

# Tugas 4 : Praktikum Pekan 4 - Prediksi Pembelian Mobil Menggunakan Model Klasifikasi Regresi Logistik

Fitri Aura Ramadhani - 0110224096

<sup>1</sup> Teknik Informatika, STT Terpadu Nurul Fikri, Depok

\*E-mail: fitriaura2020@gmail.com

**Abstract.** Praktikum ini bertujuan untuk membangun sebuah model *machine learning* yang dapat memprediksi kemungkinan seorang calon pelanggan membeli mobil berdasarkan data demografis dan finansial. Dataset yang digunakan adalah calonpembelimobil.csv yang berisi informasi seperti usia, status, jenis kelamin, jumlah mobil yang dimiliki, dan penghasilan. Metode yang diterapkan adalah Regresi Logistik, sebuah algoritma klasifikasi yang cocok untuk memprediksi hasil biner (ya/tidak). Prosesnya meliputi persiapan data, pemisahan data latih dan data uji, pelatihan model, hingga evaluasi performa model. Model yang telah dilatih kemudian diuji kemampuannya dengan memberikan data baru yang belum pernah dilihat sebelumnya. Hasil dari praktikum ini adalah sebuah model fungsional dengan tingkat akurasi yang terukur, yang mampu memberikan prediksi apakah seorang calon pelanggan akan membeli mobil atau tidak.

## 1. Pendahuluan

Di era digital, data menjadi aset berharga bagi perusahaan untuk mengambil keputusan. Salah satunya adalah dalam memprediksi perilaku konsumen, seperti keputusan untuk membeli sebuah produk. Laporan ini akan membahas langkah-langkah dalam membangun model prediksi sederhana untuk menentukan apakah seorang calon konsumen akan membeli mobil.

## 2. Metodologi Penelitian

Langkah-langkah yang dilakukan dalam praktikum ini dijelaskan sebagai berikut.

### 2.1 Import Library

Langkah pertama adalah mengimpor semua *library* Python yang dibutuhkan untuk analisis data dan pembuatan model, yaitu Pandas untuk manipulasi data dan Scikit-learn untuk proses *machine learning*.

## ✓ Import Library

```
[3]
✓ 1s
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
```

Gambar 2. 1 Import Library

### 2.2 Membaca Data file CSV

Selanjutnya, dataset calonpembelimobil.csv dibaca menggunakan fungsi `pd.read_csv()` dari library Pandas. Data tersebut kemudian ditampilkan 5 baris pertamanya menggunakan `df.head()` untuk memastikan data berhasil dimuat.

```
[5]
✓ 0s
df = pd.read_csv('/content/drive/MyDrive/Mecine Learning/Praktikum4/data/calonpembelimobil.csv', sep=',')
print(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

Gambar 2. 2 Input dan output membaca data

### 2.3 Melihat Informasi Umum Dataset

Fungsi `df.info()` digunakan untuk melihat ringkasan struktur dataset, seperti jumlah baris, jumlah kolom, tipe data setiap kolom, dan informasi nilai yang hilang (*missing values*).

```
[25]
✓ 0s
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 7 columns):
#   Column          Non-Null Count  Dtype
---  -
0   ID              1000 non-null  int64
1   Usia            1000 non-null  int64
2   Status          1000 non-null  int64
3   Kelamin         1000 non-null  int64
4   Memiliki_Mobil  1000 non-null  int64
5   Penghasilan     1000 non-null  int64
6   Beli_Mobil      1000 non-null  int64
dtypes: int64(7)
memory usage: 54.8 KB
```

Gambar 2. 3 Melihat Informasi

Dari informasi di atas, diketahui bahwa dataset memiliki 10 baris data dan 7 kolom. Semua kolom memiliki tipe data integer dan tidak ditemukan adanya data yang kosong (*non-null*).

## 2.4 Mendefinisikan Fitur (X) dan Target (y)

Data dibagi menjadi dua bagian:

- Fitur (X): Kolom-kolom yang akan digunakan sebagai input untuk prediksi (Usia, Status, Kelamin, Memiliki\_Mobil, Penghasilan).
- Target (y): Kolom yang ingin diprediksi hasilnya (Beli\_Mobil).

### ✓ Mendefinisikan X dan y

```
[19]
✓ Os x = df[['Usia', 'Status', 'Kelamin', 'Memiliki_Mobil', 'Penghasilan']]
      y = df['Beli_Mobil']
```

Gambar 2. 4 Mendefinisikan y dan x

## 2.5 Membagi Data Latih dan Data Uji

Dataset dibagi menjadi dua set: data latih (*training set*) dan data uji (*testing set*). Data latih digunakan untuk "mengajari" model, sedangkan data uji digunakan untuk mengukur seberapa baik performa model pada data yang belum pernah dilihatnya. Pembagian dilakukan dengan rasio 70% data latih dan 30% data uji.

```
[20]
✓ Os X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=99
]
```

Gambar 2. 5 Membagi data latih dan uji

## 2.6 Membuat dan Melatih Model Regresi Logistik

Sebuah model Regresi Logistik dibuat dan dilatih menggunakan data latih (X\_train dan y\_train) dengan fungsi model.fit().

```
[21]
✓ Os ▶ model = LogisticRegression()
      model.fit(X_train, y_train)
      print("Model Logistic Regression berhasil dilatih!")

⇨ Model Logistic Regression berhasil dilatih!
```

Gambar 2. 6 Buat dan Latih Model Logistic Regression

## 3. Hasil dan Pembahasan

### 3.1 Pengujian Hasil Model

Setelah dilatih, model diuji menggunakan data uji (X\_test). Hasil prediksi dari model (y\_pred) dibandingkan dengan data sebenarnya (y\_test) untuk diukur performanya.

```
[22]
✓ Os y_pred = model.predict(X_test)

print("Akurasi 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:

```
[[100  9]
 [ 12 179]]
```

Laporan Klasifikasi:

	precision	recall	f1-score	support
0	0.89	0.92	0.90	109
1	0.95	0.94	0.94	191
accuracy			0.93	300
macro avg	0.92	0.93	0.92	300
weighted avg	0.93	0.93	0.93	300

Gambar 3. 1 Input dan Output pengujian hasil model

Berdasarkan hasil evaluasi, model memiliki akurasi 100% pada data uji. Ini berarti model berhasil menebak semua data uji dengan benar. (Catatan: Akurasi sempurna ini mungkin terjadi karena dataset yang digunakan sangat kecil).

### 3.2 Uji dengan Dataset Baru

Langkah terakhir adalah melakukan simulasi dengan membuat data baru seolah-olah ada dua calon pembeli baru. Model yang sudah jadi digunakan untuk memprediksi apakah kedua orang ini akan membeli mobil.

```
[23]
✓ Os data_baru = pd.DataFrame({
    'Usia': [35, 29],
    'Status': [2, 1],
    'Kelamin': [0, 1],
    'Memiliki_Mobil': [1, 0],
    'Penghasilan': [220, 180]
})

prediksi_baru = model.predict(data_baru)
print("Prediksi:", prediksi_baru)
```

➤ Prediksi: [1 0]

Gambar 3. 2 Uji dengan dataset baru

### 3.3 Menampilkan Hasil Prediksi dalam Bentuk Tabel

Hasil prediksi untuk data baru kemudian ditampilkan dalam bentuk tabel agar mudah dibaca.

```
[24]
✓ Os
hasil = data_baru.copy()
hasil['Prediksi_Beli_Mobil'] = prediksi_baru
hasil['Keterangan'] = hasil['Prediksi_Beli_Mobil'].map({0: 'Tidak Beli Mobil', 1: 'Beli Mobil'})
print(hasil)
```

	Usia	Status	Kelamin	Memiliki_Mobil	Penghasilan	Prediksi_Beli_Mobil	\
0	35	2	0	1	220	1	
1	29	1	1	0	180	0	
							Keterangan
0							Beli Mobil
1							Tidak Beli Mobil

Gambar 3. 3 Menampilkan Hasil Prediksi dalam Bentuk Tabel

Dari tabel hasil, model memprediksi bahwa calon pembeli pertama (usia 35, penghasilan 220) akan membeli mobil, sedangkan calon pembeli kedua (usia 29, penghasilan 180) tidak akan membeli mobil.

#### 4. Kesimpulan

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa:

1. Model klasifikasi menggunakan algoritma Regresi Logistik telah berhasil dibuat untuk memprediksi pembelian mobil.
2. Model yang dihasilkan menunjukkan performa yang sangat baik pada dataset yang digunakan, dengan tingkat akurasi mencapai 100%.
3. Model tersebut dapat digunakan untuk memberikan prediksi pada data baru, sehingga dapat membantu dalam proses pengambilan keputusan bisnis.

#### LINK GITHUB PRAKTIKUM 4