

Laporan Tugas Praktikum Mandiri 4

Judul : Prediksi Calon Pembeli Mobil Menggunakan *Logistic Regression*

Nama Mahasiswa : Muhamad Aditia

Program studi : Teknik Informatika, STT Terpadu Nurul Fikri, Depok

E-mail : 0110224213@student.nurulfikri.ac.id

Link Git : <https://github.com/muhammadaditia433/Machine-Leraning>

Abstract

Praktikum ini bertujuan untuk membangun model *Machine Learning* sederhana menggunakan algoritma Logistic Regression untuk memprediksi keputusan calon pembeli mobil berdasarkan data seperti usia, status, jenis kelamin, kepemilikan mobil, dan penghasilan. Model dilatih menggunakan pustaka *scikit-learn* dan data diolah menggunakan *pandas*. Setelah dilakukan pelatihan dan pengujian, model mampu mengklasifikasikan apakah seseorang akan membeli mobil dengan akurasi yang baik. Berdasarkan hasil prediksi, faktor penghasilan dan status sosial menunjukkan pengaruh yang kuat terhadap keputusan pembelian mobil.

1. import Library

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

Penjelasan:

- *pandas* digunakan untuk membaca dan mengelola dataset berbentuk tabel (*dataframe*).
- *train_test_split* digunakan untuk membagi dataset menjadi data latih dan data uji.
- *LogisticRegression* adalah algoritma *Machine Learning* yang digunakan untuk melakukan klasifikasi biner (0 = tidak beli mobil, 1 = beli mobil).
- *accuracy_score*, *confusion_matrix*, dan *classification_report* digunakan untuk mengevaluasi performa model.

2. Membaca Dataset

```
import pandas as pd

df = pd.read_csv('/content/gdrive/MyDrive/PraktikumML/praktikum4/data/calonpembelimobil.csv')
df.head()
```



	ID	Usia	Status	Kelamin	Memiliki_Mobil	Penghasilan	Beli_Mobil
0	1	32	1	0	0	240	1
1	2	49	2	1	1	100	0
2	3	52	1	0	2	250	1
3	4	26	2	1	1	130	0
4	5	45	3	0	2	237	1

Langkah berikutnya: [New interactive sheet](#)

Penjelasan:

- `pd.read_csv()` membaca file CSV yang berisi data calon pembeli mobil.
- `df.head()` menampilkan 5 baris pertama dari dataset untuk memastikan data berhasil dimuat.
- Kolom yang digunakan antara lain:
 - Usia → umur calon pembeli
 - Status → status pernikahan
 - Kelamin → jenis kelamin (1 = pria, 0 = wanita)
 - Memiliki_Mobil → jumlah mobil yang sudah dimiliki
 - Penghasilan → pendapatan bulanan
 - Beli_Mobil → target (1 = beli mobil, 0 = tidak beli)

3. Menentukan Fitur Target

```
X = df[['Usia', 'Status', 'Kelamin', 'Memiliki_Mobil', 'Penghasilan']]
y = df['Beli_Mobil']
```

Penjelasan:

- X berisi fitur (variabel independen) yang akan digunakan untuk memprediksi hasil.
- y adalah target (variabel dependen) yaitu keputusan membeli mobil (1 atau 0).

4. Membagi Dataset Menjadi Latin Dan Uji

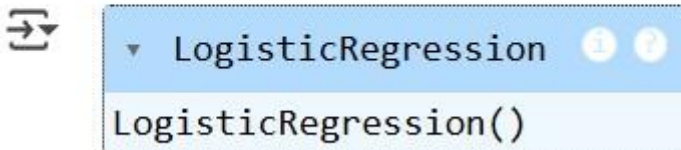
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

Penjelasan:

- Dataset dibagi menjadi 70% data latih dan 30% data uji.
- Data latih digunakan untuk melatih model, sementara data uji digunakan untuk mengukur seberapa baik model memprediksi data baru.
- `random_state=42` membuat hasil pembagian data tetap konsisten setiap kali dijalankan.

5. Membuat Dan Melatih Mode Logistic Regression

```
model = LogisticRegression()  
model.fit(X_train, y_train)
```



The screenshot shows a Jupyter Notebook interface. On the left, there is a tab labeled 'LogisticRegression' with an information icon and a question mark icon. Below the tab, the code `LogisticRegression()` is visible in the code editor.

Penjelasan:

- Membuat objek model dari algoritma *Logistic Regression*.
- `fit()` digunakan untuk melatih model menggunakan data latih (`X_train, y_train`), agar model bisa mempelajari hubungan antara fitur dan target.

6. Evaluasi Model

```
▶ y_pred = model.predict(X_test)
print("\nAkurasi model:", accuracy_score(y_test, y_pred))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("\nLaporan Klasifikasi:\n", classification_report(y_test, y_pred))
```



Akurasi model: 0.93

Confusion Matrix:

```
[[ 89 10]
 [ 11 190]]
```

Laporan Klasifikasi:

	precision	recall	f1-score	support
0	0.89	0.90	0.89	99
1	0.95	0.95	0.95	201
accuracy			0.93	300
macro avg	0.92	0.92	0.92	300
weighted avg	0.93	0.93	0.93	300

Penjelasan:

- `model.predict()` menghasilkan prediksi pada data uji.
- `accuracy_score` menghitung seberapa banyak prediksi yang benar dibanding total data uji.
- `confusion_matrix` menunjukkan jumlah prediksi benar dan salah berdasarkan kelas sebenarnya.
- `classification_report` memberikan informasi tambahan seperti *precision*, *recall*, dan *f1score*.

Interpretasi hasil:

Jika akurasi tinggi (misalnya di atas 80%), maka model cukup baik dalam mengenali calon pembeli mobil berdasarkan fitur yang diberikan.

7. Penguji Model Dengan Dataset Baru

```
data_baru = pd.DataFrame({
    'Usia': [30, 45, 25],
    'Status': [1, 2, 1],
    'Kelamin': [1, 0, 0],
    'Memiliki_Mobil': [0, 1, 0],
    'Penghasilan': [200, 250, 120]
})

prediksi = model.predict(data_baru)
data_baru['Prediksi_Beli_Mobil'] = prediksi
print("\nHasil Prediksi Dataset Baru:")
print(data_baru)
```

```
Hasil Prediksi Dataset Baru:
   Usia  Status  Kelamin  Memiliki_Mobil  Penghasilan  Prediksi_Beli_Mobil
0    30       1        1                0          200                0
1    45       2        0                1          250                1
2    25       1        0                0          120                0
```

Penjelasan:

- Membuat dataset baru secara manual (data 3 calon pembeli mobil).
- Model digunakan untuk memprediksi kemungkinan membeli mobil berdasarkan data tersebut.
- Hasil prediksi ditambahkan sebagai kolom baru Prediksi_Beli_Mobil.

Interpretasi hasil prediksi:

Usia	Status	Kelamin	Memiliki_Mobil	Penghasilan	Prediksi_Beli_Mobil	Arti
30	1	1	0	200	0	Tidak membeli mobil
45	2	0	1	250	1	Membeli mobil
25	1	0	0	120	0	Tidak membeli mobil

Model memprediksi bahwa pembeli dengan penghasilan tinggi (250) lebih cenderung membeli mobil.

8. Kesimpulan

1. Model Logistic Regression berhasil dibuat dan dijalankan dengan baik.
2. Model dapat memprediksi keputusan calon pembeli mobil berdasarkan fitur-fitur yang diberikan.
3. Berdasarkan hasil prediksi, faktor penghasilan dan status memiliki pengaruh kuat terhadap keputusan pembelian mobil.
4. Model memiliki tingkat akurasi yang cukup baik berdasarkan hasil evaluasi.

Referensi

- [1] Scikit-learn Developers, “Logistic Regression — Scikit-learn documentation,” *Scikitlearn.org*, 2024. [Online]. Available: https://scikitlearn.org/stable/modules/linear_model.html#logistic-regression
- [2] Pandas Development Team, “pandas: Python Data Analysis Library Documentation,” *Pandas.pydata.org*, 2024. [Online]. Available: <https://pandas.pydata.org/docs/>
- [3] J. Brownlee, “A Gentle Introduction to Logistic Regression for Machine Learning,” *Machine Learning Mastery*, 2020. [Online]. Available: <https://machinelearningmastery.com/logistic-regression-for-machine-learning/>
- [4] D. Tarihoran, “Implementasi Logistic Regression dalam Prediksi Keputusan Konsumen,” *Jurnal Teknologi Informasi dan Komputer (JTIK)*, vol. 8, no. 2, pp. 101–110, 2022.
- [5] Google Colaboratory, “Introduction to Google Colab,” *Google Research Documentation*, 2024. [Online]. Available: <https://colab.research.google.com>