Chapter 9 - Exercise 1: Wine

Yêu cầu: Áp dụng Cross Validation cho bài Wine đã làm trước đó.

In [1]:
```
# from google.colab import drive
# drive.mount("/content/gdrive", force_remount=True)
```

In [2]:
```
# %cd '/content/gdrive/My Drive/LDS6_MachineLearning/practice/Chapter9_KyThuatBo
```

In [3]:
```
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn import svm
from sklearn.model_selection import train_test_split
import numpy as np
import pandas as pd
```

In [4]:
```
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
```

In [5]:
```
data = pd.read_csv('wine.data.txt', sep=',', header= None)
#data.info()
```

In [6]:
```
data.head()
```

Out[6]:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	1	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
2	1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
3	1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
4	1	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735

In [7]:
```
X = data.iloc[:, 1:14]
y = data.iloc[:, 0]
```

In [8]:
```
X.head()
```

Out[8]:

	1	2	3	4	5	6	7	8	9	10	11	12	13
0	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
2	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
3	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
4	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735

In [9]:
```
y.head()
```

Out[9]:
```
0    1
1    1
2    1
3    1
4    1
Name: 0, dtype: int64
```

Cross validation

In [10]:
```
from sklearn import svm
```

In [11]:
```
# 70%, 75%, 80% training and 30%, 25%, 25% test
test_size_lst = [0.3, 0.25, 0.2]
for i in test_size_lst:
    X_train_1, X_test_1, y_train_1, y_test_1 = train_test_split(X, y,
                                                                test_size=i)
    clf1= svm.SVC(kernel='linear')
    clf1.fit(X_train_1,y_train_1)

    score_train = clf1.score(X_train_1, y_train_1)
    score_test = clf1.score(X_test_1, y_test_1)

    print("With [", 1-i, ":", i, "], score train is ", round(score_train,2),
          ", score test is", round(score_test,2),
          "diff is", round(abs(score_train-score_test),2))
```

```
With [ 0.7 : 0.3 ], score train is  0.99 , score test is 0.96 diff is 0.03
With [ 0.75 : 0.25 ], score train is  0.99 , score test is 0.98 diff is 0.01
With [ 0.8 : 0.2 ], score train is  0.99 , score test is 0.94 diff is 0.05
```

In [12]:
```
# Compare: 70%-30%, 75%-25% and 80%-20%
# Choose the best one
# (Can run many times to make sure your choice)
```

K-folds

In [13]:
```
from sklearn import model_selection
from sklearn.model_selection import KFold
```

In [14]:
```
clf_k=svm.SVC(kernel='linear')
kfold = KFold(n_splits=10, random_state=42)
results = model_selection.cross_val_score(clf_k, X, y, cv=kfold)
print("Accuracy: %.3f%% (%.3f%%)" % (results.mean()*100.0,
                                     results.std()*100.0))
```

```
Accuracy: 94.444% (7.027%)
```

In [15]:
```
results
```

Out[15]:
```
array([1.        , 0.94444444, 1.        , 0.77777778, 0.88888889,
       0.94444444, 1.        , 0.88888889, 1.        , 1.        ])
```

In [16]:
```
# Nhận xét: Model có tính ổn định khá tốt.
```

Bổ sung: Turning Parameter, Select model