ETL Summary:

A Jupyter notebook was used to clean the original data set. Once the file was loaded to the notebook, the data was put into a Pandas dataframe. Each input variable was inspected for any missing data. If a row was missing an input, the row was dropped from the dataframe. This created a dataframe with 13 input categorical and numerical variables with one categorical output variable indicating if the patient was healthy or sick with heart disease. The cleaning process condensed the original file from 303 entries down to 296. This cleaned data was then saved into a new csv file for reference.

As part of the ETL process to develop machine learning models, the 13 different input variables were evaluated through several feature selection techniques to better understand their potential influence to determining the output variable if someone was healthy or sick.

First, the numerical inputs like age and maximum heart rate were plotted on a histogram to understand if any of the input data was skewed. The only numerical input variable that appeared skewed was “old peak” which refers to a finding on an electrocardiogram related to ST depression induced by exercise. The majority of patients have an old peak value of 0. It is not possible to have a negative old peak value, which makes the histogram appear to have a right skewed distribution.

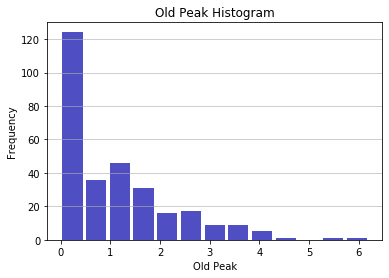
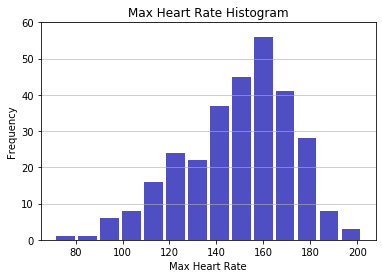


Figure 1. Example of a normal distribution with the max heart rate (left) compared to right skewed distribution in the old peak variable (right)

Next, Sci-Kit Learn feature selection functions were used to evaluate the importance of the different inputs. In order to use these functions, the data had to treated in the way the team wanted to analyze the data with the models. Therefore, a function was created to split, encode, and scale the data as it would be used for the models. Within the function, the split test and train data was saved into csv files, using a random seed to prevent the data from splitting into new groups each time the code was run. These are the csv files used to train the different machine learning models. Numerical inputs were scaled with the MinMax scaler function from Sci-kit Learn, while categorical input data was encoded using Pandas get\_dummies function. Finally, the categorical output data was encoded with the Sci-Kit Learn LabelEncoder function.

The first Sci-Kit Learn feature selection function used was the SelectKBest function; an univariate chi-square analysis which is used to identify significant input features based on univariate statistical tests. As an input to the function, “chi2” was called out to apply chi-squared statistics to the data set, since the output is categorical. The return values were ranking scores from highest to lowest of the input data. Because the categorical inputs were encoded, each option for one categorical input variable was given its own score. For example, the “thal” input had the choices of “rev”, “norm”, and “fix”. “Thal\_rev” ranked high with a score of 34.2, while the “thal\_fix” ranked low with a score of 0.74. Overall, the most unintuitive result was cholesterol, which ranked near the bottom. Common sense would make the average person assume cholesterol to be a more significant contributor to heart disease.

Table 1. Chi-Square Score Rank of the Input Features

|  |  |  |
| --- | --- | --- |
| Input ID | Specs | Score |
| 27 | thal\_rev | 34.200294 |
| 17 | exerciseInducedAngina\_true | 33.418728 |
| 9 | chestPain\_asympt | 32.604218 |
| 26 | thal\_norm | 28.482451 |
| 21 | vesselsColored\_0.0 | 21.540903 |
| 20 | slope\_up | 18.872001 |
| 19 | slope\_flat | 15.823077 |
| 16 | exerciseInducedAngina\_fal | 14.756322 |
| 7 | chestPain\_abnang | 14.126050 |
| 10 | chestPain\_notang | 13.687783 |
| 23 | vesselsColored\_2.0 | 12.695136 |
| 5 | sex\_fem | 12.124316 |
| 22 | vesselsColored\_1.0 | 11.352036 |
| 4 | oldPeak | 10.745139 |
| 24 | vesselsColored\_3.0 | 7.004118 |
| 6 | sex\_male | 5.700838 |
| 13 | ecg\_abn | 3.529412 |
| 8 | chestPain\_angina | 2.826471 |
| 15 | ecg\_norm | 2.605742 |
| 3 | maxHeartRate | 1.955725 |
| 14 | ecg\_hyp | 1.548633 |
| 0 | age | 0.975826 |
| 25 | thal\_fix | 0.741422 |
| 18 | slope\_down | 0.684007 |
| 1 | trestBps | 0.355538 |
| 2 | cholesterol | 0.136939 |
| 12 | bloodSugar\_true | 0.022368 |
| 11 | bloodSugar\_fal | 0.004620 |

The next feature selection function used was the Extra Tree Classifier. It essentially applies the same logic as a decision tree in machine learning to determine an importance score to each classifier input. The top 15 inputs with the highest score were plotted in a horizontal bar chart to help visualize their scores relative to one another. Results from this function mirrored those from the SelectKBest function.

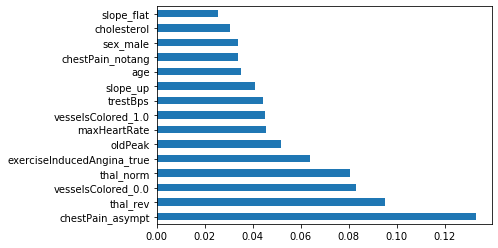


Figure 2. Top 15 scored classifier inputs by the Extra Tree Classifier function

The final feature selection technique used on the input data was a correlation matrix with a heat map to help visualize the results. This technique shows how input variables relate to each other. It helps to identify if any of the inputs are dependent on one another and may be confounded. The correlation function and a Seaborn heat map were used to create the graph. The top 5 identified features from the previous functions were mapped to help simplify the graph. Overall, the correlations between the tested features are relatively low, with the highest being between maximum heart rate and age at -0.4.



Figure 3. Input correlation results with Seaborn heat map

When deciding what data to include in the machine learning models, the team decided to not eliminate any from the original data set. The original data set that was cleaned for this ETL work had already been paired down by other individuals from a larger set with around 48 inputs.