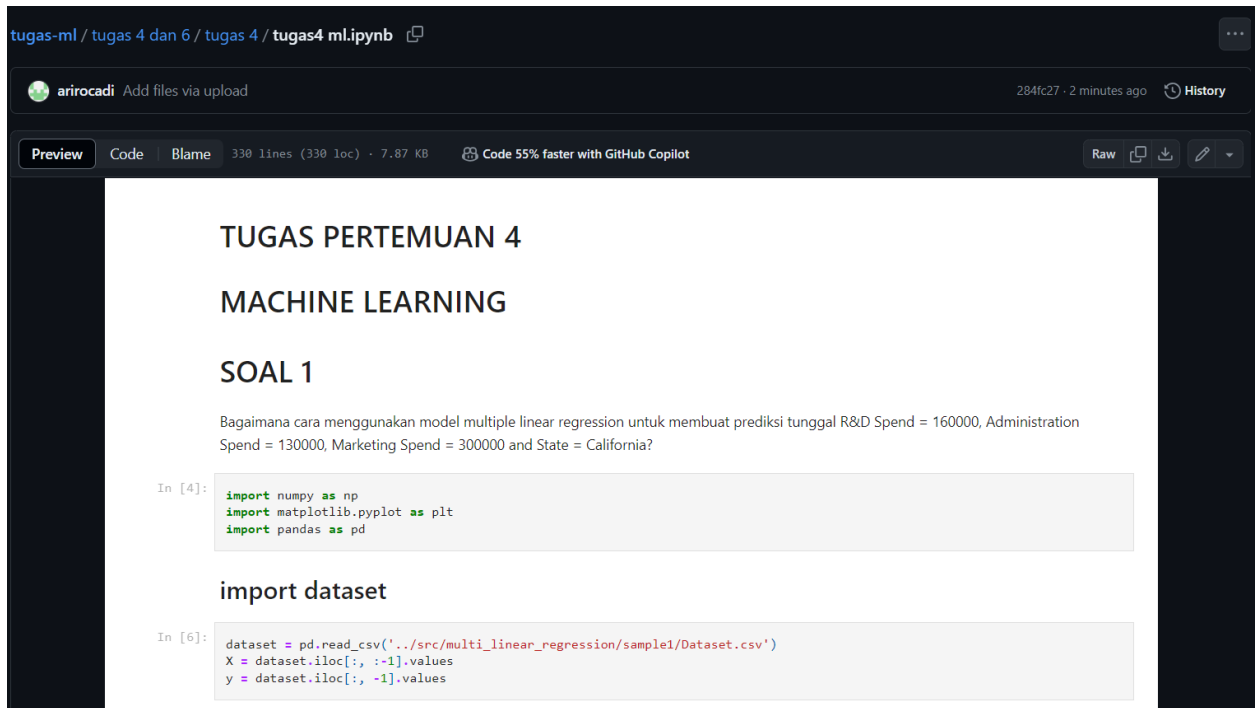


**Nama : Ari Rocadi**  
**NIM : 20220801339**  
**Mata Kuliah : Machine Learning**

---

## Tugas Pertemuan 4

### 1. Jawaban



```
tugas-ml / tugas 4 dan 6 / tugas 4 / tugas4 ml.ipynb
```

ariocadi Add files via upload 284fc27 · 2 minutes ago History

Preview Code Blame 330 lines (330 loc) · 7.87 KB Code 55% faster with GitHub Copilot Raw

## TUGAS PERTEMUAN 4

## MACHINE LEARNING

### SOAL 1




Bagaimana cara menggunakan model multiple linear regression untuk membuat prediksi tunggal R&D Spend = 160000, Administration Spend = 130000, Marketing Spend = 300000 and State = California?

```
In [4]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

### import dataset

```
In [6]: dataset = pd.read_csv('../src/multi_linear_regression/sample1/Dataset.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

tugas-ml / tugas 4 dan 6 / tugas 4 / tugas4 ml.ipynb ↑ Top

Preview Code Blame 330 lines (330 loc) · 7.87 KB Code 55% faster with GitHub Copilot Raw   

### print dataset

```
In [7]: dataset.head()
```

```
Out[7]:
```

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	Florida	166187.94




### Encoding categorical data

```
In [8]: from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [3])], remainder='passthrough')
X = np.array(ct.fit_transform(X))
```

### Splitting the dataset into the Training set and Test set

```
In [9]: 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 = 0)
```

tugas-ml / tugas 4 dan 6 / tugas 4 / tugas4 ml.ipynb ↑ Top

Preview Code Blame 330 lines (330 loc) · 7.87 KB Code 55% faster with GitHub Copilot Raw   

### Training the Multiple Linear Regression model on the Training set

```
In [19]: from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

```
Out[19]: LinearRegression()
```

### Predict Test Result

```
In [20]: y_pred = regressor.predict(X_test)
np.set_printoptions(precision=2)
print(np.concatenate((y_pred.reshape(len(y_pred),1), y_test.reshape(len(y_test),1)),1))
```

```
[[103015.2  103282.38]
 [132582.28 144259.4 ]
 [132447.74 146121.95]
 [ 71976.1   77798.83]
 [178537.48 191050.39]
 [116161.24 105008.31]
 [ 67851.69  81229.06]
 [ 98791.73  97483.56]
 [113969.44 110352.25]
 [167921.07 166187.94]]
```

[103282.38 100107.94]]

**JAWABAN R&D Spend = 160000, Administration Spend = 130000, Marketing Spend = 300000 and State = 'California'**

```
In [23]: print(regressor.predict([[1, 0, 0, 160000, 130000, 300000]]))
```

```
[181566.92]
```

## 2. Jawaban

### SOAL 2

Bagaimana cara mendapatkan persamaan regresi akhir  $y = b_0 + b_1 x_1 + b_2 x_2 + \dots$  dengan nilai akhir koefisien?

```
In [21]: print(regressor.coef_)  
         print(regressor.intercept_)  
  
[ 8.66e+01 -8.73e+02  7.86e+02  7.73e-01  3.29e-02  3.66e-02]  
42467.52924856474
```

Link Github

<https://github.com/ariocadi/tugas-ml/blob/main/tugas%202%20ml.ipynb>