Naïve Bayesian

1. dataset ← milk.csv

+ Kode + Teks

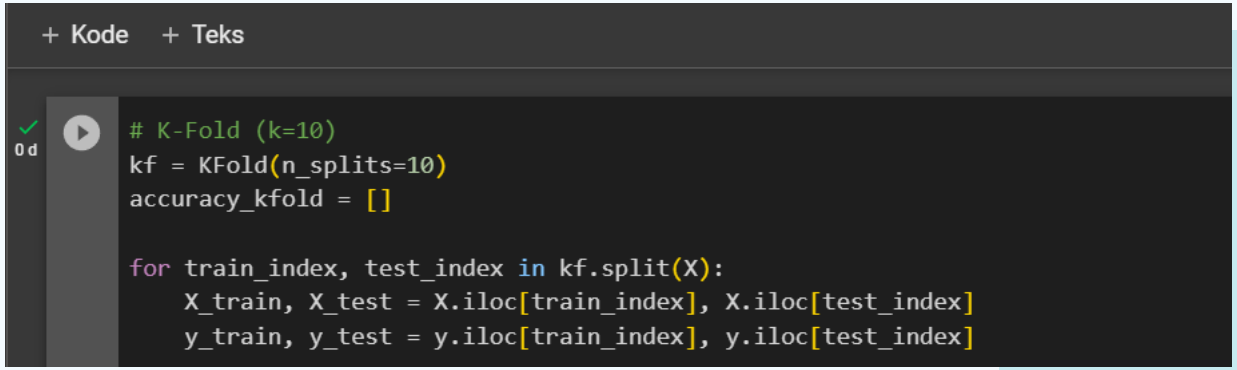
0d

```
import pandas as pd
from sklearn.model_selection import train_test_split, KFold, LeaveOneOut
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import accuracy_score

# Load dataset
dataset = pd.read_csv('milk.csv')

# Split atribut dan label
X = dataset.drop('Grade', axis=1)
y = dataset['Grade']
```

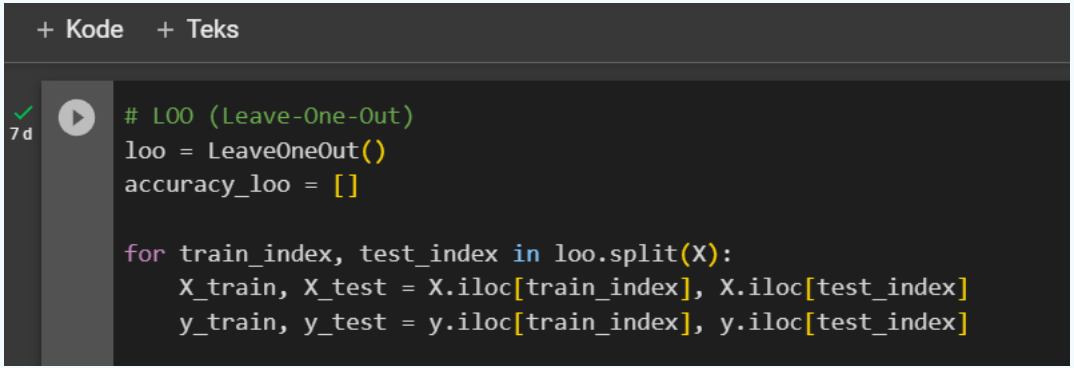
2a. Lakukan validation Model dengan metode Hold-out Method (70% - 30%)

```
[33] # Hold-out Method (70%-30%)
     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=100)
```

2b. Lakukan validation Model dengan metode K-Fold(k = 10)



2c. Lakukan validation Model dengan metode LOO



3. Lakukan klasifikasi naïve bayes untuk masing – masing pendekatan validasi dan hitunglah akurasi untuk masing – masing metode validation

```
[36] # Klasifikasi Naïve Bayes pada Hold-out Method
     nb = GaussianNB()
     nb.fit(X_train, y_train)
     y_pred_holdout = nb.predict(X_test)
     accuracy_holdout = accuracy_score(y_test, y_pred_holdout)
     print("2a. Akurasi Hold-out Method:", accuracy_holdout)
```

```
2a. Akurasi Hold-out Method: 0.9809523809523809
```

```
    nb.fit(X_train, y_train)
    y_pred_kfold = nb.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred_kfold)
    accuracy_kfold.append(accuracy)

mean_accuracy_kfold = sum(accuracy_kfold) / len(accuracy_kfold)
print("2b. Akurasi K-Fold (k=10):", mean_accuracy_kfold)
```

```
2b. Akurasi K-Fold (k=10): 0.926397124887691
```

3. Lakukan klasifikasi naïve bayes untuk masing – masing pendekatan validasi dan hitunglah akurasi untuk masing – masing metode validation

```
    nb.fit(X_train, y_train)
    y_pred_loo = nb.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred_loo)
    accuracy_loo.append(accuracy)

mean_accuracy_loo = sum(accuracy_loo) / len(accuracy_loo)
print("2c. Akurasi LOO:", mean_accuracy_loo)
```

```
2c. Akurasi LOO: 0.931067044381492
```

4. Lakukan normalisasi pada data training dan data test dengan metode minimax (menggunakan nilai min & max pada data training)

+ Kode + Teks

```
[41] # Normalisasi data dengan MinMaxScaler
     scaler = MinMaxScaler()
     X_train_normalized = scaler.fit_transform(X_train)
     X_test_normalized = scaler.transform(X_test)
```

```
# Klasifikasi Naïve Bayes pada data yang telah dinormalisasi
nb.fit(X_train_normalized, y_train)
y_pred_normalized = nb.predict(X_test_normalized)
accuracy_normalized = accuracy_score(y_test, y_pred_normalized)
print("4. Akurasi Naïve Bayes (dengan normalisasi):", accuracy_normalized)
```

```
4. Akurasi Naïve Bayes (dengan normalisasi): 1.0
```

5. Bandingkan nilai akurasi klasifikasi dengan naïve bayes pada salah satu metode validasi jika dat atraining & data test dilakukan normalisasi & tidak di normalisasi

```
[44] # Klasifikasi Naïve Bayes pada Hold-out Method tanpa normalisasi
     nb.fit(X_train, y_train)
     y_pred_holdout_nonnormalized = nb.predict(X_test)
     accuracy_holdout_nonnormalized = accuracy_score(y_test, y_pred_holdout_nonnormalized)
     print("5. Akurasi Hold-out Method (tanpa normalisasi):", accuracy_holdout_nonnormalized)
```

```
5. Akurasi Hold-out Method (tanpa normalisasi): 1.0
```

```
# Perbandingan akurasi Hold-out Method dengan dan tanpa normalisasi
print("Perbandingan akurasi Hold-out Method:")
print("- Dengan normalisasi:", accuracy_holdout)
print("- Tanpa normalisasi:", accuracy_holdout_nonnormalized)
```

```
Perbandingan akurasi Hold-out Method:
- Dengan normalisasi: 0.9809523809523809
- Tanpa normalisasi: 1.0
```