Fundamentals of Data Science

Assignment 2

Objective

In this assignment, you will implement different predictive modeling approaches based on the random forest classifier and naïve Bayes classifier using Python.

Detailed requirement

Random forest is an ensemble predictive modeling approach which combines multiple decision trees, with each tree modeling a different aspect of the data set. Specifically, each component tree is constructed by sampling the original training set with replacement to create a new training set, based on which the tree structure is determined. In addition, a random subset of attributes, instead of the complete set of attributes, is used for evaluating the best attribute for splitting the data records at each node. A random forest model can be constructed by using the `ensemble` module in the Python package `scikit-learn`.

In this assignment, you will implement and evaluate different predictive modeling approaches using the ***Vertebral Column*** data set in Assignment 1.

You may visualize the component trees of the random forest model using the package `python-graphviz`.

Assignment Submission

You should submit a report to summarize your work. For the above data set, the following tasks are to be performed:

(a) Construct random forest models using different numbers of component trees based on a specific training set/test set partition, and analyze the resulting change in classification performance. (25%)

(b) For the random forest model corresponding to the best classification performance, select different component decision trees in the model, and compare the classification performances of these trees with that of the original random forest model. (25%)

(c) For a random forest classifier (or one of its component trees), the relative importance of the attributes can be measured through the `feature_importances_` field of the classifier. For selected component trees in (b), compare their associated lists of relative attribute importance values. (25%)

(d) Construct a naïve Bayes classifier model based on our data set, and compare the classification performance with that of the random forest model. (25%)

Please provide a detailed description of the results of the above tasks in your report.

Supplementary Instructions for Assignment 2

To construct random forest and naïve Bayes models in Python, we need to include the following modules:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB
import numpy as np
```

We again use the Iris data set as an example:

```
from sklearn import datasets
iris = datasets.load_iris()
irisTest =
np.array([[4.6,3.5,1.1,0.25],[5.7,2.5,2.8,1.2],[7.3,2.8,6.6,2.2]])
```

A random forest model can be constructed as follows:

```
clf = RandomForestClassifier(n_estimators = 6)
clf = clf.fit(iris.data, iris.target)
prediction = clf.predict(irisTest)
print(iris.target_names[prediction])
```

Information about the component trees of the random forest classifier can be obtained using the `estimators_` field of the classifier. For example, we can display information about the first component tree in the random forest as follows:

```
clf.estimators_[0]
```

A naïve Bayes model can be constructed as follows:

```
nb = GaussianNB()
nb = nb.fit(iris.data, iris.target)
prediction = nb.predict(irisTest)
print(iris.target_names[prediction])
```