

Materi 9 Decision Tree  
Moch Alldho Candra Ramadhan  
1227030020

- **Jawaban Soal Nomor 1**

Kode Pemrograman:

```
from sklearn.tree import DecisionTreeClassifier
# Dataset (Fitur dan Target)
x = [[0 , 0, 0],
      [0 , 5, 0],
      [0 , 0, 5],
      [0 , 5, 5],
      [5 , 5, 0],
      [5 , 0, 5],
      [5 , 5, 5],
      [10, 5, 5],
      [5 , 10, 5],
      [10, 10, 10]]

y = [0, 0, 0, 5, 5, 5, 10, 10, 5, 0]

# Inisialisasi dan latih model
clf = DecisionTreeClassifier(random_state=1)
clf = clf.fit(x, y)

# Data untuk diprediksi
data_prediksi = [[10, 10, 5],
                  [5, 10, 2],
                  [2, 0, 10],
                  [5, 0, 2],
                  [0, 0, 2],
                  [2, 10, 2],
                  [1, 12, 5],
                  [2, 2, 6],
                  [10, 5, 7]]

# Prediksi hasil
print("Logika = Prediksi")
for data in data_prediksi:
    hasil = clf.predict([data])
    print(f"{data} = {hasil[0]}")
```

Hasil:

```
Logika = Prediksi
[10, 10, 5] = 10
[5, 10, 2] = 5
[2, 0, 10] = 0
[5, 0, 2] = 5
[0, 0, 2] = 0
[2, 10, 2] = 0
[1, 12, 5] = 5
[2, 2, 6] = 0
[10, 5, 7] = 10
```

- **Jawaban Soal Nomor 2**

Kode Pemrograman

```
from google.colab import drive
import pandas as pd
import numpy as np
from sklearn.tree import DecisionTreeRegressor
import matplotlib.pyplot as plt

# Mount Google Drive
drive.mount('/content/drive')

# Path ke file yang di-mount dari Google Drive
Filedb = '//content/Cosinus.txt' # Sesuaikan dengan path file di Google Drive Anda
Database = pd.read_csv(Filedb, sep=",", header=0)

# Lihat data
print("-----")
print(Database)

# x data (fitur) dan y target
x = Database[['Feature']] # Kolom 'Feature'
y = Database['Target']    # Kolom 'Target'

# Inisialisasi dan latih model Decision Tree Regressor
reg = DecisionTreeRegressor(random_state=1)
reg = reg.fit(x, y)

# Prediksi data (range 1 hingga 20)
xx = np.arange(1, 21, 1)
n = len(xx)
print("xx(i) Decision Tree")
```

```

for i in range(n):
    y_dct = reg.predict([[xx[i]]])
    print(f"{xx[i]:.2f} = {y_dct[0]:.5f}")

# Plot data prediksi dan data asli
y_dct2 = reg.predict(x)
plt.figure()
plt.plot(x, y_dct2, color='red', label="Decision Tree Prediction")
plt.scatter(x, y, color='blue', label="Actual Data")
plt.title('Prediksi Data Menggunakan Decision Tree')
plt.xlabel('Feature')
plt.ylabel('Target')
plt.legend(loc="upper left")
plt.show()

```

## Hasil:

Drive already mounted at /content/drive; to attempt to forcibly remount, call `drive.mount("/content/drive", force_remount=True)`.

```

-----
      Feature      Target
0          1  0.540302
1          2 -0.416147
2          3 -0.989992
3          4 -0.653644
4          5  0.283662
5          6  0.960170
6          7  0.753902
7          8 -0.145500
8          9 -0.911130
9         10 -0.839072
10         11  0.004426
11         12  0.843854
12         13  0.907447
13         14  0.136737
14         15 -0.759688
15         16 -0.957659
16         17 -0.275163
17         18  0.660317
18         19  0.988705
19         20  0.408082
xx(i) Decision Tree
1.00 = 0.54030
2.00 = -0.41615
3.00 = -0.98999
4.00 = -0.65364
5.00 = 0.28366
6.00 = 0.96017
7.00 = 0.75390
8.00 = -0.14550
9.00 = -0.91113

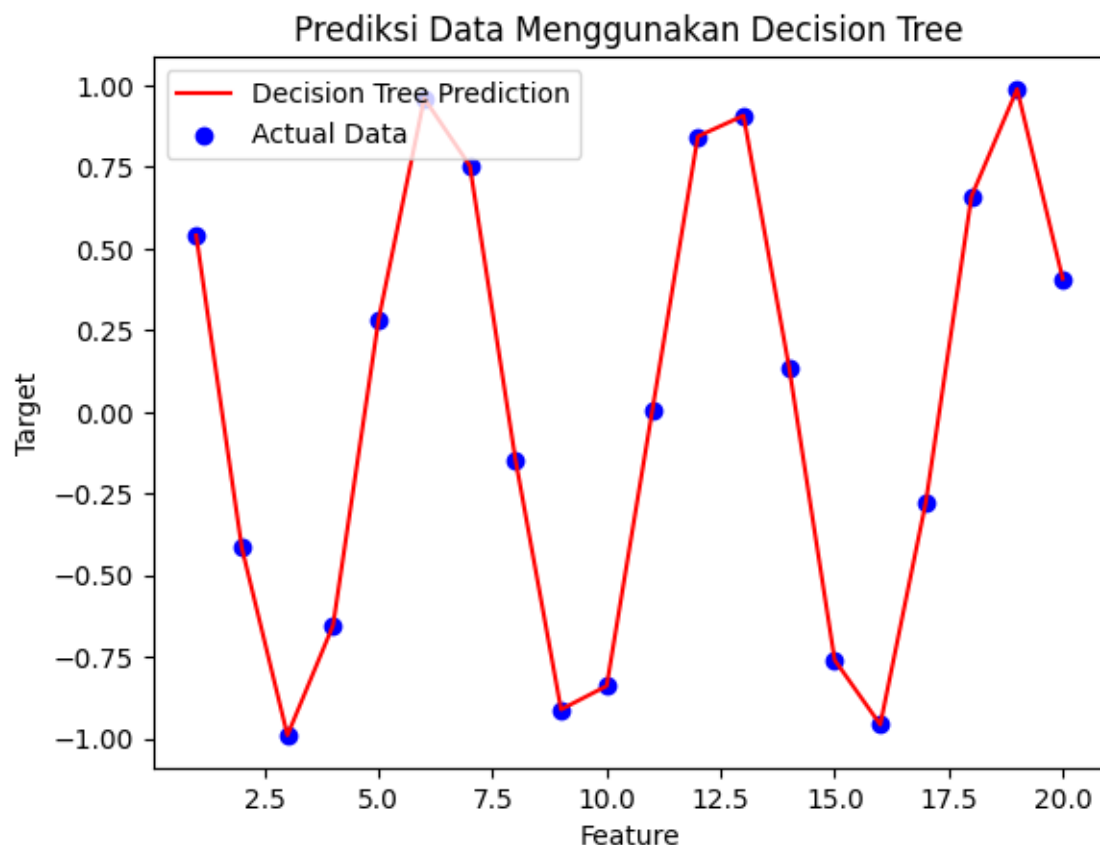
```

[illegible]

```

/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X
does not have valid feature names, but DecisionTreeRegressor was fitted with
feature names
  warnings.warn(

```



- **Jawaban Soal Nomor 3**

Hasil dari setiap metode

Metode Decision Tree yang dipakai di tugas ini digunakan buat dua jenis masalah, yaitu klasifikasi dan regresi. Di tugas pertama, metode ini bikin aturan klasifikasi dari dataset yang ada, di mana data berupa pasangan nilai fitur  $x$  dan target  $y$ . Dengan cara ini, Decision Tree bikin model yang bisa nebak nilai target untuk data baru berdasarkan pola yang ada di dataset awal. Hasil prediksinya nunjukin kalau modelnya bisa ngebedain atau ngelompokkan data ke dalam kategori tertentu. Sedangkan di tugas kedua, Decision Tree dipakai buat prediksi data cosinus. Dataset yang isinya fitur Feature dan target Target dipakai buat melatih model Decision Tree Regressor. Model ini berhasil ngikutin pola naik-turun fungsi cosinus, jadi bisa nebak nilai target untuk tiap fitur yang dikasih. Hasilnya nunjukin kalau Decision Tree bisa menangkap pola data dengan baik, meskipun terkadang ada risiko overfitting kalau modelnya terlalu rumit atau datanya sedikit.

- **Jawaban Soal Nomor 4**

Penggunaan metode Decision Tree di perkuliahan Fisika

1. **Klasifikasi Material Berdasarkan Sifat Fisikanya:**  
Menggunakan fitur seperti massa jenis, konduktivitas termal, atau resistivitas listrik untuk mengklasifikasikan jenis material (misalnya, logam, semikonduktor, atau isolator).
2. **Analisis Eksperimen Fisika:**  
Membantu mengidentifikasi hasil eksperimen berdasarkan kondisi awal, seperti suhu, tekanan, atau kecepatan awal dalam percobaan mekanika atau termodinamika.
3. **Prediksi Fenomena Fisika:**  
Menggunakan data simulasi untuk memprediksi fenomena, seperti prediksi distribusi medan listrik atau magnet di ruang tertentu berdasarkan konfigurasi muatan atau arus.

Lampiran Semua contoh dari modul runtime kode pemrograman data file sinus dan cosinus:

[https://colab.research.google.com/drive/1Te35gtGSmnJcBg7ogWFM\\_waytz0WqKVy#scrollTo=-w5PeOlcMTVG](https://colab.research.google.com/drive/1Te35gtGSmnJcBg7ogWFM_waytz0WqKVy#scrollTo=-w5PeOlcMTVG)



Cosinus.txt



Sinus.txt