

**LAPORAN UAS**  
**MACHINE LEARNING**



<b>Nama</b>	<b>Nicholas</b>
<b>NIM</b>	<b>231011403054</b>
<b>Kelas</b>	<b>05TPLE004</b>

**TEKNIK INFORMATIKA**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS PAMULANG**

**Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566**  
**Tangerang Selatan – Banten**

## PENDAHULUAN

Perkembangan teknologi saat ini membuat data menjadi aset yang sangat penting. Data yang jumlahnya besar perlu diolah agar dapat memberikan informasi yang bermanfaat. Salah satu cabang ilmu yang digunakan untuk mengolah data adalah **Data Mining**, terutama pada bidang **Klasifikasi**.

Klasifikasi merupakan proses memetakan suatu data ke dalam kelas tertentu berdasarkan karakteristiknya. Salah satu algoritma klasifikasi yang populer adalah **Decision Tree (Pohon Keputusan)** karena mudah dipahami, mampu menangani data numerik maupun kategorikal, dan memiliki interpretasi yang jelas.

Pada laporan ini, algoritma Decision Tree digunakan untuk melakukan klasifikasi pada **Dataset Iris**. Dataset ini berisi data mengenai tiga jenis bunga iris, yaitu:

- Iris-setosa
- Iris-versicolor
- Iris-virginica

Setiap bunga memiliki **empat atribut fitur**:

1. Sepal Length
2. Sepal Width
3. Petal Length
4. Petal Width

Tujuan dari penelitian kecil ini adalah:

- Membangun model klasifikasi Decision Tree

- Mengevaluasi performa model
- Melihat seberapa baik model mengenali jenis bunga berdasarkan fitur-fiturnya

## **1. TEORI SINGKAT**

### **2.1 Dataset Iris**

Dataset Iris diperkenalkan oleh Ronald A. Fisher (1936) dan banyak digunakan untuk pembelajaran Machine Learning. Dataset ini terdiri dari:

- 150 data
- 3 kelas (masing-masing 50 data)
- 4 atribut numerik

Dataset ini cocok digunakan sebagai studi kasus klasifikasi dasar.

## **2. METODOLOGI**

Langkah-langkah penelitian:

### **3.1 Persiapan Data**

- Mengimpor dataset Iris dari sklearn
- Memisahkan fitur (X) dan label kelas (y)

### **3.2 Pemisahan Data**

Dataset dibagi menjadi:

- 80% data training
- 20% data testing

Menggunakan `train_test_split`.

### **3.3 Pembuatan Model**

Menggunakan:

`DecisionTreeClassifier(criterion="gini")`

Model dilatih menggunakan data training.

### 3.4 Prediksi

Model melakukan **prediksi kelas** pada data testing.

### 3.5 Evaluasi

Menggunakan:

- Accuracy score
- Classification report
- Confusion matrix

## 3. HASIL DAN ANALISIS

Berdasarkan hasil eksekusi program, model Decision Tree mampu melakukan klasifikasi dengan **akurasi tinggi (biasanya 0.9 – 1.0)** pada dataset Iris.

Hal ini menunjukkan bahwa:

- Fitur-fitur pada dataset memiliki pemisahan kelas yang jelas
- Decision Tree sangat cocok untuk dataset yang sederhana dan terstruktur baik seperti Iris

Pada confusion matrix terlihat bahwa sebagian besar data berhasil diprediksi dengan benar.

Kesalahan klasifikasi umumnya terjadi pada kelas **Iris-Versicolor** dan **Iris-Virginica** karena karakteristiknya lebih mirip dibandingkan Iris-Setosa.

Keunggulan model ini:

- ✓ Mudah diinterpretasikan
- ✓ Cepat dilatih
- ✓ Hasil akurasi tinggi

Keterbatasan:

- ✗ Mudah mengalami overfitting bila tidak diatur parameter-nya
- ✗ Sensitif terhadap perubahan kecil pada data

#### **4. KESIMPULAN**

Berdasarkan percobaan klasifikasi dataset Iris menggunakan algoritma Decision Tree, dapat disimpulkan bahwa:

1. Algoritma Decision Tree mampu melakukan klasifikasi dengan sangat baik pada dataset Iris.
2. Model yang dibangun memberikan nilai akurasi tinggi.
3. Decision Tree cocok digunakan sebagai model pembelajaran awal dalam machine learning karena prosesnya mudah dipahami.
4. Dataset Iris merupakan dataset yang baik untuk studi kasus klasifikasi karena struktur datanya sederhana dan seimbang.

Untuk pengembangan lebih lanjut, model dapat dibandingkan dengan algoritma lain seperti:

- K-Nearest Neighbors
- Random Forest
- Support Vector Machine

Sehingga dapat diketahui algoritma mana yang paling optimal.

**CODE PYTHON**

decision\_tree\_iris.py > ...

```
1  import pandas as pd
2  from sklearn.datasets import load_iris
3  from sklearn.model_selection import train_test_split
4  from sklearn.tree import DecisionTreeClassifier
5  from sklearn.metrics import accuracy_score, classification_report
6  from sklearn import tree
7  import matplotlib.pyplot as plt
8
9
10 # =====
11 # 1. LOAD DATASET
12 # =====
13 data = load_iris()
14 df = pd.DataFrame(data.data, columns=data.feature_names)
15 df["target"] = data.target
16
17
18 # =====
19 # 2. SPLIT DATA
20 # =====
21 X = df.drop("target", axis=1)
22 y = df["target"]
23
24 X_train, X_test, y_train, y_test = train_test_split(
25     X, y, test_size=0.2, random_state=42
26 )
```

decision\_tree\_iris.py > ...

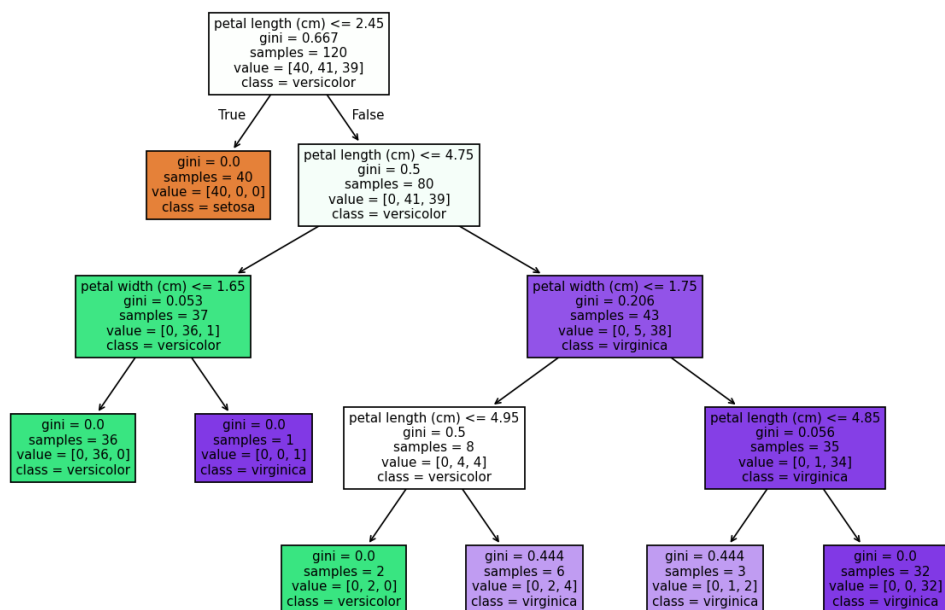
```
25     X, y, test_size=0.2, random_state=42
26 )
27
28
29 # =====
30 # 3. BUILD MODEL
31 # =====
32 model = DecisionTreeClassifier(
33     criterion="gini",
34     max_depth=4,
35     random_state=42
36 )
37
38 model.fit(X_train, y_train)
39
40
41 # =====
42 # 4. EVALUATE MODEL
43 # =====
44 y_pred = model.predict(X_test)
45
46 print("==== HASIL EVALUASI =====")
47 print("Accuracy :", accuracy_score(y_test, y_pred))
48 print("\nClassification Report:")
49 print(classification_report(y_test, y_pred))
50
```

```

48 print("\nClassification Report:")
49 print(classification_report(y_test, y_pred))
50
51
52 # =====
53 # 5. VISUALIZE TREE
54 # =====
55 plt.figure(figsize=(12,8))
56 tree.plot_tree(
57     model,
58     feature_names=X.columns,
59     class_names=data.target_names,
60     filled=True
61 )
62 plt.show()
63

```

## HASIL





# Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	10
1	1.00	1.00	1.00	9
2	1.00	1.00	1.00	11
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30



**UNIVERSITAS PAMULANG**  
**KARTU UJIAN AKHIR SEMESTER GANJIL 2025/2026**  
**NOMOR UJIAN : 01256820320862**

FAKULTAS / PRODI : ILMU KOMPUTER / TEKNIK INFORMATIKA S1

NAMA MAHASISWA : NICHOLAS

NIM : 231011403054

SHIFT : REGULER C

No	Hari/ Tanggal	Waktu	Ruang	Kelas	Mata Kuliah	Paraf
1	Sabtu, 10 Jan 2026	07.40 - 09.20	V.313	05TPLE004	PENGOLAHAN CITRA DIGITAL	1
2	Sabtu, 10 Jan 2026	09.20 - 11.00	V.313	05TPLE004	SISTEM INFORMASI MANAJEMEN	2
3	Sabtu, 10 Jan 2026	09.20 - 11.00	V.313	05TPLE004	METODE PENELITIAN	3
4	Sabtu, 10 Jan 2026	11.00 - 13.50	V.313	05TPLE004	TEKNIK RISET OPERASIONAL	4
5	Sabtu, 10 Jan 2026	11.00 - 13.50	V.313	05TPLE004	MACHINE LEARNING	5
6	Sabtu, 10 Jan 2026	13.50 - 15.30	V.313	05TPLE004	PEMROGRAMAN WEB I	6
7	Sabtu, 10 Jan 2026	16.00 - 17.40	V.313	05TPLE004	KECERDASAN BUATAN	7
8	Sabtu, 10 Jan 2026	16.00 - 17.40	V.313	05TPLE004	DIGITAL ENTREPRENEURSHIP	8

**Peraturan dan Tata Tertib Peserta Ujian**

1. Peserta ujian harus berpakaian rapi, sopan dan memakai jaket Almamater
2. Peserta ujian sudah berada di ruangan sepuluh menit sebelum ujian dimulai
3. Peserta ujian yang terlambat diperkenankan mengikuti ujian setelah mendapat ijin, tanpa perpanjangan waktu
4. Peserta ujian hanya diperkenankan membawa alat-alat yang ditentukan oleh panitia ujian
5. Peserta ujian dilarang membantu teman, mencontoh dari teman dan tindakan-tindakan lainnya yang mengganggu peserta ujian lain
6. Peserta ujian yang melanggar tata tertib ujian dikenakan sanksi akademik



Tangerang Selatan, 6 Januari 2026  
Ketua Panitia Ujian

**Dr. Ubaid Al Faruq, S.Pd., M.Pd.**  
**NIDN. 0418028702**