

Tugas 4: Tugas Praktikum Mandiri Machine Learning

Shapiere Januar Rafiansyah - 0110224191

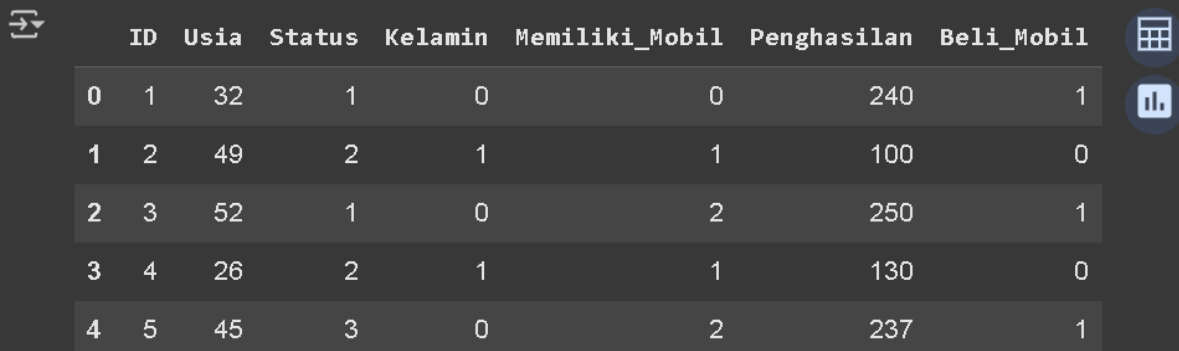
¹ Teknik Informatika, STT Terpadu Nurul Fikri, Depok

*E-mail: shapierejanuarr@gmail.com

1. Tugas Praktikum Mandiri 4

```
# 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

# membaca dataset
df = pd.read_csv(path + "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

Library pandas dipakai buat ngolah data, train_test_split buat bagi data latih dan uji, LogisticRegression buat bikin model prediksi, dan accuracy_score buat ngukur akurasi.

Dan di bawahnya berfungsi buat baca dataset dan nampilin 5 baris awal. Dari output-nya, keliatan struktur datanya lengkap: Usia, Status, Kelamin, Memiliki_Mobil, Penghasilan, dan Beli_Mobil. Kolom Beli_Mobil jadi target (Y), sedangkan sisanya jadi fitur (X).

```
# pisahkan fitur dan target
X = df[['Usia', 'Status', 'Kelamin', 'Memiliki_Mobil', 'Penghasilan']]
y = df['Beli_Mobil']
```

Bagian kode ini fungsinya buat misahin variabel fitur (X) dan target (y) sebelum data dipakai buat pelatihan model.

- X berisi kolom yang jadi faktor penentu seseorang beli mobil: Usia, Status, Kelamin, Memiliki_Mobil, dan Penghasilan.
- y cuma ambil kolom Beli_Mobil sebagai variabel target, hasil yang mau diprediksi model (1 = beli, 0 = tidak beli).

Langkah ini penting karena model machine learning harus tahu mana data yang jadi input dan mana yang jadi output biar bisa belajar dengan benar.

```
# Bagi dataset: 80% training, 20% testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Baris ini dipakai buat membagi dataset jadi dua bagian utama menjadi data latih (*training*) dan data uji (*testing*).

- `train_test_split()` dari sklearn otomatis ngebagi data, di sini 80% buat pelatihan model (`X_train, y_train`) dan 20% buat ngetes hasilnya (`X_test, y_test`).
- Parameter `random_state=42` biar hasil pembagiannya tetap sama setiap dijalankan.

Langkah ini penting supaya model bisa belajar dari sebagian data dan diuji di data baru buat ngukur seberapa bagus performanya dalam memprediksi.

```
# Buat model regresi logistik
model = LogisticRegression()
model.fit(X_train, y_train)
```



▼ LogisticRegression ⓘ ?

LogisticRegression()

Bagian ini digunakan untuk membangun dan melatih model regresi logistik.

- `LogisticRegression()` membuat objek model yang siap digunakan untuk klasifikasi biner (dalam kasus ini: “beli mobil” atau “tidak beli mobil”).
- `model.fit(X_train, y_train)` melatih model dengan data training yang udah dipisah sebelumnya.

```
# Evaluasi model
akurasi = model.score(X_test, y_test)
print("Akurasi Model:", round(akurasi * 100, 2), "%")
```

➞ Akurasi Model: 93.0 %

Bagian ini berfungsi buat mengevaluasi seberapa bagus performa model setelah dilatih.

- `model.score(X_test, y_test)` menghitung akurasi model dengan membandingkan hasil prediksi terhadap data asli di dataset testing.
- Nilainya dikali 100 biar jadi persentase, dan dibulatkan dua angka di belakang koma.

Hasil akurasi 93% menunjukkan model mampu memprediksi keputusan pembelian mobil dengan tingkat ketepatan yang sangat baik.

```
# Uji prediksi data baru
data_baru = pd.DataFrame({
    'Usia': [35],
    'Status': [1],
    'Kelamin': [1],
    'Memiliki_Mobil': [0],
    'Penghasilan': [7500000]
})

prediksi = model.predict(data_baru)
print("Hasil Prediksi:", "Beli Mobil" if prediksi[0] == 1 else "Tidak Beli Mobil")
```

➞ Hasil Prediksi: Beli Mobil

Bagian ini digunakan buat mengukur kemampuan model dalam memprediksi kasus baru yang belum pernah dilihat sebelumnya.

- Data baru (`data_baru`) dibuat dalam bentuk `DataFrame` berisi input dengan nilai tertentu, misalnya usia 35 tahun, status 1, kelamin 1, belum punya mobil, dan penghasilan 7.500.000.
- `model.predict(data_baru)` lalu digunakan untuk menebak apakah orang dengan profil seperti ini akan membeli mobil atau tidak.
- Output yang muncul adalah “Beli Mobil”, artinya model memprediksi bahwa individu tersebut berpotensi melakukan pembelian.

Secara kritis, bagian ini nunjukin kalau model bukan cuma bisa belajar dari data lama, tapi juga bisa diimplementasikan untuk pengambilan keputusan nyata. misalnya buat membantu dealer menargetkan calon pembeli yang potensial.

2. Kesimpulan

Dari praktikum ini bisa disimpulkan kalau regresi logistik cukup efektif buat klasifikasi, contohnya memprediksi calon pembeli mobil. Dengan akurasi 93%, modelnya udah tergolong bagus. Tapi secara kritis, hasil tinggi belum tentu berarti modelnya sempurna.

3. GitHub

<https://github.com/Shapiere/TugasMandiri-Praktikum-MachineLearning.git>