

NIM : 2211522011 Tanggal : Rabu, 27 Maret 2024  
Nama : Ghifari Rizki Ramadhan Asisten : 1. Muhammad Irsyadul Fikri  
2. Athifa Rifda Andra  
3. Annisa Gita Subhi  
4. Syakina Triyana  
5. Husna Afiqah Yossyafra  
6. Vania Zerlina Utami  
7. Ghina Fitri Hidayah  
8. Sukma Anggarmadi  
9. Rafiqatul Ulya

Mata Kuliah : Praktikum Data Mining  
Modul : 04  
Kelas : A

## Resume dan Tugas “IF-ELSE dan Decision Tree”

### A. Materi

#### If-Else

Dalam bahasa pemrograman python, syntax atau statement yang digunakan untuk melakukan percabangan adalah:

- if
- elif
- dan else

Di mana if merupakan kondisi utama, sedangkan elif adalah kondisi kedua atau ketiga hingga ke-x, sedangkan else adalah kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Sebagaimana pohon, cabangnya tidak hanya 2, tapi bisa 3, 4, 5 bahkan lebih. Begitu juga pada logika kita. Kita bisa membuat lebih dari 2 cabang logika. Dan pada python, untuk membuat lebih dari 2 cabang, kita bisa menggunakan blok kode `if..elif..else`. Contoh yang paling umum digunakan untuk kasus percabangan `if..elif..else` adalah menentukan grade nilai suatu siswa.

Berikut ini rules yang akan kita gunakan:

1. **Predikat A** untuk **nilai  $\geq 90$**
2. **Predikat B** untuk **nilai  $\geq 80 < 90$**
3. **Predikat C** untuk **nilai  $\geq 60 < 80$**
4. **Predikat D** untuk **nilai  $\geq 40 < 60$**
5. Selain itu, maka **predikat E**.

Dari 5 rule di atas, kita akan menggunakan satu `if`, 3 `elif`, dan 1 `else`.

Perhatikan contoh berikut:

```
nilai = int(input('Masukkan nilai: '))

if nilai >= 90:
    print('Predikat A')
elif nilai >= 80:
    print('Predikat B')
elif nilai >= 60:
    print('Predikat C')
elif nilai >= 40:
    print('Predikat D')
else:
    print('Predikat E')
```

Jalankan program lalu memasukkan angka `80`, maka kita akan mendapatkan output predikat B:

```
Masukkan nilai: 80
Predikat B
```

## Decision Tree

Decision Tree adalah metode pengambilan kesimpulan berdasarkan pohon keputusan. Decision Tree termasuk salah satu sub bagian dalam Artificial Intelligence. Seperti namanya, Decision Tree menggunakan skema algoritma

pengambilan keputusan berantai dan tersusun menjadi hierarki keputusan. Jika digambarkan maka sekilas akan mirip dengan gambar sebuah pohon keputusan

Decision tree adalah salah satu algoritma populer yang digunakan untuk membangun model machine learning dalam bentuk struktur pohon. Algoritma ini termasuk ke dalam kategori supervised learning dan biasanya digunakan untuk masalah klasifikasi. Namun demikian, decision tree juga dapat digunakan untuk menangani masalah regresi.

Tujuan dari penggunaan decision tree adalah untuk membuat sebuah model yang dapat digunakan untuk memprediksi kelas atau nilai dari variabel target dengan mempelajari aturan pengambilan keputusan sederhana yang disimpulkan dari data sebelumnya.

## B. Tugas

1. Buatlah codingan dengan bahasa Python yang menghasilkan output nilai huruf mahasiswa berdasarkan IPK dengan kondisi sama seperti Portal Unand. Dengan ketentuan program: Nilai IPK diinputkan oleh user.

Nilai Angka (NA)	Nilai Mutu (NM)	Angka Mutu (AM)	Sebutan Mutu
$80 \leq NA \leq 100$	A	4,00	Sangat Cemerlang
$75 \leq NA < 80$	A <sup>-</sup>	3,75	Cemerlang
$70 \leq NA < 75$	B <sup>+</sup>	3,50	Sangat Baik
$65 \leq NA < 70$	B	3,00	Baik
$60 \leq NA < 65$	B <sup>-</sup>	2,75	Hampir Baik
$55 \leq NA < 60$	C <sup>+</sup>	2,50	Lebih dari cukup
$50 \leq NA < 55$	C	2,00	Cukup
$45 \leq NA < 50$	D	1,00	Kurang
$< 45$	E	0,00	Gagal

Source code :

```
ipk = float(input("masukkan nilai untuk ipk anda ="))

if (ipk>=80 and ipk<=100):
    nm = 'A'
    am = '4.00'
elif (ipk>=75 and ipk<80):
    nm = 'A-'
    am = '3.75'
elif (ipk>=70 and ipk<75):
    nm = 'B+'
    am = '3.50'
elif (ipk>=65 and ipk<70):
    nm = 'B'
    am = '3.00'
elif (ipk>=60 and ipk<65):
    nm = 'B-'
    am = '2.75'
elif (ipk>=55 and ipk<60):
    nm = 'C+'
    am = '2.50'
elif (ipk>=50 and ipk<55):
    nm = 'C'
    am = '2.00'
elif (ipk>=45 and ipk<50):
    nm = 'D'
    am = '1.00'
elif (ipk>0 and ipk<45):
    nm = 'E'
    am = '0.00'

print("nilai mutu anda = " + nm)
print("angka mutu anda = " + am)
```

Hasil :

```
masukkan nilai untuk ipk anda = 56
nilai mutu anda = C+
angka mutu anda = 2.50
```

2. Buatlah script python sederhana dalam memrediksi apakah kita akan pergi ke kampus atau tidak dengan menggunakan data set “hujan”, function if – else, library sklearn, dan decision tree!

Dengan situasi:

- Outlook: cerah, berawan, hujan
- Humidity: tinggi, normal

- Temperature: panas, ringan, dingin
- Windy: iya, tidak Dengan hasil output
- Pergi ke kampus? Ya/Tidak

```
[1] import pandas as pd

[2] data = pd.read_excel("hujan.xlsx")

[3] data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14 entries, 0 to 13
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  -
0   outlook     14 non-null    int64
1   temperature 14 non-null    int64
2   humidity    14 non-null    int64
3   windy       14 non-null    int64
4   play        14 non-null    int64
dtypes: int64(5)
memory usage: 688.0 bytes
```

```
[4] data.head()
```

	outlook	temperature	humidity	windy	play
0	0	0	0	0	0
1	0	0	0	1	0
2	1	0	0	0	1
3	2	1	0	0	1
4	2	2	1	0	1

```
[47] x = data[['outlook', 'temperature', 'humidity', 'windy']]
y = data['play']
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.1, random_state = 2)
```

X\_train

	outlook	temperature	humidity	windy
5	2	2	1	1
0	0	0	0	0
9	2	1	1	0
3	2	1	0	0
1	0	0	0	1
10	0	1	1	1
7	0	1	0	0
12	1	0	1	0
2	1	0	0	0
6	1	2	1	1
13	2	1	0	1
8	0	2	1	0

```
[49] from sklearn.tree import DecisionTreeClassifier
tree_model = DecisionTreeClassifier()
tree_model = tree_model.fit(X_train, y_train)
```

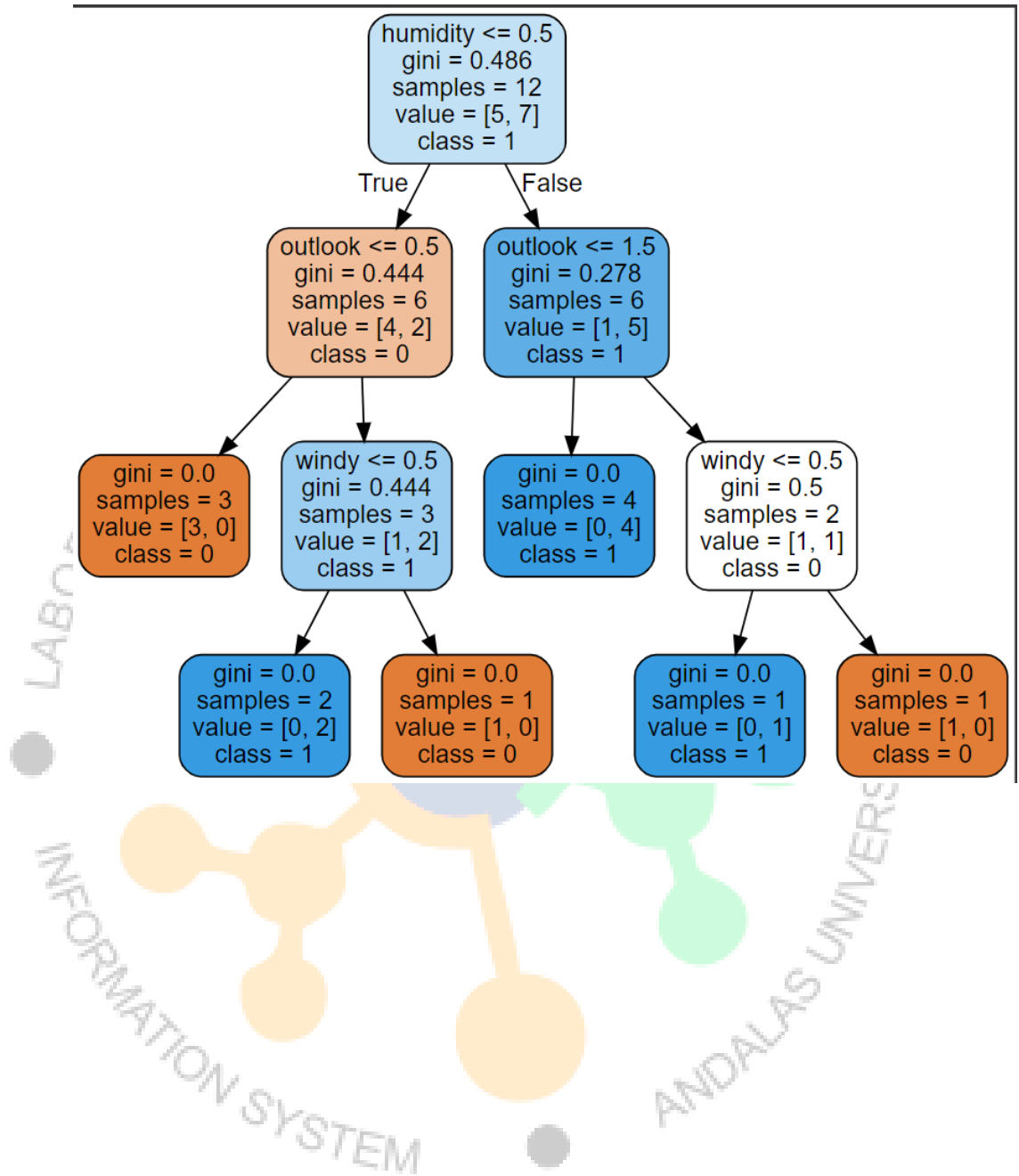
```
[50] from sklearn.metrics import accuracy_score
y_pred = tree_model.predict(X_test)
acc_score = accuracy_score(y_pred, y_test)
print('Accuracy : ', acc_score)
print(tree_model.predict([[2, 1, 1, 1]])[0])
```

Accuracy : 0.5  
0

/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning:

```
from sklearn.tree import export_graphviz
from graphviz import Source
export_graphviz(
tree_model, out_file = 'hujan_tree.dot', feature_names=['outlook', 'temperature', 'humidity', 'windy'], class_names=['0', '1'], rounded=True,
filled=True)
graph = Source.from_file('hujan_tree.dot')
graph
```

Hasil :



## Daftar Pustaka

“INTRODUCTION TO DECISION TREE METHOD WITH PYTHON (Step by Step).”

*Medium*, 26 November 2019,

<https://medium.com/@mmasdar/introduction-to-decision-tree-method-with-python-step-by-step-719c4b695d9c>. Accessed 31 March 2024.

“Python Dasar: Percabangan (If-Else dan Elif).” *Jago Ngoding*, 17 January 2021,

<https://jagongoding.com/python/dasar/percabangan/#apa-itu-percabangan>.

Accessed 31 March 2024.

