Université de Cergy-Pontoise

Introduction to Data Sciences – TD Classification I

Part I Examples

Write the comments to explain how the following programs work.

Example I.

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn import tree

balance_data = pd.read_csv('balance-scale.data', sep= ',', header= None)
balance_data.shape
X = balance_data.values[:, 1:5]
Y = balance_data.values[:,0]
X_train, X_test, y_train, y_test = train_test_split( X, Y, test_size = 0.3)

clf_entropy = DecisionTreeClassifier(criterion = "entropy", max_depth=3,
min_samples_leaf=5)
clf_entropy.fit(X_train, y_train)
y_pred_en = clf_entropy.predict(X_test)
print ("Accuracy is ", accuracy_score(y_test,y_pred_en)*100)
```

Example II.

```
import numpy as np
...
import graphviz

Jogging_data=pd.read_csv('JoggingTitre.csv', sep=',')
y=Jogging_data['Jogging']
x=Jogging_data.drop(['Jogging'], axis=1)
x_dum=pd.get_dummies(x)
x_dum
# partiel dummies : sub_dum=pd.get_dummies(x, columns=['Temps', 'Vent'])

clf_entropy = DecisionTreeClassifier(criterion = "entropy")
outputTree=clf_entropy.fit(x_dum, y)

dot_data = tree.export_graphviz(outputTree, out_file=None)
graph = graphviz.Source(dot_data)
graph.render("Td2_dum01")

dot_data = tree.export_graphviz(outputTree, out_file=None, feature_names =
x_dum.columns)
graph = graphviz.Source(dot_data)
graph.render("Td2_dum01Name")
#Check and compare the files Td2_dum01.pdf and Td2_dum01Name.pdf
```

Example III.

```
import numpy as np
...
from sklearn.model_selection import KFold

balance_data = pd.read_csv('balance-scale.data', sep= ',', header= None)
X = balance_data.values[:, 1:5]
Y = balance_data.values[:,0]

kfold = KFold(10, True, 10)
ac=0.0
ac_score=0.0
for train, test in kfold.split(X):
        X_train, X_test, y_train, y_test = X[train], X[test], Y[train], Y[test]
        clf_entropy.fit(X_train, y_train)
        y_pred_en = clf_entropy.predict(X_test)
        ac_score=accuracy_score(y_test,y_pred_en)*100
        ac=ac+ac_score
        print ("Accuracy is ", ac_score)

ac_avg=ac/10
print ("Average Accuracy is ", ac_avg)
```

Part II Exercises

Execute the following analyses and observations with the data sets :

car.data,
flag.data,
agaricus-lepiota.data,
tic-tac-toe.data, and
zoo.data

There are some information for these data sets in the ****.name*** files. You can get more of information related to these data sets from the UCI Web site

https://archive.ics.uci.edu/ml/datasets.html

Exercise1

- When using the ***DecisionTreeClassifier***, we can specify the parameter ***max_depth*** to limit the maximal depth of the decision tree. Try the *max_depth* from 3 to 10 (or to the real maximal depth of the classifier) to observe whether the maximal depth brings some impacts to the accuracy of the classifier.
- For each *max_depth*, you use the K-fold cross-validation method, for K=10, to compute the average accuracy of the classifier.
- Are there relationships among the size of datasets (the number of attributes and the number of records etc.), Decision Tree Depth, and the accuracy of classifier?
- Write your observations and put some graphical representations for a better understanding your description.

Exercise2

- Try to build the decision Tree Classifier with different sizes of training data set (10%, 25%, 33%, 50%, 66% and 75% of the whole data set).
- For each size of training data set, use the **R*andom Subsampling*** method to computer the average accuracy of the classifier. More precisely, you extract the randomly the same size of training data set to train the classifier and test its accuracy with the other part of data set. Repeat 10 times this procedure to compute the average accuracy.
- Does it exist relationships between the size of training data and the classifier accuracy? With some graphical representations, explain your observations.

Final Report

In your report, you have to return

- the programs with comments of Part I;
- python programs used for the exercises in Part II
- observations from Exercise 1 and Exercise 2 of Part II