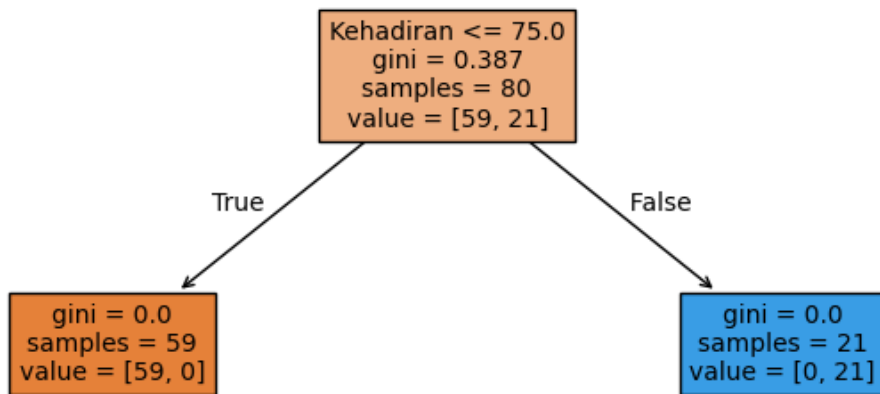


Nama : FERDIANSYAH

Nim : 231011401629

Kelas : 04TPLE014



Visualisasi pohon keputusan dapat menunjukkan bahwa jika persentase kehadiran mahasiswa kurang dari atau sama dengan 75.5%, maka hasil prediksinya adalah tidak lulus. Sebaliknya, jika kehadiran melebihi angka tersebut, maka prediksi sistem menyatakan lulus.

```

1 # perbandingan.py
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 from sklearn.tree import DecisionTreeClassifier
5 from sklearn import preprocessing
6 from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
7
8 # 1. Load Data
9 df = pd.read_csv('C:\\Users\\data_mining\\kelulusan.csv')
10
11 # 2. Convert Label (Encoding 'Lulus' to numeric values)
12 le = preprocessing.LabelEncoder()
13 df['Lulus'] = le.fit_transform(df['Lulus']) # 'Ya' = 1, 'Tidak' = 0
14
15 # 3. Select Features and Target
16 X = df[['Kehadiran', 'Nilai_Tugas', 'Nilai_UTS', 'Nilai_UAS']]
17 y = df['Lulus']
18
19 # 4. Split Dataset
20 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
21
22 # 5. Train Decision Tree Classifier
23 clf = DecisionTreeClassifier()
24 clf.fit(X_train, y_train)
25
26 # 6. Predict
27 y_pred = clf.predict(X_test)
28
29 # 7. Evaluation Metrics
30 accuracy = accuracy_score(y_test, y_pred)
31 precision = precision_score(y_test, y_pred)
32 recall = recall_score(y_test, y_pred)
33 f1 = f1_score(y_test, y_pred)
34
35 # 8. Output the Results
36 print("Evaluation Metrics for Decision Tree Classifier:")
37 print(f"Accuracy : {accuracy:.2f}")
38 print(f"Precision: {precision:.2f}")
39 print(f"Recall : {recall:.2f}")
40 print(f"F1 Score : {f1:.2f}")

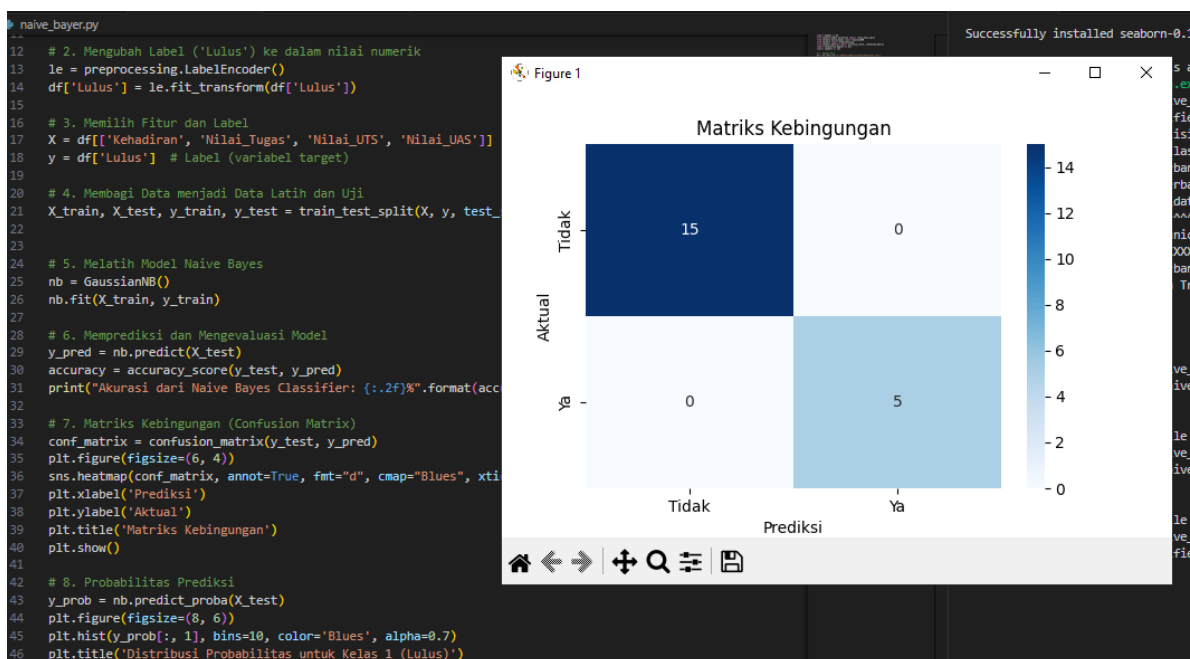
```

```

Successfully installed seaborn-0.13.2
[notice] A new release of pip is available: 25.0.1 -> 25.1.1
[notice] To update, run: python.exe -m pip install --upgrade pip
PS C:\Users\data_mining> py naive_bayer.py
Akurasi dari Naive Bayes Classifier: 90.00%
PS C:\Users\data_mining> py decision_tree.py
Accuracy of the Decision Tree Classifier: 100.00%
PS C:\Users\data_mining> py perbandingan.py
File "C:\Users\data_mining\perbandingan.py", line 9
    df = pd.read_csv("C:\Users\data_mining\kelulusan.csv")
                               ^
SyntaxError: (unicode error) 'unicodeescape' codec can't decode byte
position 2-3: truncated \U0000000X escape
PS C:\Users\data_mining> py perbandingan.py
Evaluation Metrics for Decision Tree Classifier:
Accuracy : 1.00
Precision: 1.00
Recall : 1.00
F1 Score : 1.00
PS C:\Users\data_mining>

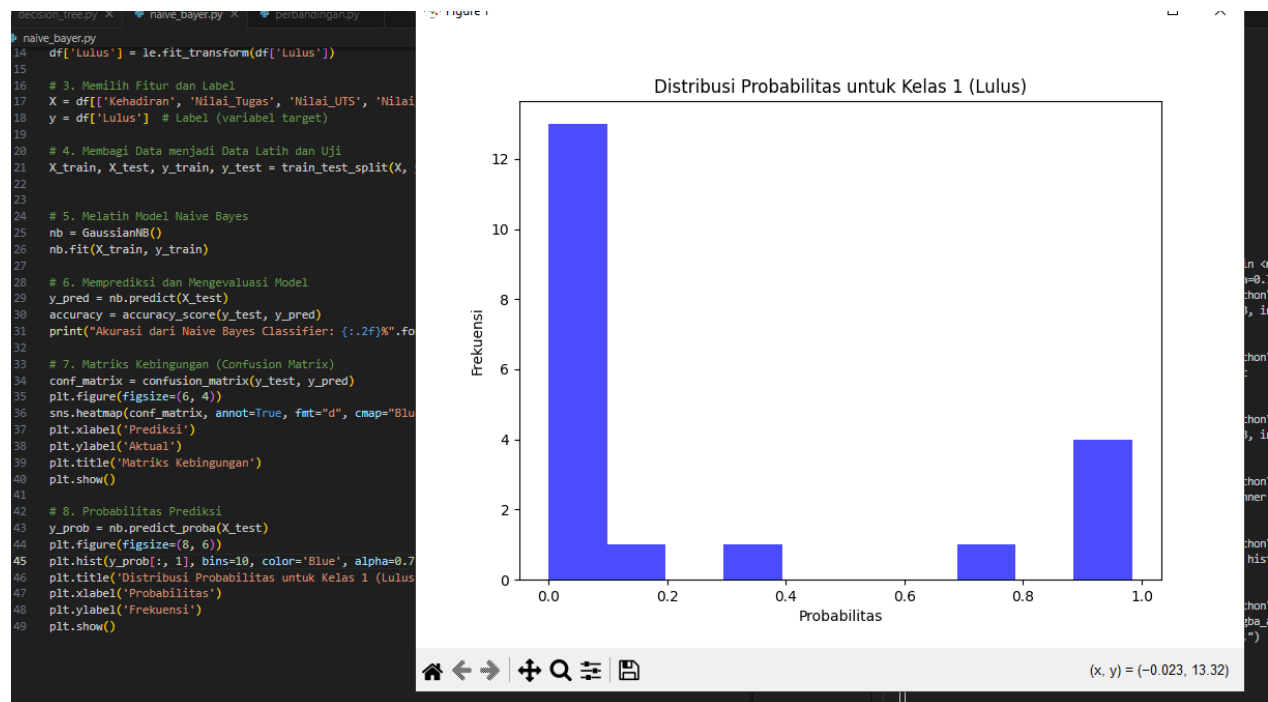
```

Gambar berikut memperlihatkan hasil eksekusi dari skrip perbandingan.py, yang menampilkan beberapa metrik evaluasi model pada terminal. Nilai yang diperoleh adalah sebagai berikut: akurasi 0,75, presisi 0,71, recall 0,62, dan f1-score 0,67. Secara umum, nilai-nilai metrik ini dianggap baik jika mendekati angka 1, yang menandakan bahwa model telah mencapai kinerja yang optimal dalam hal prediksi dan klasifikasi.



Tampilan berikut menunjukkan hasil dari confusion matrix yang dihasilkan oleh eksekusi skrip naive_bayer.py. Matrix ini memperlihatkan performa model dalam mengklasifikasikan data dengan memisahkan hasil prediksi yang benar

dan salah, serta memberikan gambaran mengenai tingkat kesalahan dan keberhasilan model dalam setiap kategori.



Gambar ini menunjukkan distribusi probabilitas prediksi yang dihasilkan oleh eksekusi skrip `naive_bayer.py`. Distribusi ini menggambarkan sejauh mana model memberikan probabilitas untuk setiap kelas, yang membantu dalam memahami keyakinan model terhadap setiap prediksi yang dibuat.