

Nama : Liza Cahyati Rami

Kelas : 005TPLE005

Nim : 231011400216

Matkul : Machine Learning

## Studi Kasus: Klasifikasi Dataset Iris Menggunakan Decision Tree

### Bagian 1 – Pemahaman Konsep (Teori)

#### 1. Pengertian Decision Tree

Decision Tree adalah algoritma machine learning yang memodelkan proses pengambilan keputusan dalam bentuk struktur pohon, di mana setiap node merepresentasikan pengujian terhadap suatu atribut, cabang merepresentasikan hasil pengujian, dan leaf node merepresentasikan hasil keputusan atau kelas.

#### 2. Konsep dalam Decision Tree

- Node : Titik pada pohon yang merepresentasikan fitur atau atribut yang digunakan untuk membagi data.
- Root : Node paling atas pada pohon yang menjadi awal proses pengambilan keputusan.
- Leaf : Node akhir yang tidak memiliki cabang lagi dan berisi hasil prediksi (kelas atau nilai).
- Splitting : Proses membagi data ke dalam beberapa subset berdasarkan nilai suatu fitur tertentu.
- Pruning : Teknik untuk memangkas cabang pohon yang tidak signifikan guna mengurangi overfitting dan meningkatkan generalisasi model.

#### 3. Perbedaan Decision Tree, Random Forest, dan Gradient Boosting

Aspek	Decision Tree	Random Forest	Gradient Boosting
Struktur	Satu pohon	Banyak pohon (ensemble)	Banyak pohon bertahap

Aspek	Decision Tree	Random Forest	Gradient Boosting
Overfitting	Tinggi	Rendah	Rendah
Akurasi	Sedang	Tinggi	Sangat tinggi
Interpretabilitas	Mudah	Sedang	Sulit
Waktu komputasi	Cepat	Sedang	Lebih lambat

#### 4. Kelebihan dan Kekurangan Tree-Based Methods

Kelebihan:

- Mudah dipahami dan diinterpretasikan
- Tidak memerlukan normalisasi data
- Dapat menangani data numerik dan kategorikal
- Cocok untuk feature importance

Kekurangan:

- Rentan terhadap overfitting
- Perubahan kecil pada data dapat mengubah struktur pohon
- Kurang optimal untuk dataset yang sangat besar (single tree)\

## Bagian 2 – Implementasi Model

### 1. Load dan Eksplorasi Dataset (EDA Singkat)

Dataset Iris berisi 150 data dengan 4 fitur:

- Sepal length
- Sepal width
- Petal length
- Petal width

Target:

- Iris-setosa

- Iris-versicolor
- Iris-virginica

## 2. Source Code (Python – scikit-learn)

# Import library

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
from sklearn.datasets import load_iris
```

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

```
from sklearn.tree import DecisionTreeClassifier, plot_tree
```

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

## 3. Preprocessing Data

- Dataset Iris tidak memiliki missing value
- Tidak memerlukan encoding karena seluruh fitur numerik

## 4. Split Data Training dan Testing

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

## 5. Membangun Model Decision tree

```
model = DecisionTreeClassifier(  
    criterion='gini',
```

```
max_depth=3,  
random_state=42  
)
```

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

## 6. Evaluasi Model

```
y_pred = model.predict(X_test)
```

```
accuracy = accuracy_score(y_test, y_pred)
```

```
print("Accuracy:", accuracy)
```

```
print("\nClassification Report:\n")
```

```
print(classification_report(y_test, y_pred,  
target_names=iris.target_names))
```

Contoh hasil evaluasi:

- Accuracy:  $\pm 0.97$
- Precision, Recall, dan F1-score menunjukkan performa yang sangat baik untuk seluruh kelas.

## 7. Visualisasi Decision Tree

```
plt.figure(figsize=(16,8))
```

```
plot_tree(  
    model,  
    feature_names=iris.feature_names,  
    class_names=iris.target_names,
```

```
filled=True  
)
```

```
plt.show()
```

### **Bagian 3 – Analisis dan Kesimpulan**

#### **1. Model Terbaik**

Model Decision Tree dengan parameter:

- criterion = gini
- max\_depth = 3

memberikan hasil akurasi tinggi tanpa overfitting.

#### **2. Faktor yang Mempengaruhi Performa Model**

- Kedalaman pohon (max\_depth)
- Pemilihan kriteria splitting (gini / entropy)
- Kualitas dan jumlah data
- Distribusi kelas target

#### **3. Kelebihan Tree-Based Methods pada Studi Kasus**

- Sangat mudah diinterpretasikan
- Visualisasi pohon membantu memahami pola data
- Cocok untuk dataset kecil hingga menengah seperti Iris

#### **4. Kesimpulan Akhir**

Decision Tree mampu mengklasifikasikan dataset Iris dengan sangat baik dan akurasi tinggi. Dengan pemilihan parameter yang tepat, model ini efektif, mudah dipahami, dan cocok digunakan sebagai baseline dalam klasifikasi data.