Nama Lengkap : Ni Putu Ayu Triana
NIM : 1908561031
Kelas : Aachen

Tugas CL-KNN (Kelas Aachen)

1. **Melakukan *Import Library***

   Library yang di-*import* merupakan *library* yang dibutuhkan untuk melakukan analisa, manipulasi, pengubahan dimensi, pemeriksaan data yang dapat dilakukan dengan format ekstensi yang mudah dibaca seperti ".csv", ".json". dan sebagainya. Selain itu, dibutuhkan juga *library* yang dapay melakukan operaasi fungsional, operasi matrix dan vector, sert6ta visualisasi data. Pada Latihan ini juga dibutuhkan algoritma untuk membangun model *machine learning*.

```
#import libraries
import pandas as pd
import itertools
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats

from sklearn import preprocessing
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics

%matplotlib inline
```

2. **Melakukan *Import Data***

   Melakukan *import data* berupa dataset yang di-*import* melalui link yang diberikan pada tugas. Kemudian, data yang telah di-*import* ditampilkan.

```
df = pd.read_csv('http://buku.dioskurn.com/buku1/ch9/churnprediction_ch9.csv', index_col=['customer_id'])
df
```

customer_id	product	reload_1	reload_2	video	music	games	chat_1	chat_2	socmed_1	socmed_2	internet	days_active	tenure	churn
285fae8412c4720a0c79d95f98434422	Kartu A	27734.30	24381.32	22000.0	33009.9	25669.97	1716.0	2145.0	0.0	792.0	11000.0	15	776	0
f45bce87ca6bf100f222fcc0db06b624	Kartu A	26433.00	26515.50	0.0	0.0	0.00	0.0	15444.0	0.0	0.0	74151.0	13	352	0
09b54557b1e2a10d998e3473a9ccd2a0	Kartu A	93212.17	67101.83	0.0	0.0	0.00	86795.5	94649.5	330.0	1485.0	27467.0	15	1987	0
11f252f48be36f93dd429f2ec86cb2f5	Kartu A	183.33	1087.17	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	2	285	0
a8df29ae9195eea348d2f74c967b978d	Kartu A	95296.67	76246.50	0.0	0.0	11000.00	118800.0	104940.0	0.0	0.0	63855.0	15	1081	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9e8b318d96caa9c0c4a50e8e59f5026c	Kartu B	1634.33	12085.33	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	1	490	1
a310627191bdbded3905ac73e77fe319	Kartu B	30000.33	45170.67	0.0	0.0	0.00	0.0	20001.0	0.0	0.0	0.0	1	3120	1
b6f11059e5c1df69b8c16d5c39af23dc	Kartu B	3333.33	13338.67	0.0	0.0	872.00	0.0	0.0	0.0	0.0	0.0	1	483	1
88709f1defd232243f729912be696f87	Kartu B	25000.00	33333.33	0.0	0.0	23497.33	0.0	0.0	0.0	0.0	0.0	15	786	1
2c5bc32bc9a9c393d393bfc11c409b0d	Kartu C	0.00	11084.00	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	1	222	1

10000 rows × 14 columns

Nama Lengkap : Ni Putu Ayu Triana
NIM : 1908561031
Kelas : Aachen

3. **Melakukan Pengubahan Data *String* Menjadi Numerik**
   Pengubahan data *string* dilakukan pada bagian kolom "product" agar data dapat dikalkulasikan.

```
# Pada product, nilai atribut Kartu A didefinisikan sebagai 0, Kartu B didefinisikan sebagai 1, dan Kartu C didefinisikan sebagai 2
df['product'] = df['product'].map({'Kartu A': 0,'Kartu B': 1, 'Kartu C':2})
df['product'].value_counts()
df
```

customer_id	product	reload_1	reload_2	video	music	games	chat_1	chat_2	socmed_1	socmed_2	internet	days_active	tenure	churn
285fae8412c4720a0c79d95f98434422	0	27734.30	24381.32	22000.0	33009.9	25669.97	1716.0	2145.0	0.0	792.0	11000.0	15	776	0
f45bce87ca6bf100f222fcc0db06b624	0	26433.00	26515.50	0.0	0.0	0.00	0.0	15444.0	0.0	0.0	74151.0	13	352	0
09b54557b1e2a10d998e3473a9ccd2a0	0	93212.17	67101.83	0.0	0.0	0.00	86795.5	94649.5	330.0	1485.0	27467.0	15	1987	0
11f252f48be36f93dd429f2ec86cb2f5	0	183.33	1087.17	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	2	285	0
a8df29ae9195eea348d2f74c967b978d	0	95296.67	76246.50	0.0	0.0	11000.00	118800.0	104940.0	0.0	0.0	63855.0	15	1081	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9e8b318d96caa9c0c4a50e8e59f5026c	1	1634.33	12085.33	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	1	490	1
a310627191bdbded3905ac73e77fe319	1	30000.33	45170.67	0.0	0.0	0.00	0.0	20001.0	0.0	0.0	0.0	1	3120	1
b6f11059e5c1df69b8c16d5c39af23dc	1	3333.33	13338.67	0.0	0.0	872.00	0.0	0.0	0.0	0.0	0.0	1	483	1
88709f1defd232243f729912be696f87	1	25000.00	33333.33	0.0	0.0	23497.33	0.0	0.0	0.0	0.0	0.0	15	786	1
2c5bc32bc9a9c393d393bfc11c409b0d	2	0.00	11084.00	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	1	222	1

10000 rows × 14 columns

4. **Menentukan Fitur yang Digunakan**
   Mencari fitur yang akan digunakan dan fitur yang akan ditemukan melalui fitur yang digunakan

```
df.keys()
```

```
Index(['product', 'reload_1', 'reload_2', 'video', 'music', 'games', 'chat_1',
       'chat_2', 'socmed_1', 'socmed_2', 'internet', 'days_active', 'tenure',
       'churn'],
      dtype='object')
```

```
X = df[['product', 'reload_1', 'reload_2', 'video', 'music', 'games', 'chat_1',
       'chat_2', 'socmed_1', 'socmed_2', 'internet', 'days_active', 'tenure',]]
```

```
Y = df['churn'].values
```

5. **Tahap Preprocessing**
   Melalukukan *preprocessing* data pada fitur yang digunakan dengan menggunakan *function* "StandardScaler".

```
X = preprocessing.StandardScaler().fit(X).transform(X.astype(int))
```

6. **Pembagian Data Training dan Testing**
   Melakukan pembagian data training dan data testing menggunakan *library* sklearn, dimana data training adalah sebanyak 80%, sementara itu data testing sebanyak 20%.

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=4)
print ('Trains set:', X_train.shape, Y_train.shape)
print ('Test set:', X_test.shape, Y_test.shape)
```

```
Trains set: (8000, 13) (8000,)
Test set: (2000, 13) (2000,)
```

Nama Lengkap : Ni Putu Ayu Triana
NIM : 1908561031
Kelas : Aachen

7. **Menentukan K yang menhasilkan nilai terbaik**

   Menentukan K yang menghasilkan nilai terbaik dapat dilakukan melakukan pengujian acak dari K=1 sampai K=10 seperti gambar dibawah

```
Ks = 10
mean_acc = np.zeros((Ks-1))
std_acc = np.zeros((Ks-1))
ConfustionMx = [];
for n in range(1,Ks):

    #Train Model and Predict
    neigh = KNeighborsClassifier(n_neighbors = n).fit(X_train,Y_train)
    yhat=neigh.predict(X_test)
    mean_acc[n-1] = metrics.accuracy_score(Y_test, yhat)


    std_acc[n-1]=np.std(yhat==Y_test)/np.sqrt(yhat.shape[0])

mean_acc
```

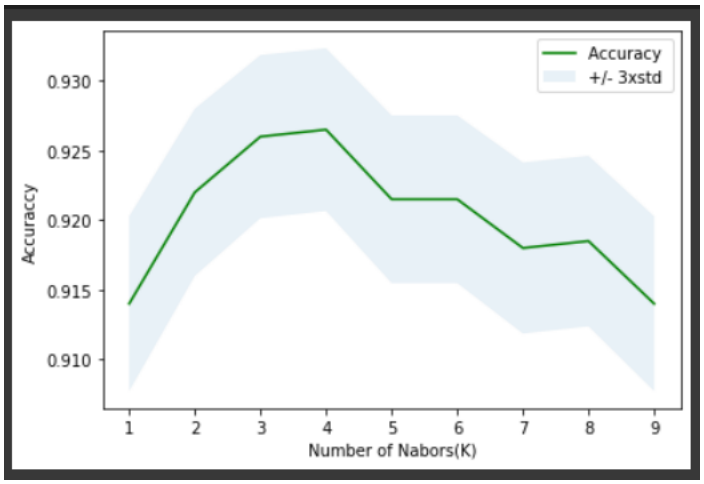
```
array([0.914 , 0.922 , 0.926 , 0.9265, 0.9215, 0.9215, 0.918 , 0.9185,
       0.914 ])
```

8. **Plotting Data**

   Membuat plot (grafik) dari data yang didapat

```
plt.plot(range(1, Ks), mean_acc, 'g')
plt.fill_between(range(1,Ks), mean_acc - 1* std_acc, mean_acc + 1* std_acc, alpha=0.10)
plt.legend(("Accuracy ", '+/- 3xstd'))
plt.ylabel("Accuraccy")
plt.xlabel("Number of Nabors(K)")
plt.tight_layout()
plt.show()
```

Berikut hasilnya:



Nama Lengkap : Ni Putu Ayu Triana
NIM : 1908561031
Kelas : Aachen

9. **Mendapat nilai K terbaik**
   Nilai K yang didapat adalah 4.

```
print("The best accuracy was with", mean_acc.max(), "with k=", mean_acc.argmax()+1)
```

```
The best accuracy was with 0.9265 with k= 4
```

10. **Memasukan model K terbaik ke dalam model data untuk di Training**

```
k = 4
#Train Model and Predict
knn = KNeighborsClassifier(n_neighbors = k).fit(X_train, Y_train)
```

11. **Menentukan prediksi nilai y**

```
yhat = knn.predict(X_test)
print(yhat)
```

```
[0 1 0 ... 1 0 0]
```

12. **Menghitung akurasi dari K=4**
    Akurasi K=4 sangat tinggi yakni 94,2% di data *training* dan 92,65% di data *testing*.

```
from sklearn import metrics
print("Train set Accuracy: ", metrics.accuracy_score(Y_train, knn.predict(X_train)))
print("Test set Accuracy: ", metrics.accuracy_score(Y_test, yhat))
```

```
Train set Accuracy:  0.942375
Test set Accuracy:  0.9265
```

Output Latihan:

Berdasarkan hasil pencarian nilai K terbaik dari proses klasifikasi data menggunakan KNN *classification* dengan percobaan nilai K dari 1-10, ditemukan bahwa model terbaik akan didapat ketika nilai K= 4. Hal ini karena pada nilai K= 4, akurasi yang didapat paling tinggi dibandingkan nilai K yang lain. Akurasi pada nilai K= 4 adalah 94,2% di data *training* dan 92,65% di data *testing*.