PROGRAM MAGISTER ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS GADJAH MADA

Ujian Akhir Semester
Periode Genap TA 2020/2021

Mata Kuliah	**: Data Science**
Hari, tanggal	**: Senin, 14 Juni 2021**
Waktu	**: 5 hari**
Dosen	**: Dr. Sigit Priyanta**
Sifat ujian	**: Dikerjakan di rumah, dikumpulkan sesuai jadwal.**

Instruksi:

1 Tuliskan jawaban singkat dan jelas pada file dokumen masing-masing.
2 Upload jawaban ke akun simaster masing-masing.
3 Kirim ke email : seagatejogja@ugm.ac.id, source code, dataset hasil preprocessing(jika dilakukan) dan file-file pendukung yang dihasilkan.

CO-3	Dapat menjelaskan konsep dan implementasi preprocessing data, cleaning, reduction, transformation, dan discretization	PLO3
CO-4	Dapat menjelaskan konsep eksplorasi data, deskripsi statistik data, dan visualisasi data	PLO3
CO-6	Dapat menjelaskan dan mengimplementasikan association rule dan sequential pattern mining	CO-6
CO-7	Dapat menjelaskan dan mengimplementasikan metode clustering	PLO3
CO-8	Dapat menjelaskan dan mengimplementasikan data mining dalam dokumen	PLO4

1 Terlampir pada soal ini dataset (data diagnosis *breast cancer*) untuk menjawab soal berikut: [CO-3 dan CO-4 : 40%]

Dataset berisi data diagnosa kanker payudara berdasarkan ciri-cirinya. Diagnosa terbagi menjadi dua kelas, yaitu kanker ganas atau kanker jinak. Ciri-ciri yang menjadi dasar klasifikasi tersebut adalah rata-rata radius kanker, rata-rata tekstur kanker, rata-rata perimeter, hingga cekungan terdalam dari area terkena kanker tersebut, hingga ada sekitar 30 ciri-ciri. Total datasetnya sejumlah 569, dengan pembagian 357 diagnosa kanker tidak ganas dan 212 diagnosa kanker ganas. Permasalahan yang timbul adalah dibutuhkan proses klasifikasi yang akurat untuk dapat mengklasifikasikan data tersebut berdasarkan ciri-ciri yang disediakan.

Untuk algoritma yang akan digunakan adalah random forest. Random forest (RF) merupakan metode yang cara kerjanya mirip *decision tree*. Prediksi didapatkan dengan merata-rata semua prediksi dari pohon regresi. RF pernah digunakan untuk mendeteksi COVID-19 berdasarkan data teks klinis dengan hasil presisi 93%, recall 94%, skor F1 93%, dan akurasi 94.3% (Khanday et al., 2020). Presisi merupakan nilai akurasi prediksi yang tepat. Recall merupakan nilai akurasi dari kelas yang sebenarnya. Skor F1 digunakan untuk mencari keseimbangan antara nilai presisi dan recall. Penelitian lain menyebutkan, RF sangat cocok disandingkan dengan algoritma pengambil fitur lain seperti Krill Herd Optimization (KHO) dengan akurasi mencapai 100% (Rani & Ramyachitra, 2018)

Langkah preprocessing yang dilakukan adalah dengan menghapus kolom yang berisi N/A semua dan menempatkan kolom diagnosa di urutan terakhir.

Langkah yang dilakukan untuk menyelesaikan masalah ini adalah sebagai berikut.

a Ekstrak dataset

```
import pandas as pd
data=pd.read_csv("breast-cancer.csv")
```

b Cari kolom yang memiliki nilai null atau N/A

```
data.isnull().sum()
data.isna().sum()
```

c Drop satu kolom yang hanya berisi nilai N/A.

```
data = data.dropna(axis='columns')
```

d Hitung jumlah diagnosa untuk mengetahui keseimbangan dataset.

```
data['diagnosis'].value_counts()
```

```
B    357
M    212
Name: diagnosis, dtype: int64
```

e Pindahkan kolom “diagnosis” ke urutan terakhir.

```
cols=data.columns.tolist()
data = data[['id',
 'radius_mean',
 'texture_mean',
 'perimeter_mean',
 'area_mean',
 'smoothness_mean',
 'compactness_mean',
 'concavity_mean',
 'concave points_mean',
 'symmetry_mean',
 'fractal_dimension_mean',
 'radius_se',
 'texture_se',
 'perimeter_se',
 'area_se',
```

```
 'smoothness_se',
 'compactness_se',
 'concavity_se',
 'concave points_se',
 'symmetry_se',
 'fractal_dimension_se',
 'radius_worst',
 'texture_worst',
 'perimeter_worst',
 'area_worst',
 'smoothness_worst',
 'compactness_worst',
 'concavity_worst',
 'concave points_worst',
 'symmetry_worst',
 'fractal_dimension_worst', 'diagnosis']]
```

f Ubah nilai diagnosis menjadi 0 dan 1

```
dataX = data.iloc[:, 1:31].values
dataY = data.iloc[:, 31].values
from sklearn.preprocessing import LabelEncoder
labelencoder_Y = LabelEncoder()
Y = labelencoder_Y.fit_transform(dataY)
```

Data X merupakan data semua fitur sedangkan data Y merupakan kelas dari data tersebut. Data Y diubah menjadi 0 dan 1 untuk memudahkan proses penghitungan menggunakan metode klasifikasi random forest (RF).

g Bagi dataset menjadi data train dan data tes dengan perbandingan 3:1

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(dataX, Y, test_size = 0.25,
random_state = 0)
```

Karena hampir semua algoritma Machine Learning menggunakan Euclidean distance, maka perlu dilakukan penyamaan magnitud. Hal ini dapat dilakukan menggunakan *feature scalling*. Semua nilai akan diubah berada pada skala tertentu, bisa 0-100 atau 0-1.

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

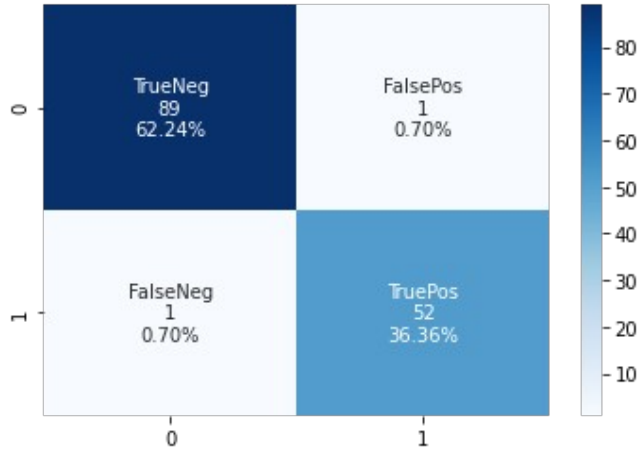
h Terapkan klasifikasi RF

```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators = 10, criterion = 'entropy',
random_state = 0)
classifier.fit(X_train, Y_train)
```

i Evaluasi menggunakan Confusion Matrix untuk dapat mengetahui akurasi, presisi, recall, dan nilai F1.

```
import numpy as np
import random
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(Y_test, y_pred)
group_names=['TrueNeg','FalsePos','FalseNeg', 'TruePos']
group_counts = ["{0:0.0f}".format(value) for value in
                cm.flatten()]
group_percentages = ["{0:.2%}".format(value) for value in
                     cm.flatten()/np.sum(cm)]
```

```
labels = [f"{v1}\n{v2}\n{v3}" for v1, v2, v3 in
          zip(group_names,group_counts,group_percentages)]
labels = np.asarray(labels).reshape(2,2)
sns.heatmap(cm, annot=labels, fmt='', cmap='Blues')
```



Klasifikasi menggunakan RF menghasilkan akurasi sebesar 98,6%, presisi sebesar 98,1%, recall juga sebesar 98,1%.

2 Terlampir pada soal ini dataset dalam format csv yang berisi tweet tentang jasa transportasi online yang sudah diberi label(0-negatif/1-positif) untuk menjawab soal berikut: [CO-3 dan CO-8 : 40%]
Buatlah model klasifikasi yang optimal untuk data tersebut untuk kepentingan opinion mining dari tweet tentang jasa transportasi online. Hal-hal yang perlu dilakukan antara lain adalah:

Data berisi cuitan dan sentimennya dengan topik gojek dan layanan turunannya. Data berisi 4000 cuitan dengan 3062 cuitan bersentimen negatif dan 938 lainnya bersentimen positif. Dari 4000 cuitan tersebut, kata “gojek” dan :gojekindonesia” mendapatkan lebih banyak sentimen negatif dengan disebut sebanyak 2197 kali. Uji coba kali ini mencoba untuk mengklasifikasikan menggunakan SVM untuk klasifikasi sentimen.

SVM dapat digunakan untuk mengklasifikasikan data teks. SVM mengungguli metode KNN dalam mengklasifikasikan spam atau ham email dengan akurasi mencapai 96.6% (Pratiwi & Ulama, 2016).

Langkah preprocessing yang dilakukan adalah menghapus stopword, melakukan stemming, dan menghapus regular expression.

SVM digunakan untuk mengklasifikasikan sentimen menghasilkan 99% akurasi di data training, 77.3% di data tes, 26,44% recall, dan presisi terbesar 53,22%. Hal tersebut disebabkan karena dataset tidak berimbang antara sentimen positif dan sentimen negatif.

Sumber kode : https://github.com/arivle/FinalExamDataScience

Khanday, A.M.U.D., Rabani, S.T., Khan, Q.R., Rouf, N. & Mohi Ud Din, M., 2020, Machine learning based approaches for detecting COVID-19 using clinical text data, *International Journal of Information Technology (Singapore)*, 12, 3, 731–739. *https://doi.org/10.1007/s41870-020-00495-9*,.

Pratiwi, S.N.D. & Ulama, B.S.S., 2016, Klasifikasi Email Spam dengan Menggunakan Metode Support Vector Machine dan k-Nearest Neighbor, *Jurnal Sains dan Seni ITS*, 5, 2, 344–349.

Rani, R.R. & Ramyachitra, D., 2018, Krill Herd Optimization algorithm for cancer feature selection and random forest technique for classification, *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, 2017-Novem, 109–113.