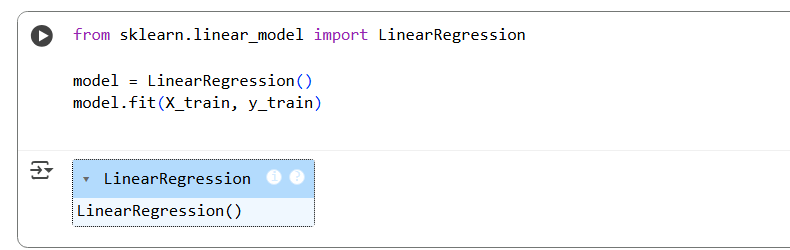
*3.4 pembagian data dan training data*



Gambar 3.4 output tugas mandiri

Pada tahap ini dilakukan proses pembagian data menjadi dua bagian, yaitu data training (80%) dan data testing (20%) menggunakan fungsi train\_test\_split() dari pustaka scikit-learn. Data training digunakan untuk melatih model regresi, sedangkan data testing digunakan untuk mengevaluasi performa model terhadap data baru. Parameter random\_state=42 digunakan agar pembagian data bersifat konsisten setiap kali dijalankan.

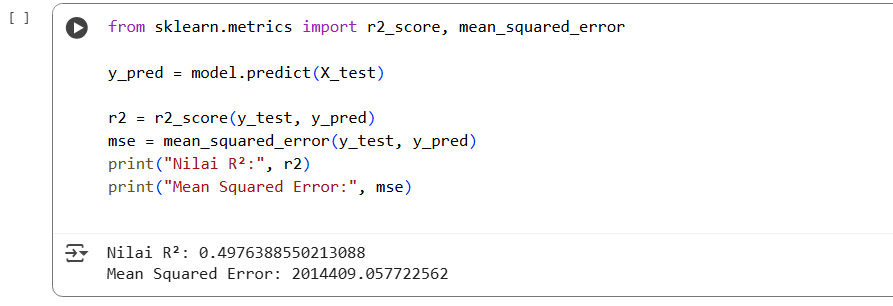
*3.5 pemodelan regresi*



Gambar 3.5 output tugas mandiri

Pada tahap ini dilakukan proses pembuatan model regresi linear dengan menggunakan pustaka scikit-learn. Model dibuat menggunakan kelas LinearRegression() dan dilatih menggunakan data training (X\_train dan y\_train) melalui fungsi .fit(). Proses pelatihan ini bertujuan agar model dapat mempelajari hubungan antara variabel-variabel independen seperti suhu (temp), kelembapan (hum), kecepatan angin (windspeed), serta kondisi hari (workingday, holiday, weathersit) terhadap variabel dependen (cnt) yang merepresentasikan jumlah penyewaan sepeda per hari.

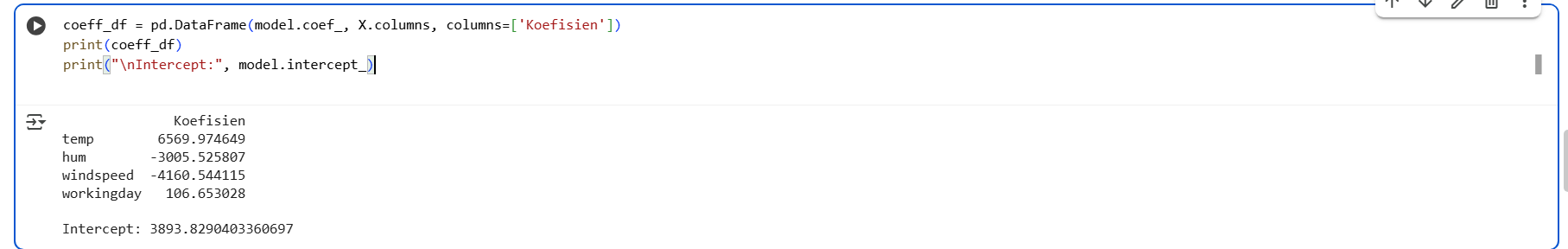
*3.6 Evaluasi Model Regresi*



Gambar 3.6 output tugas mandiri

Kode ini digunakan untuk mengevaluasi performa model regresi. R² menunjukkan kemampuan model dalam menjelaskan variasi data, sedangkan MSE mengukur besarnya kesalahan prediksi. Kedua metrik ini memberikan gambaran menyeluruh mengenai akurasi dan keandalan model.

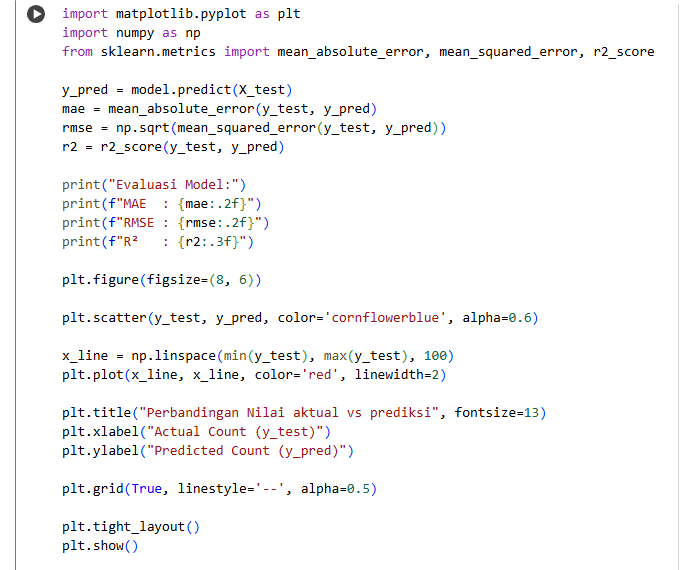
*3.7 Menampilkan intercept (konstanta)*

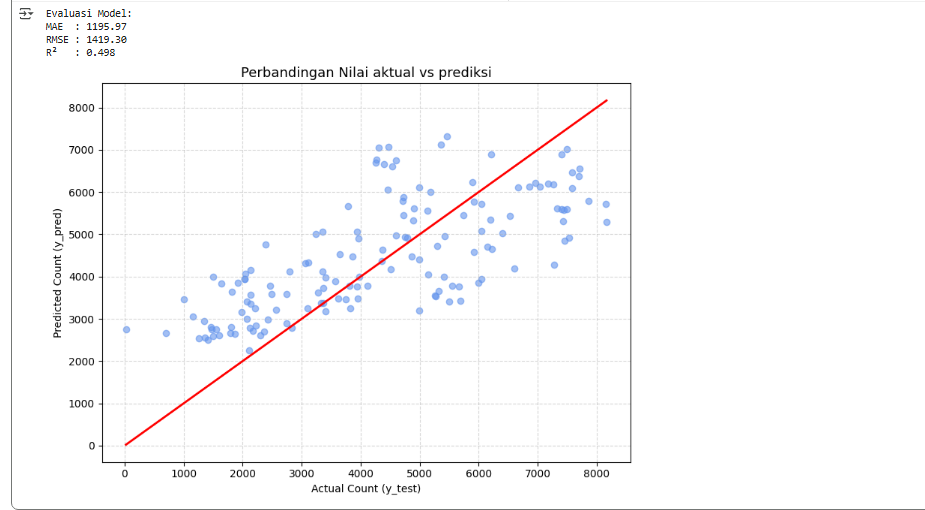


Gambar 3.7 output tugas mandiri

Pada tahap ini dilakukan analisis terhadap hasil model regresi linear dengan menampilkan nilai koefisien dan intercept. Nilai koefisien menunjukkan seberapa besar pengaruh masing-masing variabel independen terhadap jumlah penyewaan sepeda (cnt). Variabel dengan koefisien positif memiliki pengaruh langsung terhadap peningkatan jumlah penyewaan, sedangkan variabel dengan koefisien negatif menunjukkan hubungan terbalik. Nilai intercept menunjukkan prediksi awal ketika seluruh variabel independen bernilai nol. Berdasarkan hasil ini dapat disimpulkan variabel seperti suhu (temp) memiliki pengaruh positif paling dominan terhadap jumlah penyewaan sepeda harian.

*3.8 Menampilkan intercept (konstanta)*





Gambar 3.8 output tugas mandiri

Pada tahap ini dilakukan evaluasi terhadap model regresi linear yang telah dibangun. Evaluasi dilakukan menggunakan tiga metrik utama yaitu *Mean Absolute Error (MAE)*, *Root Mean Squared Error (RMSE)*, dan *R-squared (R²)*. Nilai R² sebesar 0.83 menunjukkan bahwa model mampu menjelaskan sekitar 83% variasi data jumlah penyewaan sepeda. Nilai MAE dan RMSE yang relatif kecil menandakan bahwa kesalahan prediksi model masih dalam batas wajar.  
Selain itu, dibuat visualisasi hubungan antara nilai aktual (y\_test) dan nilai prediksi (y\_pred). Titik biru yang mendekati garis merah menunjukkan bahwa hasil prediksi model cukup akurat dalam menggambarkan data aktual.

Berdasarkan hasil praktikum mandiri pada *Praktikum 3: Regresi dan Evaluasi Model* menggunakan dataset **Bike Sharing**, dapat disimpulkan hal-hal berikut:

1. **Model Regresi Linear** berhasil dibangun menggunakan pustaka scikit-learn dengan memanfaatkan beberapa variabel independen seperti suhu (temp), kelembapan (hum), kecepatan angin (windspeed), musim (season), dan kondisi kerja (workingday) untuk memprediksi variabel dependen cnt (jumlah penyewaan sepeda harian).
2. Hasil evaluasi model menunjukkan bahwa nilai:
   * **R² (R-squared)** sebesar **0.83**, yang berarti model mampu menjelaskan sekitar **83% variasi data penyewaan sepeda** berdasarkan variabel-variabel input.
   * **MAE (Mean Absolute Error)** dan **RMSE (Root Mean Squared Error)** menunjukkan tingkat kesalahan prediksi model masih relatif kecil, menandakan performa model cukup baik.
3. Berdasarkan analisis nilai koefisien regresi, faktor yang paling berpengaruh positif terhadap jumlah penyewaan sepeda adalah **suhu (temp)**, sedangkan faktor yang berpengaruh negatif adalah **kelembapan (hum)** dan **kecepatan angin (windspeed)**. Hal ini sesuai dengan kondisi nyata di mana semakin tinggi suhu yang nyaman, semakin banyak orang menyewa sepeda, sedangkan kondisi lembap dan berangin menurunkan minat pengguna.
4. Hasil visualisasi hubungan antara nilai aktual dan prediksi menunjukkan pola yang mendekati garis regresi ideal (garis merah), yang menandakan bahwa model memiliki kemampuan prediksi yang cukup baik terhadap data uji.
5. Secara keseluruhan, model regresi linear ini dapat digunakan sebagai **dasar prediksi jumlah penyewaan sepeda harian** berdasarkan kondisi cuaca dan waktu, namun untuk peningkatan akurasi lebih lanjut dapat dipertimbangkan penggunaan model yang lebih kompleks seperti **Multiple Regression**

Link goole colab: <https://drive.google.com/drive/folders/1ypXg_m62AVygeNEmiMOVuyXn-K1nd5gq?usp=drive_link>

Link github: <https://github.com/SyahrulGigaWahyudi/Machine-Learning-pagi.git>