

**MODUL PRAKTIKUM
BUSINESS INTELLIGENCE**

MODUL KE : _____

NAMA : _____

NPM : _____

KELAS: _____

INSTITUT INFORMATIKA DAN BISNIS DARMAJAYA

2025

MODUL IX

KLASIFIKASI PELANGGAN MENGGUNAKAN PYTHON UNTUK MENDUKUNG KEPUTUSAN BISNIS

TUJUAN:

Setelah mengikuti praktikum ini, mahasiswa mampu:

1. Mengimport dataset pelanggan ke Python.
2. Melakukan pemrosesan data dasar menggunakan pandas.
3. Membangun model klasifikasi sederhana:
 - Logistic Regression
 - Decision Tree
 - Naive Bayes
4. Menghitung Confusion Matrix, Accuracy, Precision, Recall, dan F1-Score.
5. Mengaitkan hasil model dengan pengambilan keputusan bisnis terkait promosi.

DATA SOURCE

Gunakan dataset berikut yang telah disediakan dosen:

File: dataset_klasifikasi_modul9.csv

Jumlah baris: 100

Struktur kolom:

- CustomerID
- Usia
- Pendapatan
- FrekuensiBelanja
- TotalBelanja
- TrenPenjualan (Naik/Turun/Stabil)
- Respon_Promo (1 = merespon promo, 0 = tidak)

I. BAGIAN PRAKTIKUM 1 – PERSIAPAN LINGKUNGAN & IMPORT DATA

1) Buka Google Colab

- Pergi ke <https://colab.research.google.com>

- Klik New Notebook

2) Upload Dataset

Klik ikon Folder → Upload → pilih dataset_klasifikasi_modul9.csv

3) Import Library

```
import pandas as pd
```

```
import numpy as np
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.preprocessing import StandardScaler, LabelEncoder
```

```
from sklearn.linear_model import LogisticRegression
```

```
from sklearn.tree import DecisionTreeClassifier
```

```
from sklearn.naive_bayes import GaussianNB
```

```
from sklearn.metrics import confusion_matrix, classification_report
```

4) Baca Dataset

```
df = pd.read_csv('dataset_klasifikasi_modul9.csv')
```

```
df.head()
```

II. BAGIAN PRAKTIKUM 2 – EKSPLORASI DATA SEDERHANA

1) Informasi dataset

```
df.info()
```

```
df.describe()
```

2) Cek proporsi target

```
df['Respon_Promo'].value_counts(normalize=True)
```

3) Encode kolom kategori (TrenPenjualan)

```
encoder = LabelEncoder()
```

```
df['TrenPenjualan_encoded'] = encoder.fit_transform(df['TrenPenjualan'])
```

Mapping akan berbeda per dataset, contoh:

- Naik = 2
- Stabil = 1
- Turun = 0

III. BAGIAN PRAKTIKUM 3 - PEMISAHAN FITUR DAN TARGET

1) Tentukan fitur (X) dan target (y)

```
X = df[['Usia', 'Pendapatan', 'FrekuensiBelanja', 'TotalBelanja', 'TrenPenjualan_encoded']]
```

```
y = df['Respon_Promo']
```

2) Train-test split

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

3) Standardisasi fitur

```
scaler = StandardScaler()
```

```
X_train_scaled = scaler.fit_transform(X_train)
```

```
X_test_scaled = scaler.transform(X_test)
```

Catatan: Decision Tree tidak wajib scaling, tapi Logistic Regression dan Naive Bayes lebih baik dengan scaling.

IV. BAGIAN PRAKTIKUM 4 - MODEL 1: LOGISTIC REGRESSION

1) Latih model

```
log_reg = LogisticRegression()
```

```
log_reg.fit(X_train_scaled, y_train)
```

2) Prediksi

```
y_pred_log = log_reg.predict(X_test_scaled)
```

3) Evaluasi

```
print("Confusion Matrix:")
```

```
print(confusion_matrix(y_test, y_pred_log))
```

```
print("\nClassification Report:")
print(classification_report(y_test, y_pred_log))
```

V. BAGIAN PRAKTIKUM 5 – MODEL 2: DECISION TREE

1) Latih model

```
tree_clf = DecisionTreeClassifier(max_depth=4, random_state=42)
tree_clf.fit(X_train, y_train)
```

2) Prediksi & Evaluasi

```
y_pred_tree = tree_clf.predict(X_test)
```

```
print("Confusion Matrix:")
print(confusion_matrix(y_test, y_pred_tree))
```

```
print("\nClassification Report:")
print(classification_report(y_test, y_pred_tree))
```

VI. BAGIAN PRAKTIKUM 6 – MODEL 3: NAIVE BAYES

1) Latih model

```
nb_clf = GaussianNB()
nb_clf.fit(X_train_scaled, y_train)
```

2) Prediksi & Evaluasi

```
y_pred_nb = nb_clf.predict(X_test_scaled)
```

```
print("Confusion Matrix:")
print(confusion_matrix(y_test, y_pred_nb))
```

```
print("\nClassification Report:")
print(classification_report(y_test, y_pred_nb))
```

VII. BAGIAN PRAKTIKUM 7 – PERBANDINGAN MODEL & INTERPRETASI BISNIS

Model	Akurasi	Precision (kelas 1)	Recall (kelas 1)	F1-score
Logistic Regression
Decision Tree
Naive Bayes

Pertanyaan analisis:

1. Model mana paling baik dalam mendeteksi pelanggan yang *merespon promo* (lihat Recall kelas 1)?
2. Apakah Precision tinggi? Jika rendah, berarti banyak promo “terbuang” pada pelanggan yang tidak tertarik.
3. Bagaimana model ini membantu perusahaan menghemat biaya promosi?

Kaitan dengan Business Intelligence:

- Model dengan Recall tinggi membantu menangkap lebih banyak pelanggan potensial.
- Model dengan Precision tinggi mengurangi pemborosan biaya promosi.
- Model bisa dipakai untuk *targeted marketing* dan meningkatkan ROI promosi.

VIII. UJI PEMAHAMAN

1. Jelaskan perbedaan hasil Logistic Regression, Decision Tree, dan Naive Bayes berdasarkan laporan evaluasi yang Anda peroleh.
2. Dalam konteks bisnis, mana lebih berbahaya: False Positive atau False Negative dalam prediksi Respon_Promo? Jelaskan.
3. Jika perusahaan hanya mengirim promo ke pelanggan yang diprediksi “1”, bagaimana hasil model Anda mempengaruhi biaya marketing?
4. Berikan satu contoh keputusan bisnis lain yang dapat ditingkatkan dengan model klasifikasi.