

Tugas 5: Tugas Praktikum Mandiri

Oryza Ayunda Putri - 0110224030

Teknik Informatika, STT Terpadu Nurul Fikri, Depok
E-mail: nasi.tektekmangudin@gmail.com

1. Tugas Praktikum Mandiri

1.1 Import Library

Library seperti pandas, numpy, matplotlib, seaborn, dan modul sklearn digunakan untuk mengelola data, membuat model, serta mengevaluasi hasil klasifikasi.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

Gambar 1 Import Library

1.2 Load Dataset

```
Loading Dataset

> ~
# Dataset Iris (contoh data bisa langsung dari sklearn atau CSV)
from sklearn.datasets import load_iris

iris = load_iris()
df = pd.DataFrame(
    data = np.c_[iris['data'], iris['target']],
    columns = iris['feature_names'] + ['species']
)
df.head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
0	5.1	3.5	1.4	0.2	0.0
1	4.9	3.0	1.4	0.2	0.0
2	4.7	3.2	1.3	0.2	0.0
3	4.6	3.1	1.5	0.2	0.0
4	5.0	3.6	1.4	0.2	0.0

Gambar 2 Load Dataset

Dataset Iris berisi **150 data bunga** dengan empat fitur:

- sepal length (cm)
- sepal width (cm)
- petal length (cm)
- petal width (cm) dan target species (0=setosa, 1=versicolor, 2=virginica).

1.3 Preprocessing Data

```
Cek Informasi Data

df.info()
df.isnull().sum()
df.duplicated().sum()
```

```
> ~
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column              Non-Null Count  Dtype  
---  --
0   sepal length (cm)    150 non-null   float64
1   sepal width (cm)     150 non-null   float64
2   petal length (cm)    150 non-null   float64
3   petal width (cm)     150 non-null   float64
4   species              150 non-null   float64
dtypes: float64(5)
memory usage: 6.0 KB
np.int64(1)
```

Gambar 3 Preprocessing Data

- df.info() menampilkan tipe data dan jumlah kolom.
- df.isnull().sum() memeriksa nilai kosong (missing value).

- `df.duplicated().sum()` mengecek data ganda (duplikat).
Hasil: Tidak ada missing value maupun data duplikat → dataset bersih.

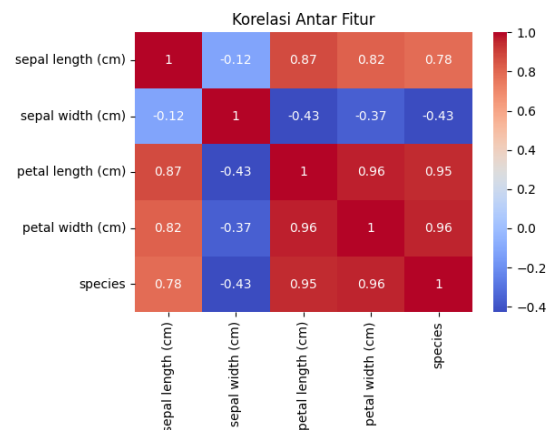
1.4 Analisis Korelasi Antar Fitur

```
iris = load_iris()
df = pd.DataFrame(
    data = np.c_[iris['data'], iris['target']],
    columns = iris['feature_names'] + ['species']
)

# 2. Hitung korelasi antar fitur
plt.figure(figsize=(6,4))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Korelasi Antar Fitur')
plt.show()
```

Gambar 4 Analisis Korelasi Antar Fitur

- Korelasi digunakan untuk melihat hubungan antar fitur numerik.
- `sns.heatmap()` menampilkan peta warna hubungan antar variabel.
- Warna **merah**= hubungan positif kuat,
Warna **biru**= hubungan negatif kuat.



Gambar 5 Hasil

- petal length dan petal width memiliki korelasi sangat tinggi (**0.96**) → saling berhubungan kuat.
- sepal width memiliki korelasi negatif terhadap fitur lain.
- Kesimpulan: fitur *petal* lebih berperan penting dalam membedakan spesies bunga.

1.5 Split Data (Training dan Testing)

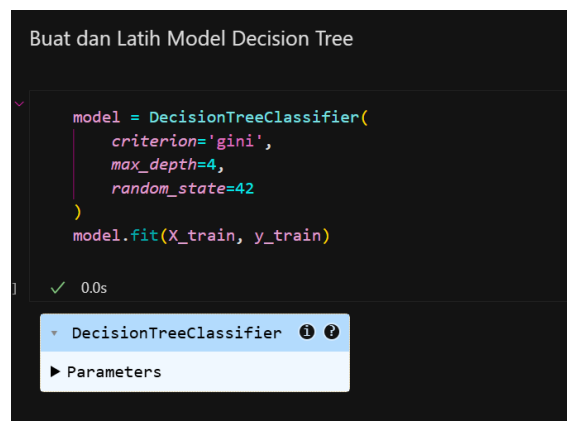
```
Pisahkan Fitur dan Target

X = df.iloc[:, :-1] # fitur
y = df.iloc[:, -1] # target

✓ 0.0s
```

Gambar 6 Split Data

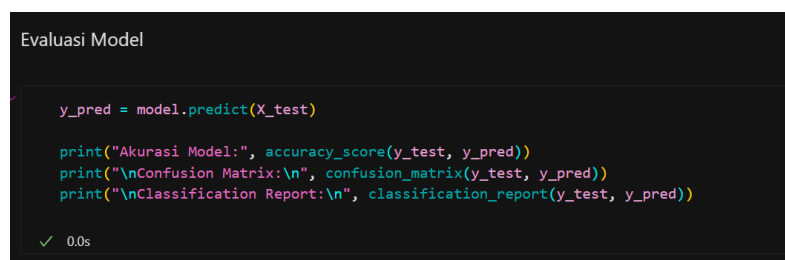
- X berisi empat kolom fitur.
- y berisi kolom target (species).



Gambar 7 Membuat Model Decision Tree

- Data dibagi:
 - 80% → *training set*
 - 20% → *testing set*
- random_state=42 menjaga hasil tetap konsisten.
- stratify=y menjaga proporsi kelas seimbang antara data latih dan data uji.

1.6 Evaluasi Model



Gambar 8 Evaluasi Model

Berdasarkan hasil evaluasi, model Decision Tree memiliki akurasi sebesar 93%. Hasil confusion matrix menunjukkan bahwa seluruh data kelas *Iris-setosa* berhasil diklasifikasikan dengan benar, sedangkan terdapat sedikit kesalahan pada kelas *versicolor* dan *virginica*. Hal ini disebabkan karena kedua spesies tersebut memiliki ukuran kelopak yang saling berdekatan, sehingga batas pemisahan pohon keputusan menjadi kurang tegas. Secara keseluruhan, model sudah bekerja dengan baik dan mampu mengenali pola pada dataset dengan akurasi tinggi.

```

Akurasi Model: 0.9333333333333333

Confusion Matrix:
[[10  0  0]
 [ 0  9  1]
 [ 0  1  9]]

Classification Report:

```

	precision	recall	f1-score	support
0.0	1.00	1.00	1.00	10
1.0	0.90	0.90	0.90	10
2.0	0.90	0.90	0.90	10
accuracy			0.93	30
macro avg	0.93	0.93	0.93	30
weighted avg	0.93	0.93	0.93	30

Gambar 9 Interpretasi Hasil Evaluasi Model Decision Tree

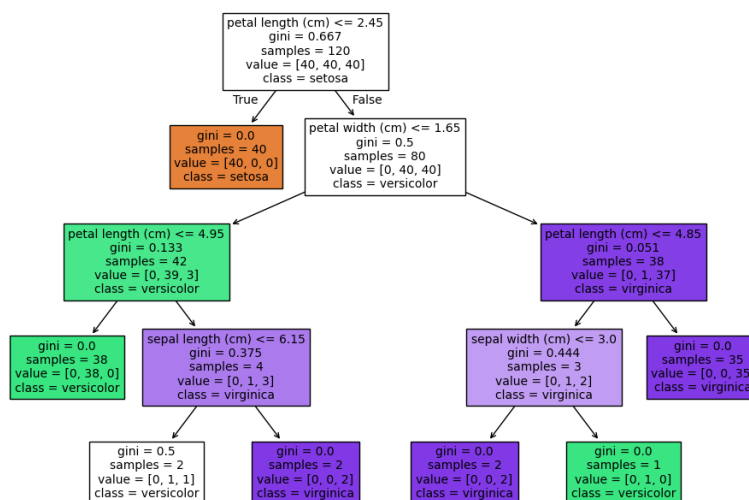
1.7 Visualisasi Pohon Keputusan

```
Visualisasi Decision Tree

plt.figure(figsize=(12,8))
plot_tree(
    model,
    feature_names=iris.feature_names,
    class_names=iris.target_names,
    filled=True,
    fontsize=10
)
plt.show()
```

Gambar 10 Visualisasi Pohon Keputusan

- Setiap node berisi fitur, nilai batas, impurity, dan jumlah sampel.
- Warna menunjukkan kelas dominan di node tersebut.
- Dari hasil pohon:
 - Fitur petal length (cm) menjadi pembeda utama antar spesies.
 - Model mengambil keputusan berlapis sampai mencapai label akhir (setosa, versicolor, virginica).



Gambar 11 Output Pohon Keputusan

1.8 Analisis Feature Importance

```
Analisis Fitur yang Paling Berpengaruh

fi = pd.DataFrame({
    'Feature': iris.feature_names,
    'Importance': model.feature_importances_
}).sort_values(by='Importance', ascending=False)
print(fi)
```

```
✓ 0.0s
```

	Feature	Importance
2	petal length (cm)	0.565639
3	petal width (cm)	0.411154
1	sepal width (cm)	0.016878
0	sepal length (cm)	0.006329

Gambar 12 Analisis Feature Importance

- Fitur petal length dan petal width paling berpengaruh.
- Kedua fitur ini sering digunakan model untuk menentukan kelas bunga.
- Sebaliknya, sepal length dan sepal width kurang berpengaruh terhadap hasil klasifikasi.

Kesimpulan:

1. Algoritma Decision Tree berhasil diterapkan untuk melakukan klasifikasi dataset Iris dengan tingkat akurasi sebesar 93%. Hal ini menunjukkan bahwa model mampu mengenali pola antar spesies bunga dengan sangat baik.
2. Dari hasil evaluasi model, semua data pada kelas *Iris-setosa* dapat diprediksi dengan benar, sedangkan terdapat sedikit kesalahan pada kelas *Iris-versicolor* dan *Iris-virginica*. Kesalahan ini terjadi karena kedua spesies tersebut memiliki kemiripan karakteristik pada panjang dan lebar petal (*petal length* dan *petal width*).
3. Berdasarkan analisis feature importance dan korelasi antar fitur, diketahui bahwa fitur petal length (cm) dan petal width (cm) merupakan atribut paling berpengaruh dalam menentukan klasifikasi spesies bunga Iris.
4. Dengan parameter `criterion='gini'` dan `max_depth=4`, model sudah cukup baik tanpa menimbulkan *overfitting*. Namun, peningkatan kedalaman pohon (*max_depth*) atau penggunaan metode lain seperti Random Forest dapat meningkatkan akurasi model.
5. Secara keseluruhan, praktikum ini menunjukkan bahwa algoritma Decision Tree mudah diimplementasikan, interpretasinya jelas melalui visualisasi pohon, dan efektif digunakan untuk kasus klasifikasi data sederhana seperti dataset Iris.