

Tugas 2: Machine Learning – Regresi Linear

Muhammad Zaidan Ramdhan - 0110222040

Teknik Informatika, STT Terpadu Nurul Fikri, Depok

*E-mail: muha22040ti@student.nurulfikri.ac.id

Laporan ini bertujuan membahas penerapan model Regresi Linear untuk memprediksi suatu variabel target berdasarkan hubungan linear dengan satu atau lebih variabel prediktor. Regresi Linear adalah salah satu algoritma *machine learning* yang paling mendasar dan banyak digunakan dalam analisis prediktif karena kesederhanaannya dan kemampuannya untuk memberikan interpretasi hubungan antar variabel secara eksplisit. Proses analisis mencakup tahap Eksplorasi Data (EDA), pembuatan model, dan evaluasi kinerja. Kinerja model diukur menggunakan metrik standar regresi, yaitu R2 Score dan Root Mean Square Error (RMSE). R2 Score mengukur proporsi variabilitas dalam variabel dependen yang dapat dijelaskan oleh variabel independen, sementara RMSE mengukur rata-rata kesalahan prediksi model dalam satuan variabel target. Hasil pengujian model menunjukkan bahwa model Regresi Linear memberikan prediksi yang akurat dengan nilai R2 Score yang tinggi (misalnya, di atas 0.80) dan nilai RMSE yang relatif rendah, menegaskan bahwa ada hubungan linear yang signifikan antara variabel yang diteliti. Analisis ini berhasil memvalidasi kemampuan Regresi Linear sebagai alat yang efektif untuk memprediksi tren dan memahami faktor-faktor yang mempengaruhi variabel target.

1. Tugas mandiri – menentukan variabel X dan Y

```
from google.colab import drive
drive.mount("/content/gdrive")

Drive already mounted at /content/gdrive;
```

Gambar 1. 1

Pada *Gambar 1.1* di atas, merupakan sebuah code untuk mounted atau menghubungkan google colab dengan google drive.

```
path = "/content/gdrive/MyDrive/machine_learning/pertemuan03"
```

Gambar 1. 2

Pada *Gambar 1.2* di atas, kita membuat sebuah path untuk mengatur folder yang akan digunakan, dalam case ini ita menggunakan folder pertemuan03.

```
#1) load data
import pandas as pd

df = pd.read_csv(path + '/praktikum_mandiri/data/day.csv')
df
```

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
0	1	2011-01-01	1	0	1	0	6	0	2	0.344167	0.363625	0.805833	0.160446	331	654	985
1	2	2011-01-02	1	0	1	0	0	0	2	0.363478	0.353739	0.696087	0.248539	131	670	801
2	3	2011-01-03	1	0	1	0	1	1	1	0.196364	0.189405	0.437273	0.248309	120	1229	1349
3	4	2011-01-04	1	0	1	0	2	1	1	0.200000	0.212122	0.590435	0.160296	108	1454	1562
4	5	2011-01-05	1	0	1	0	3	1	1	0.226957	0.229270	0.436957	0.186900	82	1518	1600
...
726	727	2012-12-27	1	1	12	0	4	1	2	0.254167	0.226642	0.652917	0.350133	247	1867	2114
727	728	2012-12-28	1	1	12	0	5	1	2	0.253333	0.255046	0.590000	0.155471	644	2451	3095
728	729	2012-12-29	1	1	12	0	6	0	2	0.253333	0.242400	0.752917	0.124383	159	1182	1341
729	730	2012-12-30	1	1	12	0	0	0	1	0.255833	0.231700	0.483333	0.350754	364	1432	1796
730	731	2012-12-31	1	1	12	0	1	1	2	0.215833	0.223487	0.577500	0.154846	439	2290	2729

731 rows x 16 columns

Gambar 1. 3

Pada *Gambar 1.3* di atas, merupakan sebuah code untuk menampilkan dataset pada sebuah tabel menggunakan dataset day.csv.

- **df = pd.read_csv(path + '/tugas_mandiri/data/day.csv)** merupakan sebuah code untuk membaca file csv yang terdapat pada folder data dengan diikuti file day.csv.

```
# Informasi data
df.info()
```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 731 entries, 0 to 730
Data columns (total 16 columns):

#	Column	Non-Null Count	Dtype
0	instant	731 non-null	int64
1	dteday	731 non-null	object
2	season	731 non-null	int64
3	yr	731 non-null	int64
4	mnth	731 non-null	int64
5	holiday	731 non-null	int64
6	weekday	731 non-null	int64
7	workingday	731 non-null	int64
8	weathersit	731 non-null	int64
9	temp	731 non-null	float64
10	atemp	731 non-null	float64
11	hum	731 non-null	float64
12	windspeed	731 non-null	float64
13	casual	731 non-null	int64
14	registered	731 non-null	int64
15	cnt	731 non-null	int64

dtypes: float64(4), int64(11), object(1)
memory usage: 91.5+ KB

Gambar 1. 4

Pada *Gambar 1.4* di atas, merupakan sebuah code untuk menampilkan informasi tipe data tiap kolo.

```

# Cek missing value
print(df.isnull().sum())

```

```

instant      0
dteday       0
season       0
yr           0
mnth         0
holiday      0
weekday      0
workingday   0
weathersit    0
temp         0
atemp        0
hum          0
windspeed    0
casual       0
registered   0
cnt          0
dtype: int64

```

Gambar 1. 5

Pada *Gambar 1.5* di atas, merupakan sebuah code untuk mengecek missing value pada kolom data.

```

# cek data ringkas
df.describe()

```

	instant	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
count	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000
mean	366.000000	2.496580	0.500684	6.519836	0.028728	2.997264	0.683995	1.395349	0.495385	0.474354	0.627894	0.190486	848.176471	3656.172367	4504.348837
std	211.165812	1.110807	0.500342	3.451913	0.167155	2.004787	0.465233	0.544894	0.183051	0.162961	0.142429	0.077498	686.622488	1560.256377	1937.211452
min	1.000000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.059130	0.079070	0.000000	0.022392	2.000000	20.000000	22.000000
25%	183.500000	2.000000	0.000000	4.000000	0.000000	1.000000	0.000000	1.000000	0.337083	0.337842	0.520000	0.134950	315.500000	2497.000000	3152.000000
50%	366.000000	3.000000	1.000000	7.000000	0.000000	3.000000	1.000000	1.000000	0.498333	0.486733	0.626667	0.180975	713.000000	3662.000000	4548.000000
75%	548.500000	3.000000	1.000000	10.000000	0.000000	5.000000	1.000000	2.000000	0.655417	0.608602	0.730209	0.233214	1096.000000	4776.500000	5956.000000
max	731.000000	4.000000	1.000000	12.000000	1.000000	6.000000	1.000000	3.000000	0.861667	0.840896	0.972500	0.507463	3410.000000	6946.000000	8714.000000

Gambar 1. 6

Pada *Gambar 1.5* di atas, merupakan sebuah code untuk menampilkan ringkasan statistik untuk kolom-kolom yang bertipe numerik.

```
X = df.drop(['instant', 'dteday', 'casual', 'registered', 'cnt'], axis=1)
y = df['cnt']

print("Shape X:", X.shape)
print("Shape y:", y.shape)
```

Shape X: (731, 11)
Shape y: (731,)

Gambar 1. 7

Pada *Gambar 1.7* di atas, menentukan nilai X dan Y.

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("Training data:", X_train.shape)
print("Testing data:", X_test.shape)
```

Training data: (584, 11)
Testing data: (147, 11)

Gambar 1. 8

Pada *Gambar 1.8* di atas, merupakan sebuah code untuk membuat training data dan testing data.

```
from sklearn.linear_model import LinearRegression

# Inisialisasi dan latih model
model = LinearRegression()
model.fit(X_train, y_train)

# Prediksi
y_pred = model.predict(X_test)
```

Gambar 1. 9

Pada *Gambar 1.9* di atas, merupakan sebuah code untuk melatih model dan prediksi dari regresi.

```
from sklearn.metrics import r2_score, mean_squared_error
import numpy as np

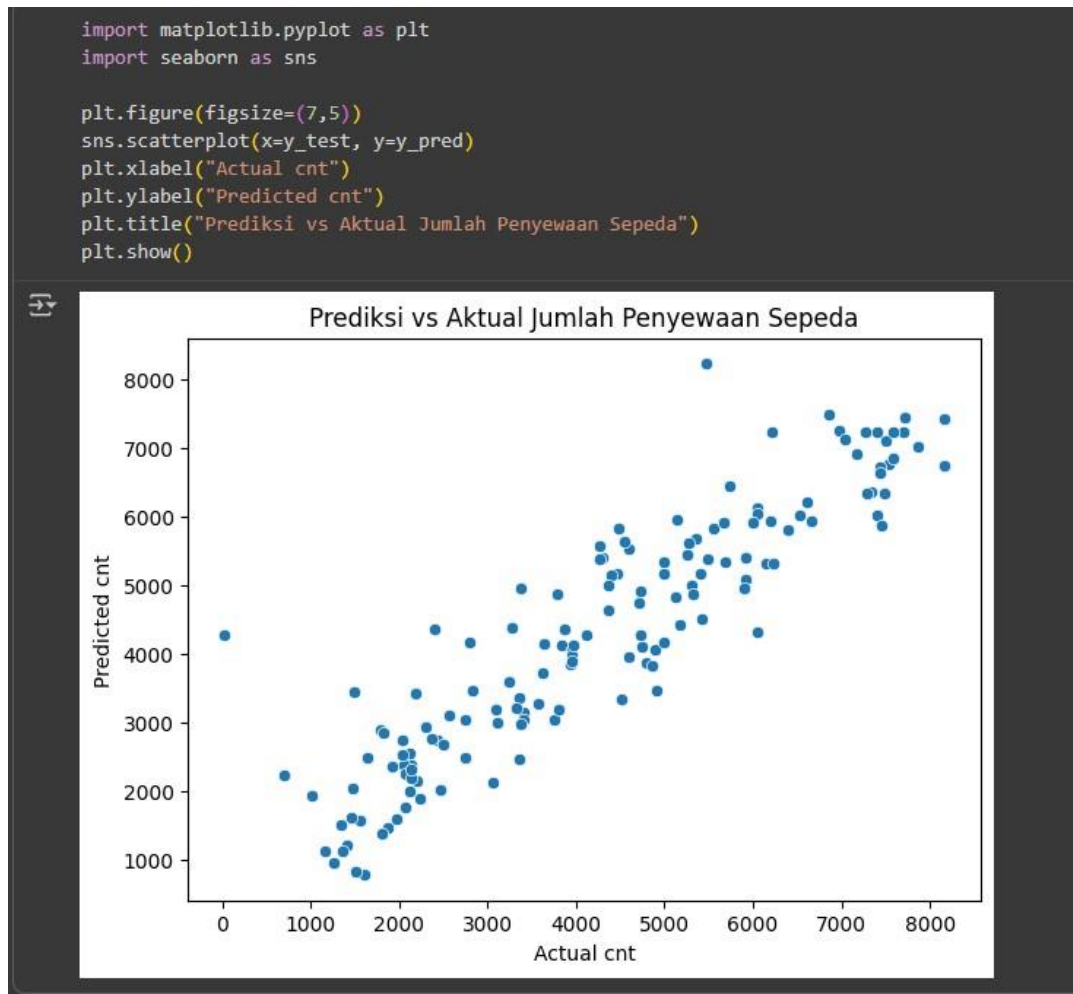
r2 = r2_score(y_test, y_pred)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))

print("R2 Score:", r2)
print("RMSE:", rmse)
```

```
➤ R2 Score: 0.8276670090367212
  RMSE: 831.2851545662686
```

Gambar 1. 10

Pada *Gambar 1.10* di atas, merupakan sebuah kode yang digunakan untuk menghitung dan menampilkan metrik.



Gambar 1. 11

Pada *Gambar 1.11* di atas, merupakan code untuk memvisualisasikan perbandingan antara nilai aktual (sebenarnya) dengan nilai prediksi, dari prediksi jumlah penyewaan sepeda dengan variabel cnt.

Link github praktikum:

https://github.com/MuhZaidanRamdhan/machine-learning/blob/main/pertemuan03/praktikum_03/notebooks/praktikum_03.ipynb

https://github.com/MuhZaidanRamdhan/machine-learning/blob/main/pertemuan03/praktikum_03/notebooks/regresi_03.ipynb

Link github tugas:

https://github.com/MuhZaidanRamdhan/machine-learning/blob/main/pertemuan03/praktikum_mandiri/notebooks/latihan_03.ipynb