TUGAS MANDIRI
Fundamentals of Data Mining

PREDIKSI LOYALITAS PELANGGAN BERDASARKAN DATA PERILAKU BELANJA

Nama : Muhammad Azhar
NPM : 231510076
Dosen : Erlin Elisa,S.Kom., M.Kom

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM

2026

KATA PENGANTAR

Penulis berterima kasih kepada Tuhan Yang Maha Esa atas kemampuan penulis menyelesaikan laporan Tugas Mandiri (TM) mata kuliah Fundamentals of Data Mining berjudul "Prediksi Loyalitas Pelanggan Berdasarkan Data Perilaku Belanja" tepat waktu.

Laporan ini disusun sebagai salah satu bentuk pemenuhan tugas mandiri dalam mata kuliah Fundamentals of Data Mining. Tujuan penyusunan laporan ini adalah untuk meningkatkan pemahaman dan penerapan konsep-konsep yang berkaitan dengan data mining, khususnya tentang proses pengolahan data, pemodelan klasifikasi, dan evaluasi model dalam memprediksi loyalitas pelanggan berdasarkan data perilaku belanja.

Penulis menyadari bahwa mereka menerima bantuan, bimbingan, dan dukungan dari berbagai pihak selama proses penyusunan laporan ini. Oleh karena itu, penulis mengucapkan terima kasih kepada guru mata kuliah Data Mining yang telah memberikan bimbingan dan pengetahuan selama proses pembelajaran. Penulis juga mengucapkan terima kasih kepada teman-teman yang telah membantu dan mendorong mereka untuk menyelesaikan laporan ini.

Penulis menyadari bahwa laporan ini memiliki kekurangan. Oleh karena itu, penulis berharap kritik dan saran yang bermanfaat untuk membantu mereka memperbaiki laporan ini di masa depan. Semoga laporan ini bermanfaat dan menambah pengetahuan, terutama tentang penerapan data mining di bidang analisis data pelanggan.

Batam, 5 January 2026

Muhammad Azhar

1. Deskripsi Dataset

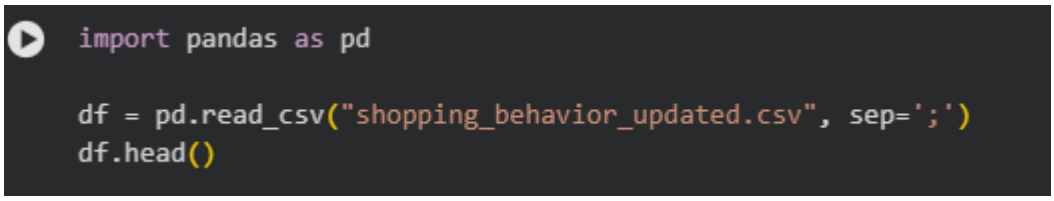
- Sumber Dataset :

  kaggle.com https://www.kaggle.com/datasets/sahilislam007/shopping-trends-and-customer-behaviour-dataset
- Jumlah Record : 3.900
- Jumlah Atribut : 16
- Tipe Data : CSV
- Tujuan penggunaan dataset :

Tugas mandiri ini menggunakan dataset perilaku belanja pelanggan yang terdiri dari 3900 data yang memiliki 16 atribut, termasuk informasi demografis pelanggan, karakteristik produk yang dibeli, dan kebiasaan transaksi pelanggan. Data ini menunjukkan perilaku pelanggan saat berbelanja, sehingga analisis lebih lanjut diperlukan.

Dataset ini berisi informasi seperti usia dan jenis kelamin pelanggan, item yang dibeli, kategori produk, jumlah, lokasi, ukuran dan warna produk, musim, rating ulasan, status langganan, diskon, jumlah pembelian sebelumnya, metode pembayaran, dan frekuensi. Kombinasi atribut ini memberikan gambaran lengkap tentang perilaku belanja pelanggan.

Loyalitas pelanggan diprediksi dengan data set ini, di mana status langganan, atau langganan, menunjukkan loyalitas pelanggan. Diharapkan, dengan menggunakan data perilaku belanja, dapat dibuat model yang dapat menempatkan pelanggan ke dalam kategori loyal atau tidak loyal.



1 to 5 of 5 entries Filter

index	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Discount Applied	Previous Purchases	Payment Method	Frequency of Purchases
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Yes	14	Venmo	Fortnightly
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Yes	2	Cash	Fortnightly
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Yes	23	Credit Card	Weekly
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Yes	49	PayPal	Weekly
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Yes	31	PayPal	Annually

Show 25 per page

Gambar 1 Dataset

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 16 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Customer ID             3900 non-null   int64  
 1   Age                     3900 non-null   int64  
 2   Gender                  3900 non-null   object 
 3   Item Purchased          3900 non-null   object 
 4   Category                3900 non-null   object 
 5   Purchase Amount (USD)   3900 non-null   int64  
 6   Location                3900 non-null   object 
 7   Size                    3900 non-null   object 
 8   Color                   3900 non-null   object 
 9   Season                  3900 non-null   object 
 10  Review Rating           3900 non-null   float64
 11  Subscription Status     3900 non-null   object 
 12  Discount Applied        3900 non-null   object 
 13  Previous Purchases      3900 non-null   int64  
 14  Payment Method          3900 non-null   object 
 15  Frequency of Purchases  3900 non-null   object 
dtypes: float64(1), int64(4), object(11)
memory usage: 487.6+ KB
```

Gambar 2 Struktur Data

2. Penentuan Target dan Fitur

- Penjelasan Target
- Alasan memilih target
- Penjelasan Fitur

Dalam penelitian ini, atribut Langganan Status digunakan sebagai target (label) untuk menunjukkan loyalitas pelanggan; pelanggan dengan status berlangganan dianggap sebagai pelanggan loyal, sedangkan pelanggan tanpa status dianggap sebagai pelanggan tidak loyal. Pemilihan fitur ini didasarkan pada asumsi bahwa pelanggan berlangganan memiliki

kecenderungan untuk membeli barang berulang. Usia, kategori produk, jumlah pembelian, metode pembayaran, jumlah pembelian sebelumnya, dan frekuensi pembelian digunakan sebagai fitur karena mencerminkan kebiasaan dan pola belanja konsumen. Karena identitas pelanggan hanya berfungsi sebagai identitas dan tidak mempengaruhi loyalitas pelanggan, artibut pelanggan ID tidak digunakan dalam proses pemodelan.

```
df['Subscription Status'].value_counts()
```

Subscription Status	count
No	2847
Yes	1053

dtype: int64

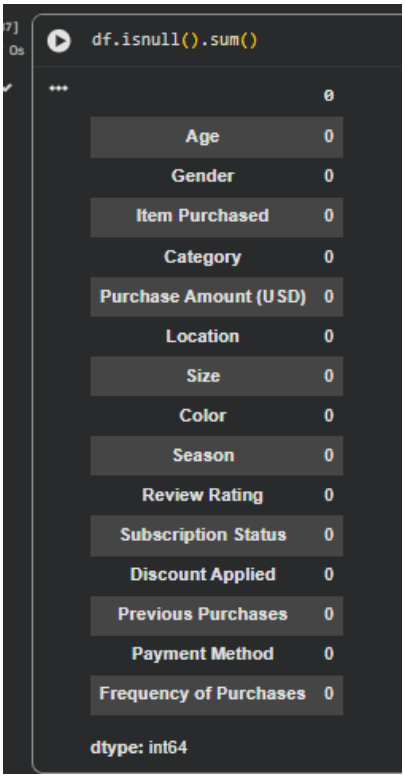
```
X = df.drop('Subscription Status', axis=1)
y = df['Subscription Status']
```

3. Preprocessing Data

Preprocessing, tahap penting dalam pengolahan data, bertujuan untuk memastikan bahwa data berada dalam kondisi yang siap untuk digunakan dalam proses pemodelan.

a) Pembersihan Data (Missing Value)

Fungsi df.isnull().sum() digunakan untuk mengecek nilai kosong atau nilai yang tidak ada. Hasil pengecekan menunjukkan bahwa seluruh atribut dataset memiliki nilai 0, yang menunjukkan bahwa dataset tidak memiliki data kosong. Oleh karena itu, dataset dapat digunakan langsung pada tahap berikutnya tanpa proses penanganan nilai yang hilang.



b) Penghapusan Kolom Tidak Relevan

Karena kolom Customer ID hanya berfungsi sebagai identitas unik pelanggan, dihapus dari dataset. Tidak adanya kolom ini akan memberikan informasi yang relevan untuk proses prediksi loyalitas pelanggan dan dapat menyebabkan model mempelajari pola yang tidak berguna.

```
df = df.drop('Customer ID', axis=1)
```

c) Encoding Data Kategorikal

Dataset memiliki jenis kelamin, kategori produk, lokasi, metode pembayaran, dan musim pembelian. Karena algoritma data mining tidak dapat memproses data dalam bentuk teks, proses pengkodean label digunakan untuk mengubah data kategorikal menjadi nilai numerik. Tujuan dari proses ini adalah untuk memastikan bahwa algoritma klasifikasi dapat memproses semua atribut.

```
X.dtypes
```

	0
Age	int64
Gender	object
Item Purchased	object
Category	object
Purchase Amount (USD)	int64
Location	object
Size	object
Color	object
Season	object
Review Rating	float64
Discount Applied	object
Previous Purchases	int64
Payment Method	object
Frequency of Purchases	object

dtype: object

Gambar 3 Sebelum Encoding

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

for col in X.columns:
    if X[col].dtype == 'object':
        X[col] = le.fit_transform(X[col])
```

```
X.dtypes
```

	0
Age	int64
Gender	int64
Item Purchased	int64
Category	int64
Purchase Amount (USD)	int64
Location	int64
Size	int64
Color	int64
Season	int64
Review Rating	float64
Discount Applied	int64
Previous Purchases	int64
Payment Method	int64
Frequency of Purchases	int64

dtype: object

Gambar 4 Sesudah Encoding

d) Normalisasi Data

Untuk menyamakan skala nilai setiap fitur ke dalam rentang 0 hingga 1, metode Min-Max Scaling digunakan untuk normalisasi data. Proses ini sangat penting untuk algoritma K-Nearest Neighbor (KNN) yang sensitif terhadap perbedaan skala data. Normalisasi memastikan bahwa setiap fitur memberikan kontribusi yang sama dalam proses perhitungan jarak.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)
```

```
X_scaled[:5]
```

```
array([[0.71153846, 1.        , 0.08333333, 0.33333333, 0.4125    ,
        0.32653061, 0.        , 0.29166667, 1.        , 0.24      ,
        1.        , 0.26530612, 1.        , 0.5       ],
       [0.01923077, 1.        , 0.95833333, 0.33333333, 0.55      ,
        0.36734694, 0.        , 0.5       , 1.        , 0.24      ,
        1.        , 0.02040816, 0.2       , 0.5       ],
       [0.61538462, 1.        , 0.45833333, 0.33333333, 0.6625    ,
        0.40816327, 0.66666667, 0.5       , 0.33333333, 0.24      ,
        1.        , 0.44897959, 0.4       , 1.        ],
       [0.05769231, 1.        , 0.58333333, 0.66666667, 0.875     ,
        0.7755102 , 0.33333333, 0.5       , 0.33333333, 0.4       ,
        1.        , 0.97959184, 0.8       , 1.        ],
       [0.51923077, 1.        , 0.08333333, 0.33333333, 0.3625    ,
        0.73469388, 0.33333333, 0.875     , 0.33333333, 0.08      ,
        1.        , 0.6122449 , 0.8       , 0.        ]])
```

Gambar 5 Output x_scaled [5]

e) Pembagian Data

Dengan menggunakan metode train_test_split, dataset dibagi menjadi dua bagian: data latih (80 persen) dan data uji (20 persen). Pembagian ini bertujuan untuk menguji kemampuan model untuk memprediksi data baru.

```
from sklearn.model_selection import train_test_split

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

```
print("X_train shape:", X_train.shape)
print("X_test shape:", X_test.shape)
print("y_train shape:", y_train.shape)
print("y_test shape:", y_test.shape)
```

```
X_train shape: (3120, 14)
X_test shape: (780, 14)
y_train shape: (3120,)
y_test shape: (780,)
```

4. Analisis Data dan Visualisasi

Sebelum proses pemodelan, analisis data dilakukan untuk mendapatkan pemahaman tentang karakteristik dan distribusi data. Menurut visualisasi distribusi loyalitas pelanggan,

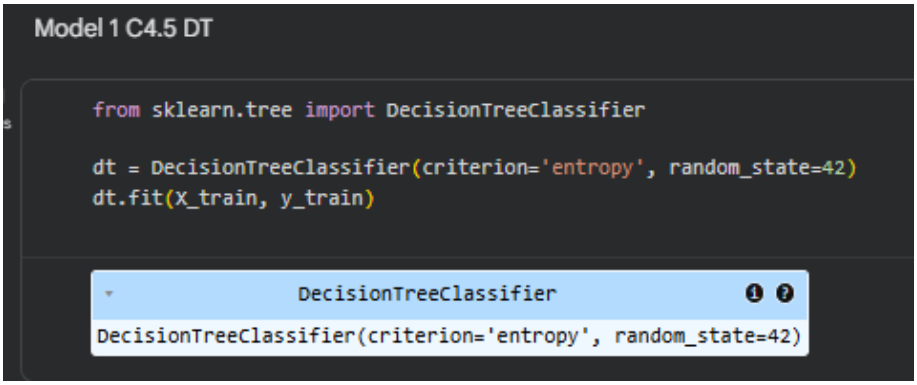
ada lebih banyak pelanggan yang tidak loyal daripada pelanggan yang loyal. Ini menunjukkan bahwa beberapa pelanggan mungkin tidak begitu loyal.

Selain itu, analisis korelasi antar fitur menunjukkan bahwa, meskipun tidak ada korelasi yang signifikan, beberapa fitur memiliki korelasi yang signifikan dengan loyalitas pelanggan. Hasil analisis ini menjadi dasar untuk memahami pola perilaku pelanggan sebelum melakukan pemodelan klasifikasi.

5. Implementasi Algoritma

a) C4.5 (Decision Tree)

Model klasifikasi berbasis pohon keputusan dengan kriteria entropy dibangun dengan algoritma C4.5. Algoritma ini memiliki kemampuan untuk membuat aturan keputusan yang mudah dipahami dan digunakan untuk menentukan loyalitas pelanggan berdasarkan karakteristiknya.



b) K-Nearest Neighbor (KNN)

Algoritma KNN digunakan sebagai metode klasifikasi berbasis kedekatan data, di mana kelas data ditentukan berdasarkan mayoritas kelas tetangga terdekatnya. Algoritma ini digunakan sebagai pembanding karena memiliki pendekatan yang berbeda dibandingkan dengan pohon keputusan.

```
MODEL 2: K-Nearest Neighbor (KNN)

from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train, y_train)

KNeighborsClassifier
KNeighborsClassifier()
```

c) Random Forest

Dengan menggabungkan beberapa pohon keputusan, algoritma Random Forest digunakan untuk meningkatkan kinerja prediksi. Metode ini dapat menghasilkan model prediksi loyalitas pelanggan yang lebih stabil dan mengurangi risiko overfitting.

```
MODEL 3: RANDOM FOREST

from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)

RandomForestClassifier
RandomForestClassifier(random_state=42)
```

6. Evaluasi Model

Untuk mengevaluasi model, metrik akurasi, laporan klasifikasi, dan matriks kekacauan digunakan. Hasilnya menunjukkan bahwa algoritma Random Forest menghasilkan nilai akurasi tertinggi sebesar 80,64%, diikuti oleh KNN sebesar 80,38%, dan C4.5 sebesar 78,85%.

Hasil menunjukkan bahwa, dibandingkan dengan dua algoritma lainnya, Random Forest memprediksi loyalitas pelanggan dengan lebih baik. Perbedaan akurasi yang kecil menunjukkan bahwa data memiliki tingkat kompleksitas yang cukup tinggi.

IMPORT METRIK EVALUASI

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

EVALUASI MODEL C4.5

```
y_pred_dt = dt.predict(X_test)

print("=== C4.5 (Decision Tree) ===")
print("Accuracy:", accuracy_score(y_test, y_pred_dt))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred_dt))
print("Classification Report:\n", classification_report(y_test, y_pred_dt))
```

```
=== C4.5 (Decision Tree) ===
Accuracy: 0.7884615384615384
Confusion Matrix:
 [[479  79]
 [ 86 136]]
Classification Report:
              precision    recall  f1-score   support

          No       0.85      0.86      0.85       558
         Yes       0.63      0.61      0.62       222

    accuracy                           0.79       780
   macro avg       0.74      0.74      0.74       780
weighted avg       0.79      0.79      0.79       780
```

Gambar 6 Evaluasi Model C4.5

EVALUASI MODEL KNN

```
y_pred_knn = knn.predict(X_test)

print("=== KNN ===")
print("Accuracy:", accuracy_score(y_test, y_pred_knn))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred_knn))
print("Classification Report:\n", classification_report(y_test, y_pred_knn))
```

```
=== KNN ===
Accuracy: 0.8038461538461539
Confusion Matrix:
 [[455 103]
 [ 50 172]]
Classification Report:
              precision    recall  f1-score   support

          No       0.90      0.82      0.86       558
         Yes       0.63      0.77      0.69       222

    accuracy                           0.80       780
   macro avg       0.76      0.80      0.77       780
weighted avg       0.82      0.80      0.81       780
```

Gambar 7 Evaluasi Model KNN

EVALUASI MODEL RANDOM FOREST

```
y_pred_rf = rf.predict(X_test)

print("=== Random Forest ===")
print("Accuracy:", accuracy_score(y_test, y_pred_rf))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred_rf))
print("Classification Report:\n", classification_report(y_test, y_pred_rf))
```

```
=== Random Forest ===
Accuracy: 0.8064102564102564
Confusion Matrix:
 [[444 114]
 [ 37 185]]
Classification Report:
              precision    recall  f1-score   support

          No       0.92      0.80      0.85       558
         Yes       0.62      0.83      0.71       222

    accuracy                           0.81       780
   macro avg       0.77      0.81      0.78       780
weighted avg       0.84      0.81      0.81       780
```

Gambar 8 Evaluasi Model Random Forest

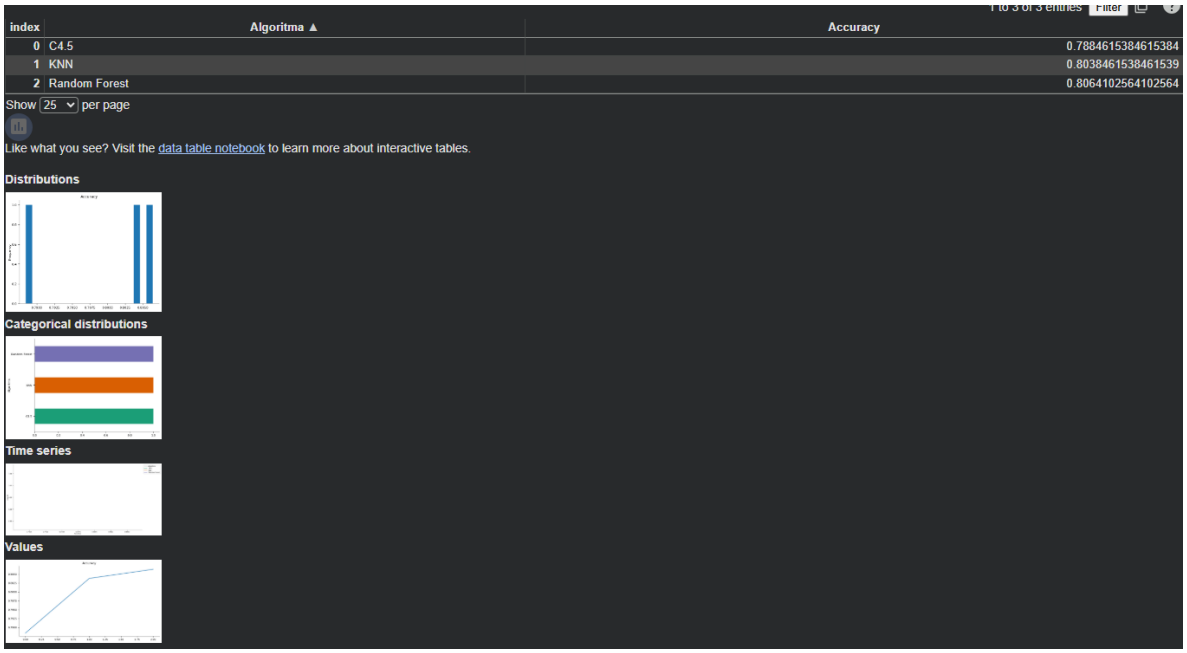
TABEL PERBANDINGAN AKURASI

```
import pandas as pd

accuracy_df = pd.DataFrame({
    'Algoritma': ['C4.5', 'KNN', 'Random Forest'],
    'Accuracy': [
        accuracy_score(y_test, y_pred_dt),
        accuracy_score(y_test, y_pred_knn),
        accuracy_score(y_test, y_pred_rf)
    ]
})

accuracy_df
```

Gambar 9 Tabel Perbandingan Akurasi



Hasil evaluasi menunjukkan bahwa algoritma Random Forest memiliki nilai akurasi tertinggi sebesar 80,64%, diikuti oleh KNN dan C4.5. Oleh karena itu, Random Forest dianggap sebagai algoritma terbaik.

7. Kesimpulan dan Saran

a) Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa memprediksi loyalitas pelanggan dengan data perilaku belanja dapat dilakukan dengan sukses. Algoritma Random Forest memiliki nilai akurasi tertinggi dan kinerja terbaik dari tiga algoritma klasifikasi yang digunakan.

b) Saran

Untuk penelitian lebih lanjut, disarankan untuk menambah jumlah data, mengubah parameter algoritma, dan mencoba teknik atau algoritma tambahan untuk meningkatkan kinerja model prediksi.

Lampiran

Link Repository Colab :
https://colab.research.google.com/drive/1lia6FhG9wMM6jcjSjWEgwNKapF-M5noY?usp=sharing