Preprocessing on Breast Cancer Dataset

Shashank Pathak(15BCE1287)*,
*School of Computing Sciences and Engineering, VIT Chennai, Tamilnadu, India 600127
Email: shashank.pathak2015@vit.ac.in*

[be innovative but precise with the title]

***Abstract*—Todays real-world dataset are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size (often several gigabytes or more) and their likely origin from multiple, heterogenous sources.So this low-quality data will lead to error-prone learning results.In order to making learning more efficient we should preprocess this data to remove noise,missing values and inconsistency.**

1. Introduction

For Learning to be efficient our data should will free from noise, missing value, misleading values or incorrect data and inconsistent data.The Breast Cancer Wisconsin Original dataset available publicly have many such aberrations that have to be removed in order to properly process the data for learning.So we will be using some techinques like data cleaning, data scaling.

2. Methodology

So for Preprocessing, first the data needs to be formated to ba proper form in which we would be able to process the data.The data may be in a relational database and you would like it in a flat file, or the data may be in a proprietary file format and you would like it in a relational database or a text file. After that the dataset needs to be cleaned.For the Breast Cancer Wisconsin Dataset cleaning included filling in missing values, smooth noisy data, identify or remove the outliers, and resolve inconsistencies.Data cleaning is required because source systems contain dirty data that must be cleaned. Certain time there will be a need to transform the attributes present in the dataset into a particular form,this could be done either by decomposition or by aggregation.The transformation on dataset will be influenced by the algorithm that we are using for learning and the problem domain.Scaling the data is one such transformation technique.

3. Database - Breast Cancer Wisconsin Dataset(Original)

The Wisconsin Breast Cancer datasets from the UCI Machine Learning Repository is used,to distinguish malignant (cancerous) from benign (non-cancerous) samples.This dataset consist of 699 instances and 11 attributes that would help the classiy the data into the two classes.The attributes are described in the fig.1

#	Attribute	Domain
1.	Sample code number	id number
2.	Clump Thickness	1 - 10
3.	Uniformity of Cell Size	1 - 10
4.	Uniformity of Cell Shape	1 - 10
5.	Marginal Adhesion	1 - 10
6.	Single Epithelial Cell Size	1 - 10
7.	Bare Nuclei	1 - 10
8.	Bland Chromatin	1 - 10
9.	Normal Nucleoli	1 - 10
10.	Mitoses	1 - 10
11.	Class:	(2 for benign, 4 for malignant)

Figure 1. Breast Cancer Wisconsin Dataset Attributes

4. Algorithm

We need to check for the null values of any attribute and if null value is found in any of the rows then we need to remove that entry.

df.isnull().any()

Next we check for matching attributes(attributes which give same information but in different form for example one attribute may give true or false based on glowing of bulb and the other may give 0 or 1 based on glowing of bulb). Since these matching attributes give same result we drop one of the attributes

del df['Attribute Name']

There is a specified range for every attribute and if the attribute value is not in the specified range then we drop those absurd value.

5. Experiments

First we will analyse the data usinf the pairplot function in seaborn library to plot a figure as shown in fig.2

[H] sb.pairplot(df,vars=cols,hue='Class')

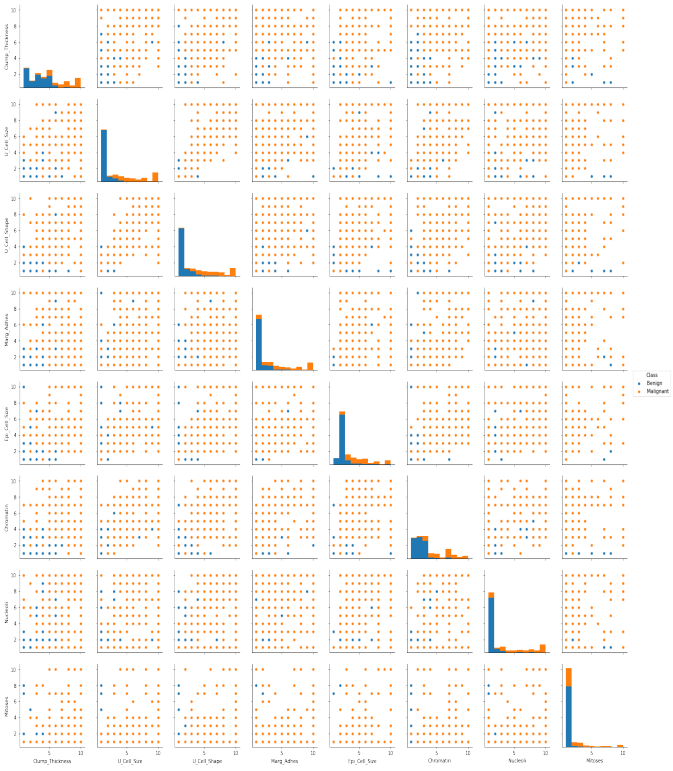

Figure 2. Pair Plot Between different attributes

Now we will make a histogram of Clump Thickness for those cases which are labelled as as malignant using the following command

df.loc[....].hist()

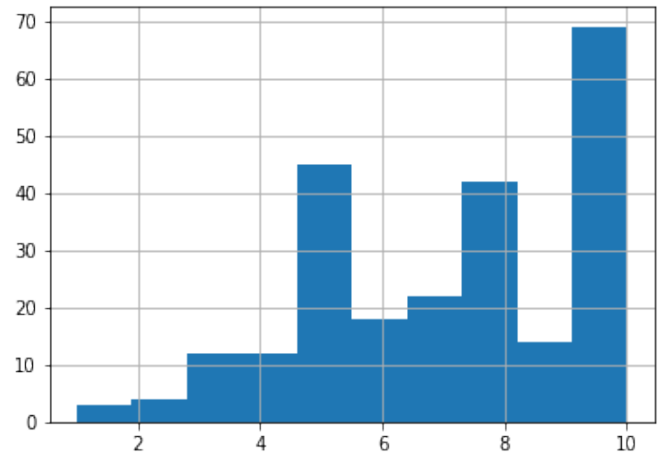

Figure 3. Histogram

After that we used matplotlib to plot violin plot and box plot of our dataset on 9 attributes which have a value between 1 and 10 and thus able to analyse our data in different aspects

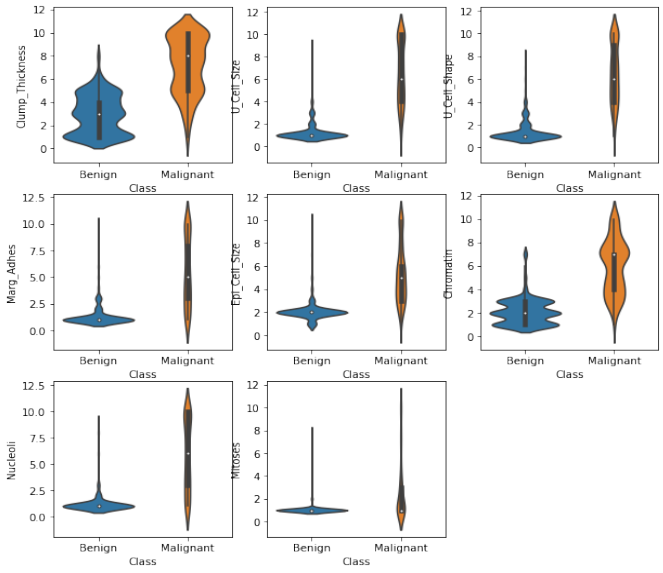

Figure 4. Violin Plot

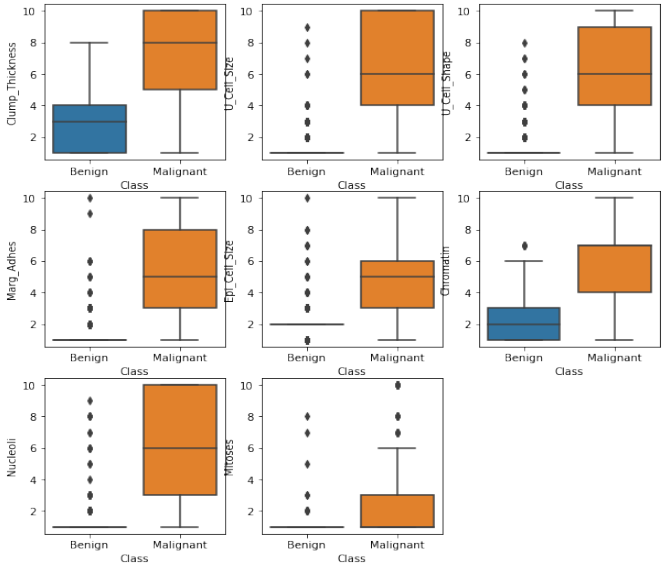

Figure 5. Box Plot

6. Conclusion

So due to preprocessing we were able to make our data better for the learning process .Also we were able to analyse and gain useful information about the possible attributes that have effect on the classification.We also learned how to use a particular dataframe from the given dataset.Therefore preprocessing in a useful and necessary step in order to learn effectively and efficently.