

# PRAKTIKUM 8

## Evaluasi Beberapa Algoritma Klasifikasi

### Tujuan :

Setelah mengikuti praktikum ini diharapkan praktikan dapat:

1. Memahami evaluasi beberapa metrik untuk mengukur kinerja klasifikasi
2. Mahasiswa dapat memahami confusion matrix dan peranannya untuk evaluasi klasifikasi
3. Memahami dan mampu menggunakan evaluasi K-fold

### Dasar Teori

#### Metode Validasi

Metode validasi digunakan untuk menguji sejauh mana akurasi klasifikasi dari metode klasifikasi yang dikembangkan. Metode ini dikenal dengan cross validation dimana data masukan dipecah menjadi data latih dan data uji. Pada metode ini terdapat beberapa metode, metode yang paling sederhana adalah adalah 2-fold cross validation dimana data dipisah secara acak menjadi dalam latih dan data uji. Pengembangan dari 2-fold cross validation adalah K-fold cross validation dimana data dipisah secara acak ke dalam K subset. Untuk pelatihan subset K-1 dipilih dan sisanya digunakan untuk pengujian. Proses ini diulangi sampai semua subset digunakan untuk pengujian. Versi lain dari K-fold cross validation adalah Leave-one-out cross validation (LOOCV) dimana K sama dengan jumlah sampel yakni masing-masing sampel digunakan untuk pengujian dan sisanya digunakan untuk pelatihan. Proses ini diulangi sampai semua sampel diuji.

#### Tolok Ukur Kinerja

Keluaran suatu klasifikasi dapat dikategorikan dalam 4 tipe sebagaimana dalam **confusion matrix**. jika keluaran klasifikasi konsisten dengan kebenaran, maka hal ini disebut True Positive (TP) atau True Negative (TN), tergantung pada nilai aktualnya. Sedangkan yang lain disebut False Positive (FP) atau False Negative (FN). Dalam domain masalah yang berbeda, FP dan TN depreciated dan diboboti secara berbeda. Sebagai contoh, FP lebih toleran daripada FN dalam diagnosis penyakit manusia.

	Hasil Prediksi	
Kelas sebenarnya	TP	FN
	FP	TN

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Sensitifitas = \frac{TP}{TP + FN}$$

$$Spesifitas = \frac{TN}{TN + FP}$$

## Tahapan Validasi Metode Klasifikasi

Beberapa metode klasifikasi yang telah dipelajari mempunyai kelebihan dan kekurangannya. Perbandingan kinerja antar metode klasifikasi ini diukur menggunakan beberapa metrik di atas. Tahapan untuk mengevaluasi suatu metode klasifikasi dilakukan sebagai berikut:

### 1. Tentukan Metode Klasifikasi yang Akan Dievaluasi

Pada bagian ini pilih metode klasifikasi yang akan dievaluasi. Simpan model klasifikasi dalam vector model. Hasil evaluasi ditampung pada vector result.

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
```

```
models = []
models.append(('KNN', KNeighborsClassifier()))
models.append(('NB', GaussianNB()))
models.append(('SVM', SVC(gamma='auto')))
```

```
# evaluate each model in turn
results = []
names = []
```

## 2. Baca Dataset

Ambil dataset yang akan dievaluasi. Untuk praktikum ini dievaluasi 3 dataset sebagai berikut :

- dataR2
- iris data
- breast cancer

Untuk praktek dataset dataR2 digunakan sebagai contoh sedangkan dua data berikutnya akan digunakan sebagai latihan.

```
! wget https://archive.ics.uci.edu/ml/machine-learning-
databases/00451/dataR2.csv
```

Alamat untuk dataset iris: <https://gist.githubusercontent.com/netj/8836201/raw/6f9306ad21398ea43cba4f7d537619d0e07d5ae3/iris.csv>

Alamat untuk dataset breast cancer: [https://raw.githubusercontent.com/frnkldgnwn/utpppm/main/breast\\_cancer.csv](https://raw.githubusercontent.com/frnkldgnwn/utpppm/main/breast_cancer.csv)

## 3. Tampilkan Dataset

```
import pandas as pd
import numpy as np
dataset = pd.read_csv('dataR2.csv')
dataset
```

### 3.1. Membuat Set Data Validasi

Tujuan evaluasi adalah memperkirakan keakuratan model terbaik pada data tak terlihat dengan mengevaluasinya pada data tak terlihat aktual. Evaluasi ini dilakukan dengan cara membagi dataset menjadi dua bagian yakni sebagian digunakan sebagai data latih dan sebagian yang lain digunakan sebagai data uji. Beberapa praktikum sebelumnya telah dipraktekkan dengan menggunakan metode **train\_test\_split**: **X\_train, X\_validation, y\_train, y\_validation = train\_test\_split(X, y, test\_size=0.20, random\_state=1, shuffle=True)**

Sebelum membuat data latih dan data uji. Langkah pertama pisahkan antara data atribut dan data label suatu dataset. Karena dataset data2R terdiri dari 9 atribut dan 1 label. Maka kita catat X sebagai data dan y sebagai label.

```
from sklearn.model_selection import train_test_split
array = dataset.values
X = array[:,0:9]
y = array[:,9]
X_train, X_validation, y_train, y_validation = train_test_split(X, y, test_size=0.20, random_state=1, shuffle=True)
```

Dataset akan terbagi menjadi data latih dan data validasi. Data latih akan digunakan untuk melatih model dengan mekanisme sebagian digunakan untuk testing dan sebagiannya lagi untuk training sebagaimana telah dipraktikkan pada praktikum sebelumnya. Pada praktikum ini pelatihan dilakukan menggunakan mekanisme K-fold cross validation. Namun demikian sebelum dilakukan K-fold cross validation pastikan data telah teracak sebelumnya.

## 3.2. K-Fold Cross Validation

K-fold Cross Validation adalah prosedur pengambilan sampel ulang yang digunakan untuk mengevaluasi model pembelajaran mesin pada sampel data terbatas. Prosedur ini memiliki parameter tunggal yang disebut K yang mengacu pada jumlah grup yang akan dipecah menjadi sampel data tertentu. Karena itu, prosedur ini sering disebut K-fold cross validation. Ketika nilai spesifik untuk K dipilih, nilai tersebut dapat digunakan sebagai pengganti K dalam referensi ke model, seperti K = 10 menjadi validasi silang 10 kali lipat.

Validasi silang terutama digunakan dalam pembelajaran mesin terapan untuk memperkirakan kemampuan model pembelajaran mesin pada data yang tidak terlihat. Yaitu, menggunakan sampel terbatas untuk memperkirakan bagaimana performa model secara umum saat digunakan untuk membuat prediksi pada data yang tidak digunakan selama pelatihan model.

Prosedur umumnya adalah sebagai berikut:

- Acak kumpulan data berdasarkan id
- Pisahkan kumpulan data menjadi kelompok K
- Untuk setiap grup unik:

- Ambil kelompok sebagai kelompok data uji
- Ambil kelompok yang tersisa sebagai kumpulan data pelatihan
- Catat hasil pengujian masing-masing grup

K-fold ini digunakan untuk validasi model yang digunakan. Evaluasi dilakukan pada dataset latih. Pada praktikum ini digunakan K=10, berarti akan membagi dataset latih menjadi 10 bagian. Satu bagian digunakan untuk testing dan sisanya sebagai data latih.

```
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import StratifiedKFold

for name, model in models:
    kfold = StratifiedKFold(n_splits=10, random_state=1, shuffle=True)
    cv_results = cross_val_score(model, X_train, y_train, cv=kfold, scoring=
'accuracy')
    results.append(cv_results)
    names.append(name)
    print('%s: %f (%f)' % (name, cv_results.mean(), cv_results.std()))
```

## 4. Buat Prediksi

Diantara metode klasifikasi yang telah dievaluasi, Kita pilih algoritma yang mempunyai kinerja paling baik yang akan digunakan untuk membuat metode prediksi. Prediksi dilakukan dengan mengevaluasi dataset evaluasi menggunakan metode klasifikasi yang telah terpilih (dalam hal ini metode terbaik adalah SVM). Dengan demikian metode SVM kita gunakan sebagai prediktor

```
model = SVC(gamma='auto')
model.fit(X_train, y_train)
predictions = model.predict(X_validation)
```

## 5. Evaluasi Prediksi

Kita dapat mengevaluasi prediksi dengan membandingkannya dengan hasil yang diharapkan dalam set validasi menggunakan confusion matrix dan laporan klasifikasi.

```
from sklearn.datasets import make_classification
X, y = make_classification(n_samples=1000, n_features=20, n_informative
=15, n_redundant=5, random_state=1)

from sklearn.metrics import classification_report
print(classification_report(y_validation, predictions))
```

Jalankan dan tampilkan hasilnya.

## Tugas

Template laporan: [https://colab.research.google.com/drive/17Na9aSs0f\\_w3JfETh-NV-3Mrz0Q96AXj?usp=sharing](https://colab.research.google.com/drive/17Na9aSs0f_w3JfETh-NV-3Mrz0Q96AXj?usp=sharing)

1. Lakukan perbandingan klasifikasi antara SVM, Naive Bayes, dan KNN untuk dataset iris dan breast cancer
2. Pada kedua dataset lakukan proses normalisasi data dan preproses data untuk menangani data yang hilang jika ada.