CIS 662 HW7 Report: Random classifier

**PART 1: Use the original citation for training Random Forest Classifier:**

Dataset selected is **71-80.csv**

Data has also been split in the same way as the previous homework 80:20 to compare the results.

A screenshot of a computer code

Description automatically generated

Data contains all continuous numerical values, however, to convert the target variable to a categorical variable we first take the ratio of **cit\_2022/cit\_2021** and round it to **2 decimals.**

**Binning** strategy is used to distribute the data present in different bins i.e Separate data into different categories.

Thus 3, separate labelled categories can be obtained as follows:

**1. Low (<1.05):** Values between -np.inf (inclusive) and 1.05 (exclusive) will be labelled as "Low".

**2. Medium (1.05-1.15):** Values between 1.05 (inclusive) and 1.16 (exclusive) will be labelled as "Medium".

**3. High (>1.15):** Values greater than or equal to 1.16 will be labelled as "High".

I have not scaled/normalised the data as scaling/normalisation has no significant influence on the output category for Decision Trees/ Random Forest.

We can observe from the above logic that the ratio of 2022/2021 influences the category for the citations.

Random forests build multiple decision trees by training each tree on a different subset of the data. Bootstrap sampling involves randomly selecting samples with replacement from the original dataset to create these subsets.

**Samples = sqrt(features)**

To determine the number of **bootstrap aggregations or samples(n\_estimators**) to be provided for the random forest classifiers for efficient training, I have written a small logic and plotted a graph using Matplotlib. For each sample, there might be features used in samples that can be repeated in multiple samples.

A screenshot of a computer program

Description automatically generated

Here, I used 5 different sample(bagging) values and passed them as a parameter to the Random Forest Classifier.

A graph with a line and a blue line

Description automatically generated

It can be observed that Accuracy is high where bootstrap sample value is 300. Hence, I have considered the n\_estimator(bagging) value as 300 for RF classifier.

**OUTPUT:**

**RF classifier code:**

A screenshot of a computer program

Description automatically generated

Based on the above results, the Model is trained with the training dataset with sample number 300 passed as a parameter to the Random Forest classifier for training.

**Output matrix:**

A screenshot of a computer screen

Description automatically generated

**CONCLUSION:**

Accuracy obtained is **75%.** Better Accuracy and F1 score can be obtained because the target variable is based on ratio of cit\_2022/cit\_2021, so when random classifier was trained using given original citation data, random classifier-built decision trees by trying to establish complex non-linear relationships with the target variable [ ratio of 2022/2021] by calculating mean decrease in Gini index for features which can be explained better below:

**Code for identifying the most important feature selected for classification.**

*A graph of blue bars

Description automatically generated*

As observed in the above screenshot, the classifier model tries to use the feature 2017 as it shows the mean decrease in Entropy value and can derive complex non-linear relationships with the target variable. Hence, it inaccurately classifies the data into 3 classes using the 2017 citation features based on spurious relationships and correlations.

As there is no direct linear/non-linear relationship of any feature variable with the target variable (ratio of cit\_2022/cit\_2021), the classifier randomly classifies the data as it has already established a spurious correlation with some values for feature column cit\_2017.

However, the results are not bad, but they are incorrect which can be observed in the above screenshot as the target variable is the ratio between cit\_2022/cit\_2021 and feature cit\_2017 is used majorly as the criteria for classification.

**PART 2: Evaluating results of new features for training Random Forest**

**Classifier**

For the second part of the question, the same classifier is used for classification, but there is a slight change in the feature values.

For the new classifier instead of using cit\_2017-cit\_2022 as feature variables we are using the updated logic for calculating new feature variables as mentioned below:

A screenshot of a computer program

Description automatically generated

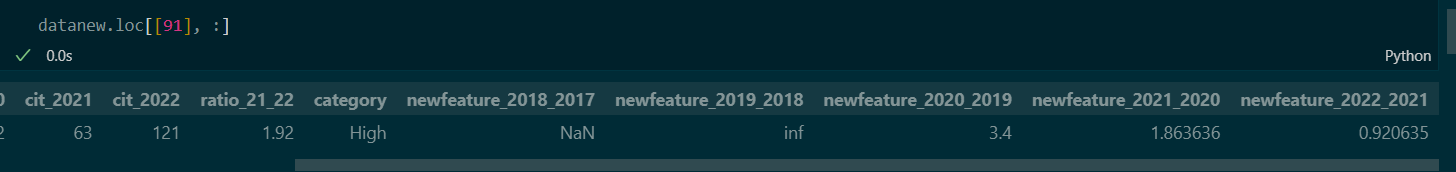
A screenshot of a computer

Description automatically generated

As shown in the above screenshot the formula for calculating the new feature:

**New\_feature = next\_year – curr\_year / curr\_year**

In the below screenshot, for the values where citations are 0 the ratio turns out to be **NaN** and **inf**.



I have handled such data by replacing the **-inf** value with a value lower than **the min** value and the **+inf** value with a value higher than **the max** value.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

I have used the same logic as mentioned in part 1 for estimating the number of bootstrap samples to be passed as a parameter to Random Forest classifier.

A computer screen shot of a program code

Description automatically generated

A graph with a line

Description automatically generated

Even for the new dataset with the new features, Accuracy score is higher with bootstrap sample size of 300.

Dataset is split into training and testing set in the ratio 80:20.

**OUTPUT:**

**A screenshot of a computer program

Description automatically generated**

**Output matrix:**

**A screenshot of a computer

Description automatically generated**

**CONCLUSION:**

Accuracy obtained is 100% for the testing set of the new dataset.

A screenshot of a graph

Description automatically generated

Since the target variable is obtained on the ratio of cit\_2022/cit\_2021 and the new dataset featured column containing the ratio **2022-2021/2021** is closer to the target variable as it shows **mean decrease in entropy/Gini index value** and since Random Forest classifier works by detecting non-linear patterns and correlations between the feature columns and the target variable, the model was able to accurately classify the test sample into the appropriate classes giving the accuracy and F1 score of 100%.