**Heart Disease Prediction-Group\_1**

**CAP 5610 Machine Learning**

**Milestone #3 Report**

# **Revised Project Statement:**

For the Heart Disease Dataset, we want to classify the presence of heart disease using the best combinations of the dataset files (Cleveland, Hungarian, Switzerland, VA) based on the most relevant patient attributes.

# **Heart Disease Dataset:** [UCI Repository- Link to Dataset](https://archive.ics.uci.edu/ml/datasets/Heart+Disease?spm=5176.100239.blogcont54260.8.TRNGoO)

This database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0). The names and social security numbers of the patients were recently removed from the database, replaced with dummy values. The input variables for all the databases used are: *age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca,* and *thal* while the output variable used is *num*.   
  
Despite the Cleveland database only being used for research to this date, we have decided to test different combinations of databases rather than only using the Cleveland database, along with imputing the missing values to see if we can achieve better results.

# **Exploratory Data Analysis (EDA)**

## **Loading the datasets and checking for Missing Values**

After loading the dataset files for Cleveland, Hungarian, Switzerland, and VA, the exploration of the data showed that the Cleveland dataset had 303 rows and 14 columns, the Hungarian dataset had 294 rows and 14 columns, the Switzerland dataset had 123 rows and 14 columns, and the VA dataset had 200 rows and 14 columns.

The datasets were further investigated for missing values and the following missing values were found:

* *Cleveland: 6 missing values*
* *Switzerland: 273 missing values*
* *Hungarian: 782 missing values*
* *VA: 698 missing values*

Finally, the descriptive statistics for each feature in each dataset were produced to identify type errors or zero rows. The results of the descriptive statistics are shown in the accompanying Jupyter Notebook for this Milestone of the project.

## **Basic Data Cleaning**

Once the missing values were checked, the next step was to look for redundancy within the datasets. First, we checked for redundant columns/features in the datasets and as a result of the investigation, we found that no column had only one value and the columns with more than one values were categorical variables. Second, we checked for duplicate rows in the datasets and as a result of this investigation, we found that the duplicate values might exist due to the missing values.

## **Imputation to replace missing values**

At this point, we decided to impute the missing values before we could proceed with further exploration and investigation of the datasets. As mentioned before, our team wanted to see if we can use different combinations of datasets rather than only using the Cleveland dataset to achieve better results.

The procedure for imputation of missing values is as follows:

The 5 combinations of datasets that we decided to use were:

* *Cleveland, Hungarian, Switzerland, and VA*
* *Cleveland, Hungarian*
* *Cleveland, Switzerland*
* *Cleveland, Hungarian, and Switzerland*
* *Cleveland (For comparison with other combinations)*

The Imputation methods we decided to use on the aforementioned combinations were:

* *KNN Imputer - Odd k values ranging 1 through 50*
* *Iterative Imputer - Order = ascending, descending, roman, arabic, and random*

Other changes made to the data for imputation:

* *The inputs were normalized using MinMax Scaler*
* *The Target variable was changed to binary (0: Absence of heart disease, 1: Presence of heart disease)*

Evaluation methods to see which combination has a better *accuracy* score:

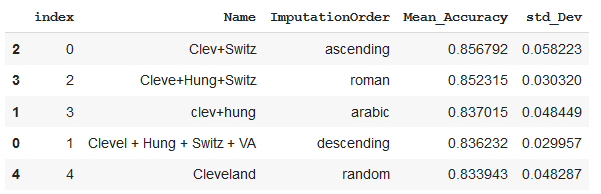
* *Model used: RandomForestClassifier - default*
* *Cross Validation: RepeatedStratifiedKFold - 10 folds, repeated 3 times, with a random state of 1*

The results of the imputation:

* KNN



* Iterative



After the imputation of the missing values using RandomForestClassifier and RepeatedStratifiedKFold to measure which combination would be better for classification, we decided that Imputation with KNN (K=3) produced the highest mean accuracy and the lowest standard deviation for the combination of Cleveland, Switzerland, and Hungarian datasets.

We decided to use the imputer method and combination of datasets mentioned above to produce our final dataset for this Project.

## **Analysis of the Imputed Dataset (Final Dataset)**

The Final dataset produced after the imputation on the combination of Cleveland, Switzerland, and Hungarian datasets had 720 rows and 14 columns. Upon checking for missing values on the Final dataset, no missing values were found, which meant all missing values were replaced as a result of the imputation. Due to the missing values that existed before imputation, the reduction of redundancy on the raw datasets did not make much sense. After the imputation, the Final Dataset was checked for redundancy and as a result, one duplicate row which existed was removed. Furthermore, the type of variables for the Final Dataset were investigated and findings showed that the input variables were scaled while the target variable remained binary. Additionally, we decided to check that the class distribution is balanced as there are 359 observations in class 0 and 360 observations in class 1.

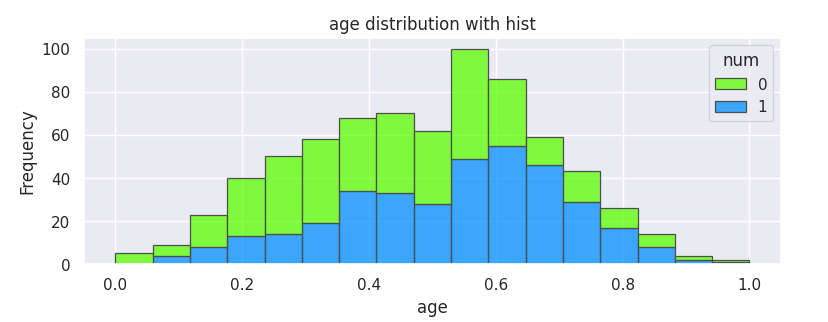
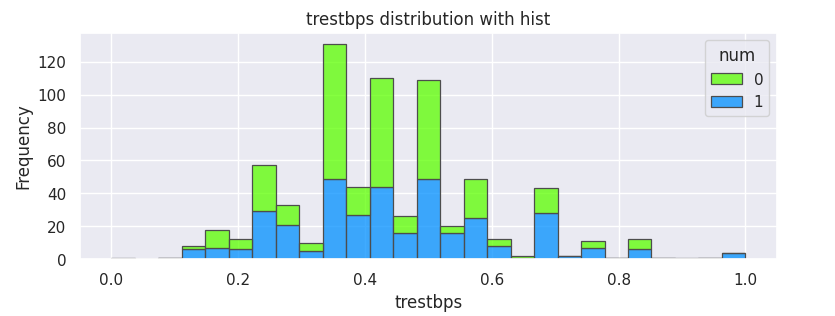
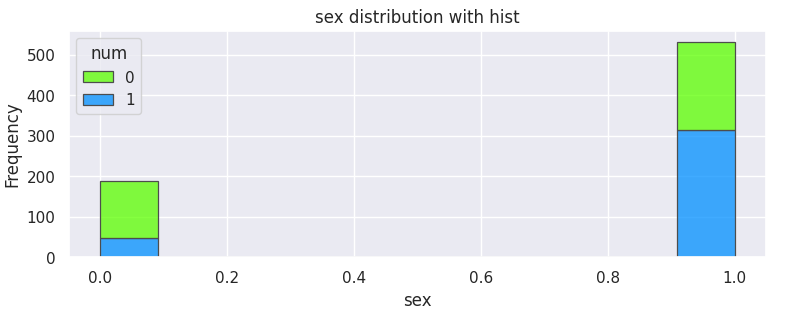
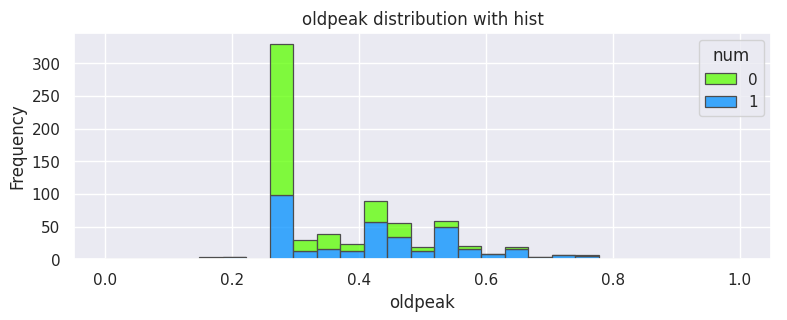
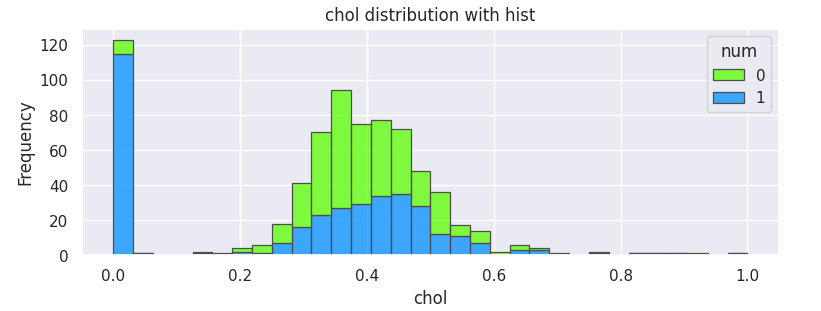
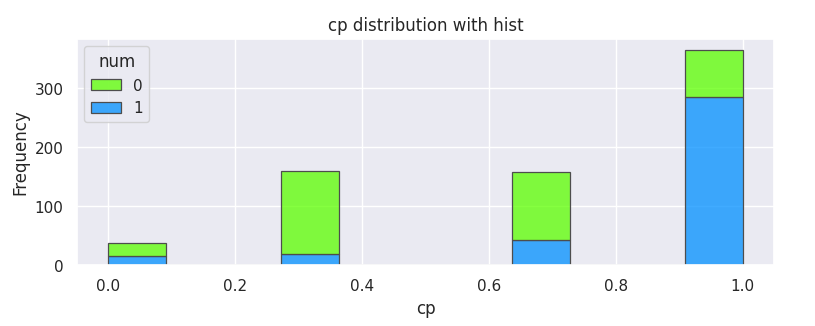
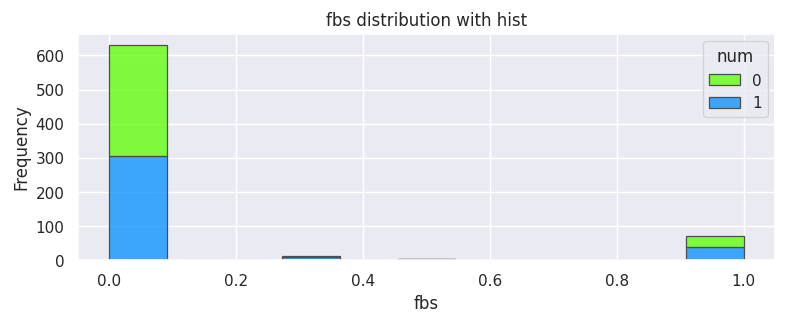
After double-checking for missing values and exploring the Final Dataset, the descriptive statistics were produced for the Final Dataset and the results are shown in the accompanying Jupyter Notebook for this Milestone of the project. Lastly, the correlation between the attributes and skewness of the attributes was produced. At this point, we decided that we needed visualizations for the descriptive statistics, correlation between the attributes, distribution of the dataset, and skewness of the attributes in order to produce more meaningful statistical inferences for the Final Dataset.

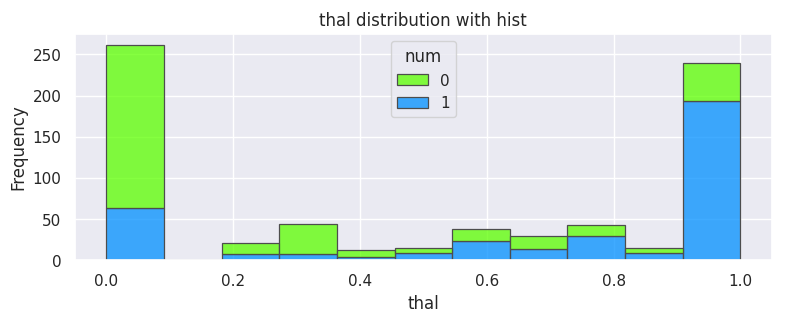
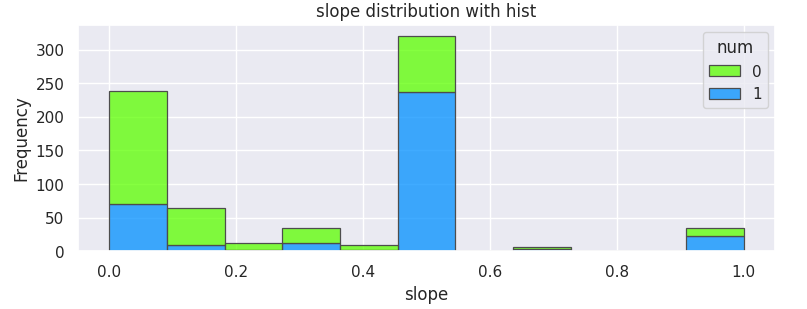
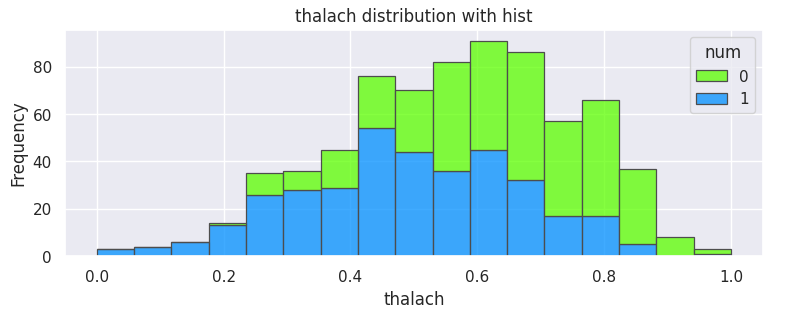
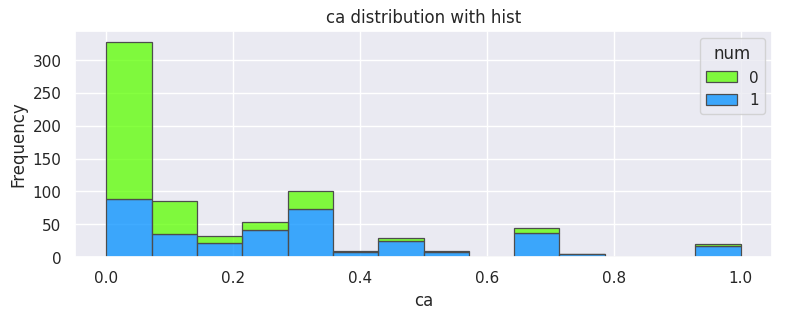
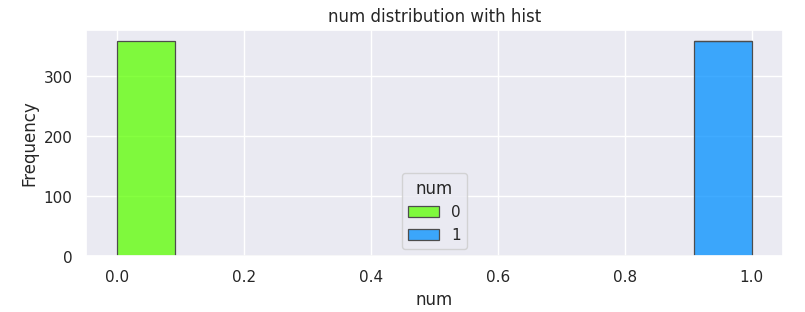
## **Visualizing the Final Dataset**

For the visualization of the Final Dataset and its attributes, we decided to produce:

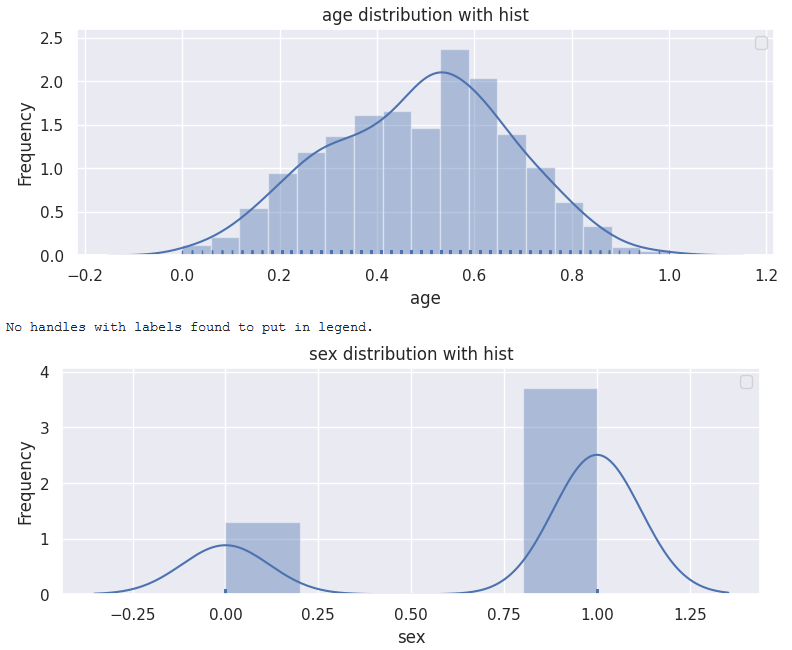
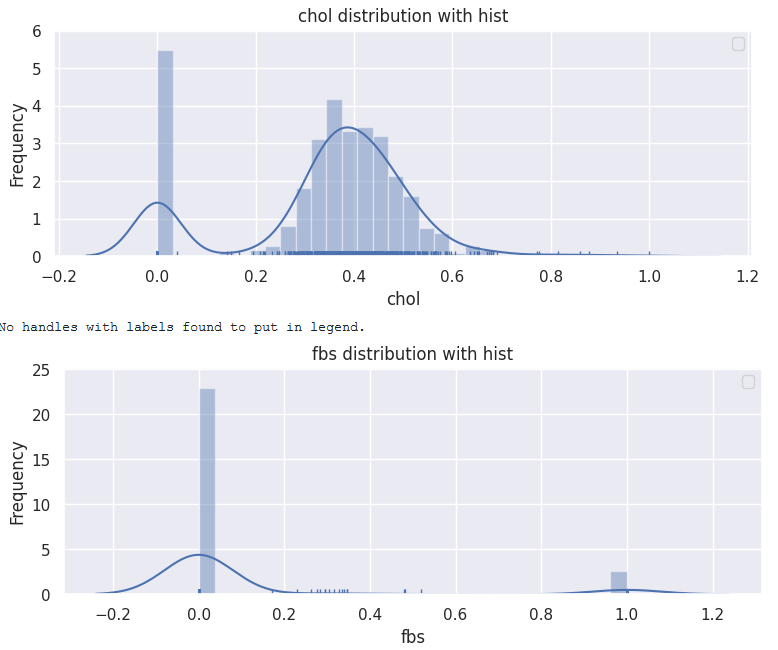
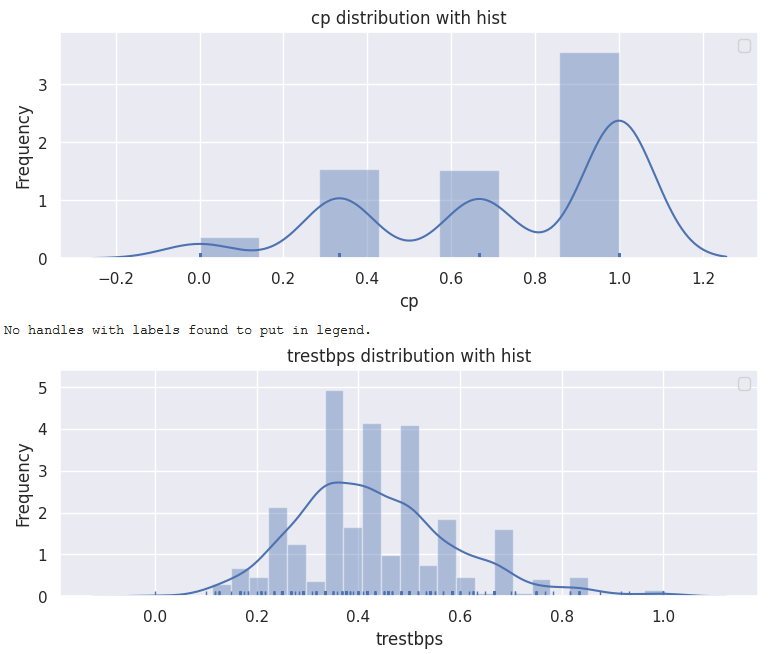
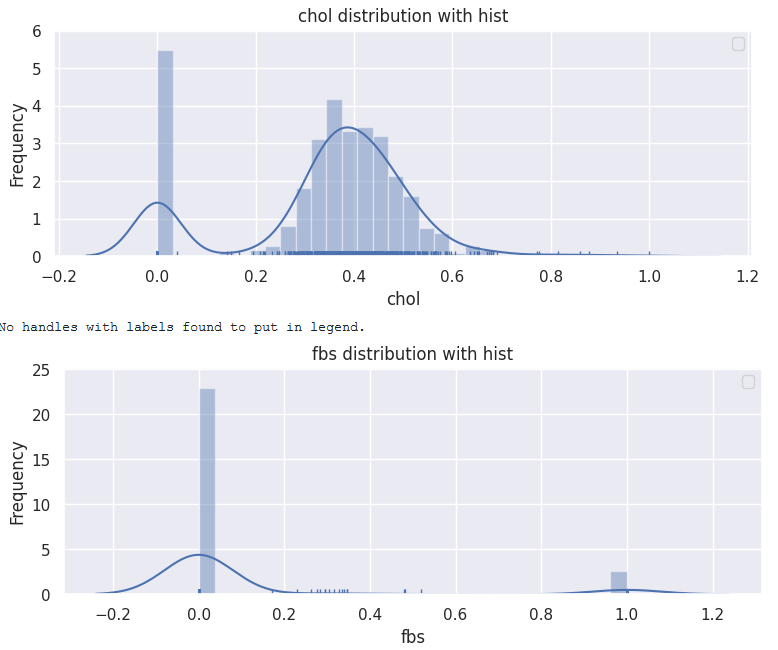
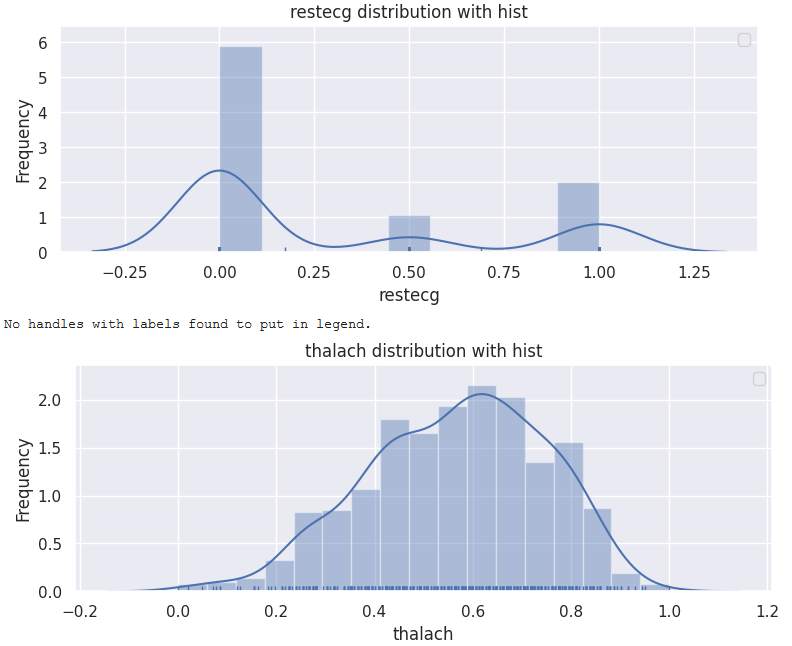
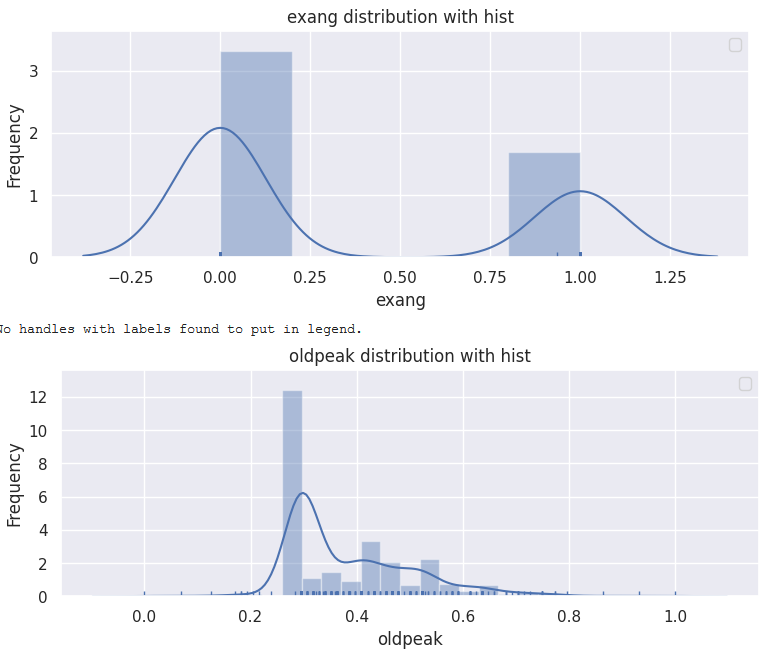
* Unimodal Visualization: *Histogram, Density Curve, and Box Plot*
* Multimodal Visualization: *Scatter Plot and Correlation Matrix*

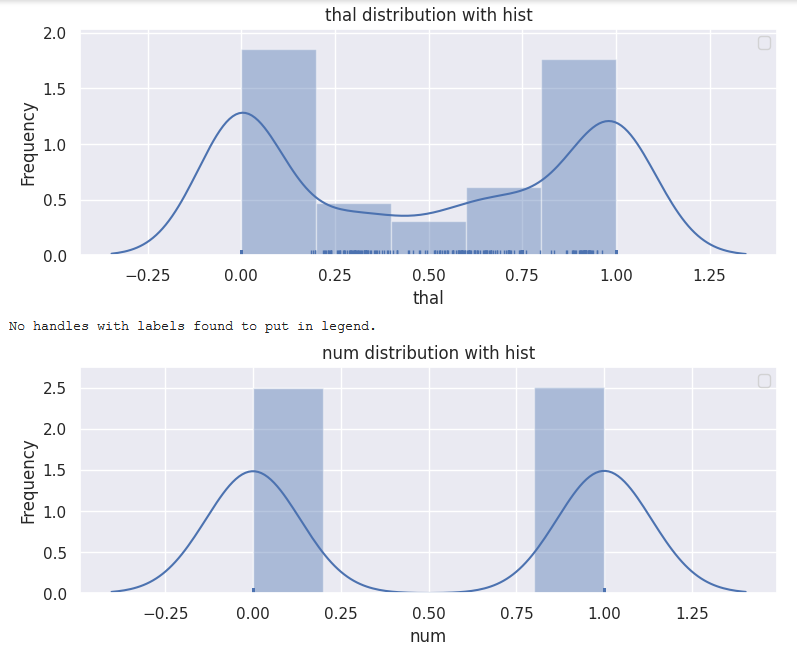
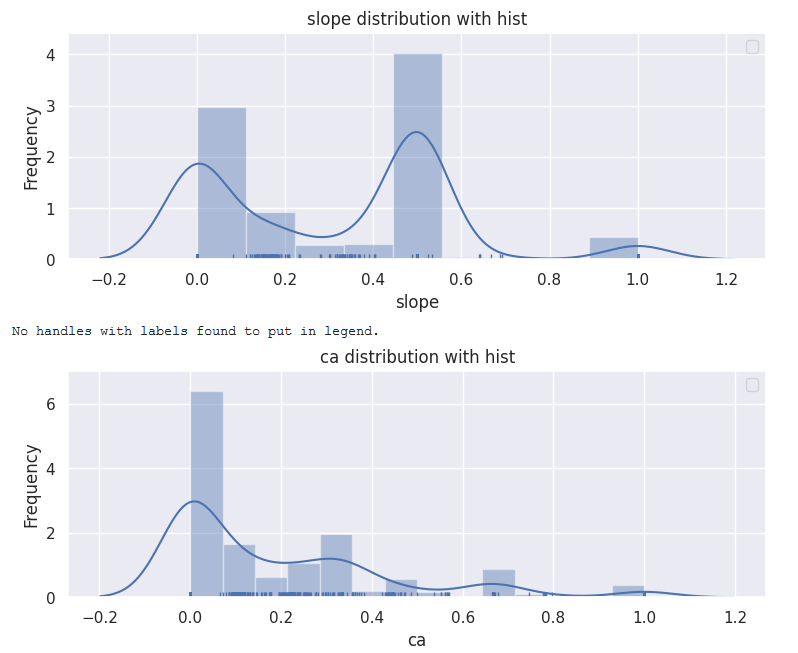
***Histograms of the attributes***



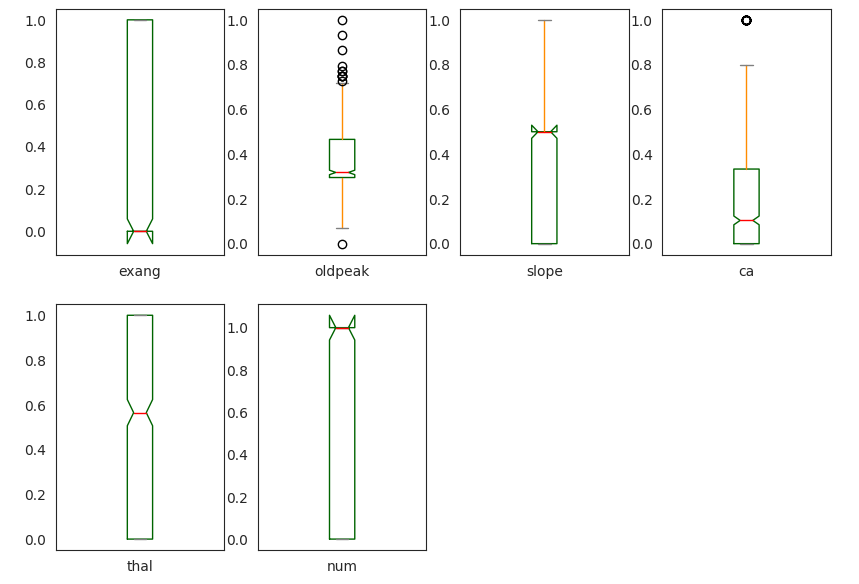
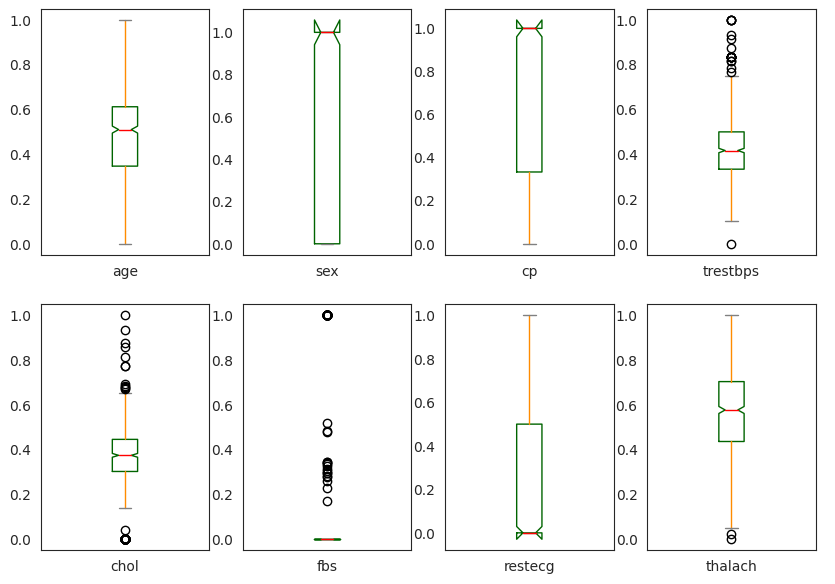


***Density Curves for the attributes***





***Box Plots for the attributes***



***Scatterplot for Multimodal Analysis***



***Correlation Matrix for Multimodal Analysis***

## **Interpreting the Visualizations**

After investigating the visualizations, the findings from the histograms and density curves stated that some of the quantitative attributes are skewed. These findings suggested that standardizing the data might produce better results. Moreover, the investigation of the boxplots indicated that some of the variables have outliers which might affect the results of some of the algorithms. Although, we decided to not remove the outliers since the data is concerned with the medical field and the outliers could work in our favor in producing accurate results. However, results produced with and without the outliers would give us the final verdict. Furthermore, upon investigating the Scatter-plot and the correlation matrix, the findings suggested strong correlation between the output variable, *num*, and some input variables such as *thal, ca, slope, oldpeak, exang, thalach, cp, and sex,* which indicated that these predictors could produce more accurate results.

After analyzing the data through producing statistics and visualizations, we produced a few questions that need to be investigated for the dataset:

* Would fewer predictors rather than all the predictors in classifying the presence of heart disease produce a more accurate result?
* If yes, which predictors are the best?
* Are the predictors controllable?

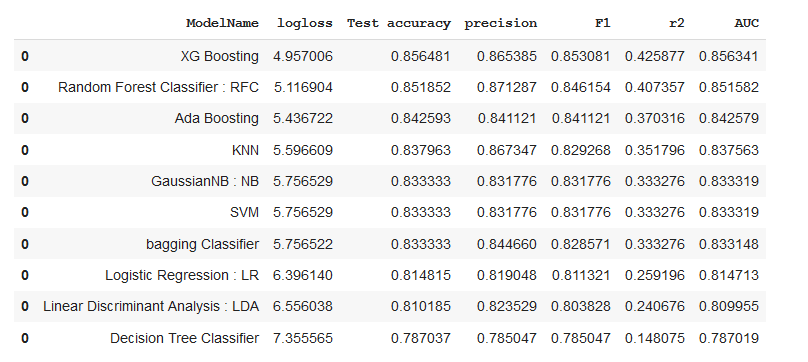
## **Baseline Model**

At this point, we decided to use all the predictors to build our Baseline Model. Before building a Baseline Mode, we decided to split the data into two parts: *70% training* and *30% testing*. After that, we considered 10 classification algorithms for model selection*: Logistic Regression, KNN Classifier, Support Vector Machine, Random Forest Classifier, Ada Boosting, Decision Tree Classifier, Bagging Classifier, XG Boosting, Linear Discriminant Analysis, and Naive Bayes*.

A spot-check along with the ROC curve and confusion matrix for the algorithms was performed to measure the most effective model performances that produced the best results. Furthermore, we have plotted the variable importance for the tree classifiers to understand the splitting in the data.

During the spot-check for algorithms, performances of the algorithms were measured using metrics such as: prediction accuracy, log loss, precision, F1- score, R-Square, and AUC.

After running the 10 aforementioned models, the results were as follows:



Based on the findings above, the results indicated that *XG Boosting, RandomForest, Ada Boosting, KNN Classifier, and SVM* were performing in the most effective manner and producing the best results.