Monica-Mohan / **Random-Classification-1** Public

forked from HEMALATHA2021/Random-Classification

Code Pull requests Actions Projects Wiki Security Insights Settings

main

Random-Classification-1 / README.md

Monica-Mohan Update README.md

2 contributors

```
44 lines (37 sloc) | 1.94 KB
```

EX NO: 01

DATE:

RANDOM CLASSIFICATION

AIM:

To write a python program to perform random classification.

EQUIPMENTS REQUIRED:

1. Hardware – PCs
2. Anaconda – Python 3.7 Installation / Google Colab /Jupiter Notebook

RELATED THEORITICAL CONCEPT:

Random classifier: It creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object. Random Forest is suitable for situations when we have a large dataset, and interpretability is not a major concern.

Purpose of Random classifier: One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification.

ALGORITHM:

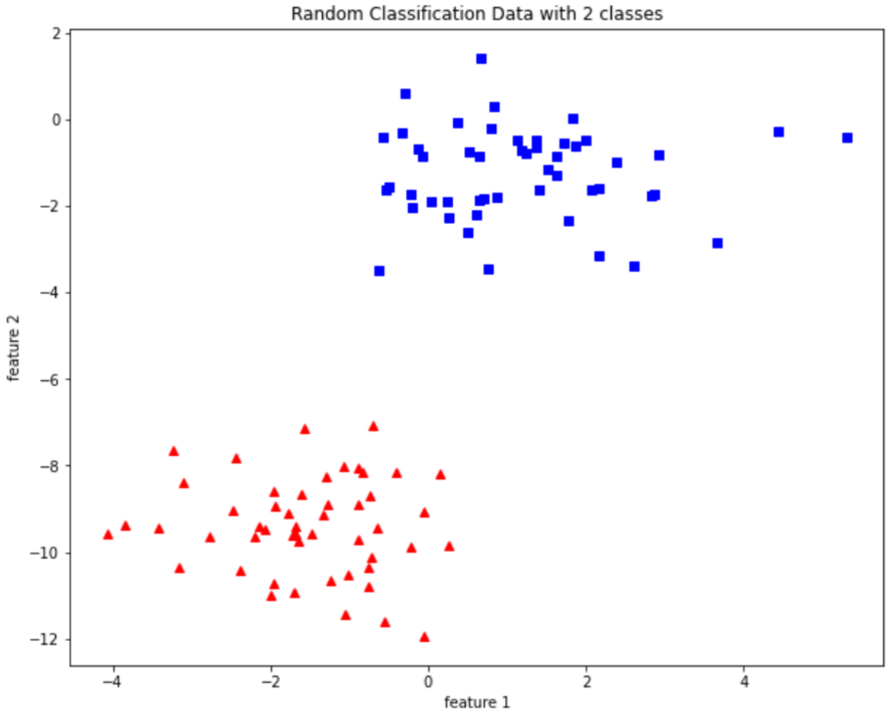
1. In Random forest n number of random records are taken from the data set having k number of records.
2. Individual decision trees are constructed for each sample.
3. Each decision tree will generate an output.
4. Final output is considered based on Majority Voting or Averaging for Classification and regression respectively.

PROGRAM:

```
/*
Program to implement random classification.
Developed by   : VIJAYARAGAVAN ARR
Register Number : 212220230059
*/
import matplotlib.pyplot as plt
from sklearn import datasets
x,y=datasets.make_blobs(n_samples=100,n_features=2,centers=2,cluster_std=1.05,rar
fig=plt.figure(figsize=(10,8))
plt.plot(x[:,0][y==0],x[:,1][y==0],'r^')
plt.plot(x[:,0][y==1],x[:,1][y==1],'bs')
plt.xlabel("feature 1")
plt.ylabel("feature 2")
plt.title('Random Classification Data with 2 classes')
```

OUTPUT:

```
Text(0.5, 1.0, 'Random Classification Data with 2 classes')
```



RESULT:

Thus the random classifier was successfully implemented using python programming.