

LINK KAGGLE KODINGAN PYTHON PENELITIAN:

<https://www.kaggle.com/code/annisarachmadhani/classificationusersatisfactiongoogleclassroom-nb>

ClassificationUserSatisfactionGoogleClassroom-... Draft saved

File Edit View Run Add-ons Help

+ Run All Code

```
[64]: # Import Modul / Library
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
```

```
[65]: # Loading Data (500)
data = pd.read_csv('/kaggle/input/gc-satisfaction-survey-datasets/Survei GC.csv')
data.head(501)
```

	Nama	Jurusan	Semester	Kualitas Sistem	Penggunaan	Responsivitas	Fleksibilitas	Informasi	Kepuasan	Klasifikasi Kepuasan
0	Berliani Afindasaputri	Keperawatan	3	Sangat Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
1	Ryek Diah mustika	Ppkn	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
2	Putri Angraini Aziz	Teknik Informatika	3	Setuju	Setuju	Setuju	Sangat Setuju	Setuju	Setuju	Puas
3	Harnelia	Teknik Informatika	3	Tidak Setuju	Tidak Setuju	Tidak Setuju	Setuju	Setuju	Tidak Setuju	Tidak Puas
4	Mursawal	Teknik Informatika	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
...
495	Ashabul	Teknik Informatika	3	Sangat Setuju	Sangat Setuju	Sangat Setuju	Sangat Setuju	Sangat Setuju	Sangat Setuju	Puas
496	Jumain Zulkaidah	Teknik Informatika	3	Setuju	Sangat Setuju	Setuju	Setuju	Setuju	Setuju	Puas
497	Novhira Sabrina	Pendidikan Bahasa Inggris	3	Setuju	Setuju	Setuju	Tidak Setuju	Sangat Setuju	Sangat Setuju	Puas
498	Dahlan Efendi	Teknik pertambangan	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
499	Reni	Ppkn	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas

500 rows x 10 columns

```
[66]: # Loading Data (10)
data = pd.read_csv('/kaggle/input/gc-satisfaction-survey-datasets/Survei GC.csv')
data.head(11)
```

	Nama	Jurusan	Semester	Kualitas Sistem	Penggunaan	Responsivitas	Fleksibilitas	Informasi	Kepuasan	Klasifikasi Kepuasan
0	Berliani Afindasaputri	Keperawatan	3	Sangat Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
1	Ryek Diah mustika	Ppkn	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
2	Putri Angraini Aziz	Teknik Informatika	3	Setuju	Setuju	Setuju	Sangat Setuju	Setuju	Setuju	Puas
3	Harnelia	Teknik Informatika	3	Tidak Setuju	Tidak Setuju	Tidak Setuju	Setuju	Setuju	Tidak Setuju	Tidak Puas
4	Mursawal	Teknik Informatika	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
5	Ni Luh Ica Ardini	Teknik Informatika	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
6	Annisa Novitria Hafsa Purba	Akuntansi	5	Sangat Setuju	Sangat Setuju	Sangat Setuju	Sangat Setuju	Setuju	Setuju	Puas
7	Nelti	Teknik Informatika	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
8	Nuzul Gusti	Teknik Informatika	3	Setuju	Setuju	Setuju	Setuju	Setuju	Setuju	Puas
9	Miftahul Jannah Ihsani Putri	Kesehatan Masyarakat	3	Setuju	Setuju	Tidak Setuju	Setuju	Setuju	Setuju	Puas
10	Lia Mirnawati	Sistem Informasi	1	Setuju	Sangat Setuju	Setuju	Sangat Setuju	Sangat Setuju	Sangat Setuju	Puas



```
# Check whether there is empty data or not
print(data.isnull().any())
print(data.isnull().sum())
```

```
Nama                False
Jurusan             False
Semester           False
Kualitas Sistem    False
Penggunaan         False
Responsivitas      False
Fleksibilitas      False
Informasi          False
Kepuasan           False
Klasifikasi Kepuasan False
dtype: bool
Nama                0
Jurusan             0
Semester           0
Kualitas Sistem    0
Penggunaan         0
Responsivitas      0
Fleksibilitas      0
Informasi          0
Kepuasan           0
Klasifikasi Kepuasan 0
dtype: int64
```

+ Code

+ Markdown

58]:

```
# Take 5 features
X = data[['Kualitas Sistem', 'Penggunaan', 'Responsivitas', 'Fleksibilitas', 'Informasi', 'Kepuasan']]

# perform coding
le = LabelEncoder()
X.loc[:, 'Kualitas Sistem'] = le.fit_transform(X['Kualitas Sistem'])
X.loc[:, 'Penggunaan'] = le.fit_transform(X['Penggunaan'])
X.loc[:, 'Responsivitas'] = le.fit_transform(X['Responsivitas'])
X.loc[:, 'Fleksibilitas'] = le.fit_transform(X['Fleksibilitas'])
X.loc[:, 'Informasi'] = le.fit_transform(X['Informasi'])
X.loc[:, 'Kepuasan'] = le.fit_transform(X['Kepuasan'])

y = le.fit_transform(data['Klasifikasi Kepuasan'])
```

[69]:

```
# Split data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=500)
```

[70]:

```
# Displays training table data
train_data = pd.DataFrame(data=X_train, columns=['Kualitas Sistem', 'Penggunaan', 'Responsivitas', 'Fleksibilitas', 'Informasi', 'Kepuasan'])
train_data['Label'] = y_train
print("Training Data:")
print(train_data)
```

```
Training Data:
   Kualitas Sistem  Penggunaan  Responsivitas  Fleksibilitas  Informasi  Kepuasan \
5                 2             2              2              2           2           2
210                0             0              0              0           0           0
44                 2             3              2              3           3           3
116                2             2              2              3           2           2
424                2             2              2              2           2           3
...              ...           ...           ...           ...           ...           ...
206                2             2              3              3           2           3
273                0             2              2              2           2           2
321                2             2              0              0           0           2
439                2             2              2              2           2           2
346                0             2              2              2           0           2

   Label
5       0
210     0
44      1
116     0
424     1
..     ...
206     1
273     0
321     0
439     0
346     0
```

[400 rows x 7 columns]

```
[71]: # Displays testing table data
test_data = pd.DataFrame(data=X_test, columns=['Kualitas Sistem', 'Penggunaan', 'Responsivitas', 'Fleksibilitas', 'Informasi', 'Kepuasan'])
test_data['Label'] = y_test
print("\nTesting Data:")
print(test_data)

import pandas as pd
```

```
Testing Data:
Kualitas Sistem  Penggunaan  Responsivitas  Fleksibilitas  Informasi  Kepuasan  \
57              2            2              2              2            2            2
70              0            2              2              0            1            2
364             2            3              3              3            2            2
108             2            2              3              3            2            3
318             0            0              0              0            0            0
..            ...            ...            ...            ...            ...            ...
215             0            0              0              0            0            0
132             1            2              3              0            0            3
290             2            2              3              3            2            2
443             3            3              2              2            2            2
124             2            0              2              0            2            0

Label
57      0
70      0
364     0
108     1
318     0
..     ...
215     0
132     1
290     0
443     0
124     0

[100 rows x 7 columns]
```

```
[72]: # Create a Naive Bayes model
naive_bayes = GaussianNB()
```

```
[73]: # Train a Naive Bayes model
naive_bayes.fit(X_train, y_train)
```

```
[73... GaussianNB
GaussianNB()
```

```
[74]: # Make predictions using the model on the test set
mapping = dict(zip(le.classes_, le.transform(le.classes_)))
print(mapping)
y_pred = naive_bayes.predict(X_test)

{'Puas': 0, 'Tidak Puas': 1}
```

```
[75]: # Model evaluation
print('Akurasi dari klasifikasi Naive Bayes pada set pengujian: {:.2f}'.format(accuracy_score(y_test, y_pred)))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

Akurasi dari klasifikasi Naive Bayes pada set pengujian: 0.95

```
Confusion Matrix:
[[ 88  0]
 [ 5  7]]
```

```
Classification Report:
              precision    recall  f1-score   support

     0       0.95         1.00         0.97         88
     1       1.00         0.58         0.74         12

 accuracy          0.97
 macro avg         0.79
 weighted avg      0.95
```