



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 1/9

NIM	242410101014
Nama	Adya Handika Putra AP
Kelas	B
Program Studi	Sistem Informasi
Asisten	Aulia Putri Maharani 232410101010 Fadhurrahman Aqil Supartha 232410101076

### LANGKAH KERJA

1. Ulangi langkah pada kelas praktikum untuk dataset Star dan Adult. Masing masing gunakan perbandingan 0.2, 0.3, 0.4. Tuliskan akurasinya

#### Data Star

```
import pandas as pd
df_stars = pd.read_csv('Stars.csv')
df_stars.head()
```

Unnamed: 0	Temperature (K)	Luminosity (L/L <sub>o</sub> )	Radius (R/R <sub>o</sub> )	Absolute magnitude (M <sub>v</sub> )	Star category	
0	0	3068	0.002400	0.1700	16.12	Brown Dwarf
1	1	3042	0.000500	0.1542	16.60	Brown Dwarf
2	2	2600	0.000300	0.1020	18.70	Brown Dwarf
3	3	2800	0.000200	0.1600	16.65	Brown Dwarf
4	4	1939	0.000138	0.1030	20.06	Brown Dwarf

```
# Menghapus kolom 'Unnamed: 0' yaitu index file csv
df_stars.drop(['Unnamed: 0'], axis='columns', inplace=True)
df_stars.head()
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 2/9

	Temperature (K)	Luminosity (L/L <sub>o</sub> )	Radius (R/R <sub>o</sub> )	Absolute magnitude (M <sub>v</sub> )	Star category
0	3068	0.002400	0.1700	16.12	Brown Dwarf
1	3042	0.000500	0.1542	16.60	Brown Dwarf
2	2600	0.000300	0.1020	18.70	Brown Dwarf
3	2800	0.000200	0.1600	16.65	Brown Dwarf
4	1939	0.000138	0.1030	20.06	Brown Dwarf

```
df_stars.isnull().sum()
```

	0
Temperature (K)	0
Luminosity (L/L <sub>o</sub> )	0
Radius (R/R <sub>o</sub> )	0
Absolute magnitude (M <sub>v</sub> )	0
Star category	0
dtype:	int64

```
inputs_stars = df_stars.drop('Star category', axis='columns')  
target_stars = df_stars['Star category']  
inputs_stars.head()
```

	Temperature (K)	Luminosity (L/L <sub>o</sub> )	Radius (R/R <sub>o</sub> )	Absolute magnitude (M <sub>v</sub> )
0	3068	0.002400	0.1700	16.12
1	3042	0.000500	0.1542	16.60
2	2600	0.000300	0.1020	18.70
3	2800	0.000200	0.1600	16.65
4	1939	0.000138	0.1030	20.06

```
from sklearn.model_selection import train_test_split  
from sklearn.naive_bayes import GaussianNB
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 3/9

```
# 0.2
X_train2, X_test2, y_train2, y_test2 = train_test_split(inputs_stars,
target_stars, test_size=0.2, random_state=42)
model2 = GaussianNB()
model2.fit(X_train2, y_train2)
print("Akurasi Stars (0.2):", model2.score(X_test2, y_test2))

# 0.3
X_train3, X_test3, y_train3, y_test3 = train_test_split(inputs_stars,
target_stars, test_size=0.3, random_state=42)
model3 = GaussianNB()
model3.fit(X_train3, y_train3)
print("Akurasi Stars (0.3):", model3.score(X_test3, y_test3))

# 0.4
X_train4, X_test4, y_train4, y_test4 = train_test_split(inputs_stars,
target_stars, test_size=0.4, random_state=42)
model4 = GaussianNB()
model4.fit(X_train4, y_train4)
print("Akurasi Stars (0.4):", model4.score(X_test4, y_test4))
```

Akurasi Stars (0.2): 0.8333333333333334

Akurasi Stars (0.3): 0.8611111111111112

Akurasi Stars (0.4): 0.84375

### Data Adult

```
import pandas as pd
df_adult = pd.read_csv('adult.csv')
df_adult.head()
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 4/9

	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States	<=50K
1	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States	<=50K
2	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States	>50K
3	44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	United-States	>50K
4	18	NaN	103497	Some-college	10	Never-married	NaN	Own-child	White	Female	0	0	30	United-States	<=50K

```
# menghapus kolom yang tidak relevan
```

```
df_adult.drop(['fnlwgt', 'educational-num', 'native-country'], axis='columns',  
inplace=True)  
df_adult.head()
```

	age	workclass	education	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	income
0	25	Private	11th	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	<=50K
1	38	Private	HS-grad	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	<=50K
2	28	Local-gov	Assoc-acdm	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	>50K
3	44	Private	Some-college	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	>50K
4	18	NaN	Some-college	Never-married	NaN	Own-child	White	Female	0	0	30	<=50K

```
dummies_gender = pd.get_dummies(df_adult.gender)  
df_adult = pd.concat([df_adult, dummies_gender], axis='columns')  
  
# Encoding kolom  
from sklearn.preprocessing import LabelEncoder  
le = LabelEncoder()  
df_adult['workclass'] = le.fit_transform(df_adult['workclass'].astype(str))  
df_adult['education'] = le.fit_transform(df_adult['education'].astype(str))  
df_adult['marital-status'] = le.fit_transform(df_adult['marital-status'].astype(str))  
df_adult['occupation'] = le.fit_transform(df_adult['occupation'].astype(str))  
df_adult['relationship'] = le.fit_transform(df_adult['relationship'].astype(str))  
df_adult['race'] = le.fit_transform(df_adult['race'].astype(str))  
  
# Hapus kolom asli yang telah di encode  
df_adult.drop(['gender', 'Male'], axis='columns', inplace=True)
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 5/9

```
df_adult.head()
```

	age	workclass	education	marital-status	occupation	relationship	race	capital-gain	capital-loss	hours-per-week	income	Female	Female
0	25	3	1	4	11	3	2	0	0	40	<=50K	False	False
1	38	3	3	2	9	0	4	0	0	50	<=50K	False	False
2	28	1	13	2	2	0	4	0	0	40	>50K	False	False
3	44	3	7	2	11	0	2	7688	0	40	>50K	False	False
4	18	8	7	4	6	3	4	0	0	30	<=50K	True	True

```
print(df_adult.isnull().sum())
```

```
inputs_adult = df_adult.drop('income', axis='columns')
```

```
target_adult = df_adult['income']
```

```
age          0
workclass    0
education    0
marital-status 0
occupation   0
relationship 0
race         0
capital-gain 0
capital-loss 0
hours-per-week 0
income       0
Female       0
Female       0
dtype: int64
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.naive_bayes import GaussianNB
```

```
# 0.2
```

```
X_train_a2, X_test_a2, y_train_a2, y_test_a2 = train_test_split(inputs_adult,
target_adult, test_size=0.2, random_state=42)
```

```
model_a2 = GaussianNB()
```

```
model_a2.fit(X_train_a2, y_train_a2)
```

```
print("Akurasi Adult (0.2):", model_a2.score(X_test_a2, y_test_a2))
```

```
# 0.3
```

```
X_train_a3, X_test_a3, y_train_a3, y_test_a3 = train_test_split(inputs_adult,
target_adult, test_size=0.3, random_state=42)
```

```
model_a3 = GaussianNB()
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 6/9

```
model_a3.fit(X_train_a3, y_train_a3)
print("Akurasi Adult (0.3):", model_a3.score(X_test_a3, y_test_a3))

# 0.4
X_train_a4, X_test_a4, y_train_a4, y_test_a4 = train_test_split(inputs_adult,
target_adult, test_size=0.4, random_state=42)
model_a4 = GaussianNB()
model_a4.fit(X_train_a4, y_train_a4)
print("Akurasi Adult (0.4):", model_a4.score(X_test_a4, y_test_a4))
```

Akurasi Adult (0.2): 0.80202681953117

Akurasi Adult (0.3): 0.8028390090766396

Akurasi Adult (0.4): 0.8010953575267441

2. Untuk masing masing dataset carilah subset kombinasi dari atribut yang memberi hasil klasifikasi lebih baik dari data asli

### Data Stars

```
# Radius dan Absolute Magnitude
inputs_sub_stars = inputs_stars[['Radius (R/Ro)', 'Absolute magnitude (Mv)']]
X_train_sub, X_test_sub, y_train_sub, y_test_sub =
train_test_split(inputs_sub_stars, target_stars, test_size=0.3,
random_state=42)

model_sub = GaussianNB()
model_sub.fit(X_train_sub, y_train_sub)
print("Subset Stars :", model_sub.score(X_test_sub, y_test_sub))
```

Subset Stars : 0.9861111111111112

### Data Adult

```
#Age, Capital Gain, Hours per Week
inputs_sub_adult = inputs_adult[['age', 'capital-gain', 'capital-loss',
'hours-per-week']]
```



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 7/9

```
X_train_sub_a, X_test_sub_a, y_train_sub_a, y_test_sub_a =  
train_test_split(inputs_sub_adult, target_adult, test_size=0.3,  
random_state=42)
```

```
model_sub_a = GaussianNB()  
model_sub_a.fit(X_train_sub_a, y_train_sub_a)  
print("Subset Adult :", model_sub_a.score(X_test_sub_a, y_test_sub_a))
```

Subset Adult : 0.7999727018358015

### HASIL DAN ANALISIS DATA

Metode Naive Bayes merupakan salah satu algoritma klasifikasi dalam machine learning. Algoritma ini bekerja dengan menghitung probabilitas suatu data termasuk ke dalam kelas tertentu berdasarkan atribut-atribut yang dimiliki. Metode Gaussian Naive Bayes, yaitu jenis Naive Bayes yang cocok untuk data numerik kontinu karena mengasumsikan bahwa data mengikuti distribusi normal atau Gaussian. Algoritma ini memiliki kelebihan seperti proses pelatihan yang cepat, mudah diimplementasikan, serta mampu bekerja dengan baik pada dataset berukuran besar.

1. Pada soal pertama dilakukan proses klasifikasi menggunakan algoritma Gaussian Naive Bayes pada dataset Stars dan Adult. Naive Bayes merupakan algoritma klasifikasi yang bekerja berdasarkan probabilitas dan Teorema Bayes. Algoritma ini cocok digunakan untuk klasifikasi data karena prosesnya cepat dan mampu menghasilkan prediksi yang cukup baik, terutama pada data numerik. Pada dataset Stars, kolom unnamed 0 dihapus karena hanya berfungsi sebagai indeks tambahan dan tidak berpengaruh pada proses klasifikasi. Setelah itu data dipisahkan menjadi input dan target, yaitu Star category. Model kemudian diuji menggunakan beberapa nilai test size, yaitu 0.2, 0.3, dan 0.4. Hasil pengujian menunjukkan bahwa Gaussian Naive Bayes mampu mengklasifikasikan kategori bintang dengan baik karena dataset berisi data numerik seperti temperature, luminosity, radius, dan absolute magnitude yang sesuai dengan metode Gaussian. Pada dataset Adult dilakukan preprocessing terlebih dahulu karena terdapat banyak data kategorikal. Beberapa kolom yang dianggap kurang relevan dihapus, kemudian data kategorikal diubah menjadi numerik menggunakan LabelEncoder dan get\_dummies.



## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 8/9

Setelah itu model dilatih menggunakan Gaussian Naive Bayes dengan variasi test size yang sama. Hasilnya menunjukkan bahwa model cukup baik dalam mengklasifikasikan income. Perbedaan nilai test size mempengaruhi jumlah data training dan testing, di mana semakin banyak data training biasanya membuat model belajar lebih optimal.

2. Pada soal kedua dilakukan pengujian menggunakan subset fitur untuk melihat pengaruh beberapa atribut tertentu terhadap hasil klasifikasi. Pada dataset Stars hanya digunakan fitur Radius dan Absolute magnitude. Sedangkan pada dataset Adult digunakan fitur age, capital gain, capital loss, dan hours-per-week. Hasil pengujian menunjukkan bahwa model Gaussian Naive Bayes masih mampu melakukan klasifikasi dengan cukup baik meskipun hanya menggunakan sebagian fitur. Namun, akurasi yang diperoleh umumnya lebih rendah dibandingkan ketika menggunakan seluruh fitur dataset. Hal ini karena informasi yang diterima model menjadi lebih sedikit. Meskipun begitu, subset fitur tersebut tetap memiliki pengaruh yang cukup besar terhadap proses klasifikasi, sehingga model masih dapat mengenali pola data dengan baik.





## Lembar Kerja Mahasiswa

Mata Kuliah : Data Mining  
Bahasan : Naïve Bayes Classification  
Halaman : 9/9

### KESIMPULAN

Penggunaan Algoritma Naive Bayes digunakan untuk klasifikasi data. Algoritma Gaussian Naive Bayes mampu digunakan untuk proses klasifikasi pada dataset Stars maupun Adult. Dataset Stars memberikan hasil yang baik karena sebagian besar atribut berupa data numerik yang sesuai dengan asumsi distribusi Gaussian. Pada dataset Adult diperlukan preprocessing tambahan seperti encoding data kategorikal agar dapat diproses oleh model. Perbedaan nilai test size mempengaruhi performa model karena menentukan jumlah data training dan testing yang digunakan. Selain itu, penggunaan subset fitur menunjukkan bahwa model masih dapat melakukan klasifikasi dengan cukup baik walaupun akurasi cenderung menurun dibandingkan penggunaan seluruh fitur dataset.

Link Google Colab

 LKM5\_Adya Handika Putra AP\_242410101014.ipynb

Link Youtube (Unlisted)

 LKM5\_Adya Handika Putra AP\_242410101014

Jember, 7 Mei 2026

Mengetahui,  
Dosen Datamining

Asisten,

Fajrin Nurman Arifin, S.T., M.Eng  
NIP. 198511282015041002

Fadhlurrahman Aqil Supartha  
NIM. 232410101076