Brian Hert
Dhruv Sharma
Richard Clinger
Prabhash Venkat Paila
CSC 177
8th, October 2024

Project 1 Data Preprocessing Report: Heart Disease Dataset

1. Introduction

In this Project, we used a heart disease data set and applied several preprocessing steps, which will be explained later in this report, to prepare the data for machine learning tasks. This report outlined the preprocessing techniques, the reason behind each technique, the results of our model after we split the dataset into training and testing, and our results when we compare the training to the testing model

2. Preprocessing Techniques

a. Duplicate Removal: We began by checking the dataset and removing any duplicate rows in it. By eliminating Redundant records, we ensured that each data point was unique.

b. Handle Missing Values: The dataset did not contain any missing values

```
No missing values to visualize.
```

c. Outlier Detection and Removal: Outliers were detected using z score normalization, and were removed beyond +- 3, This is because extreme values can influence the models training and performance.

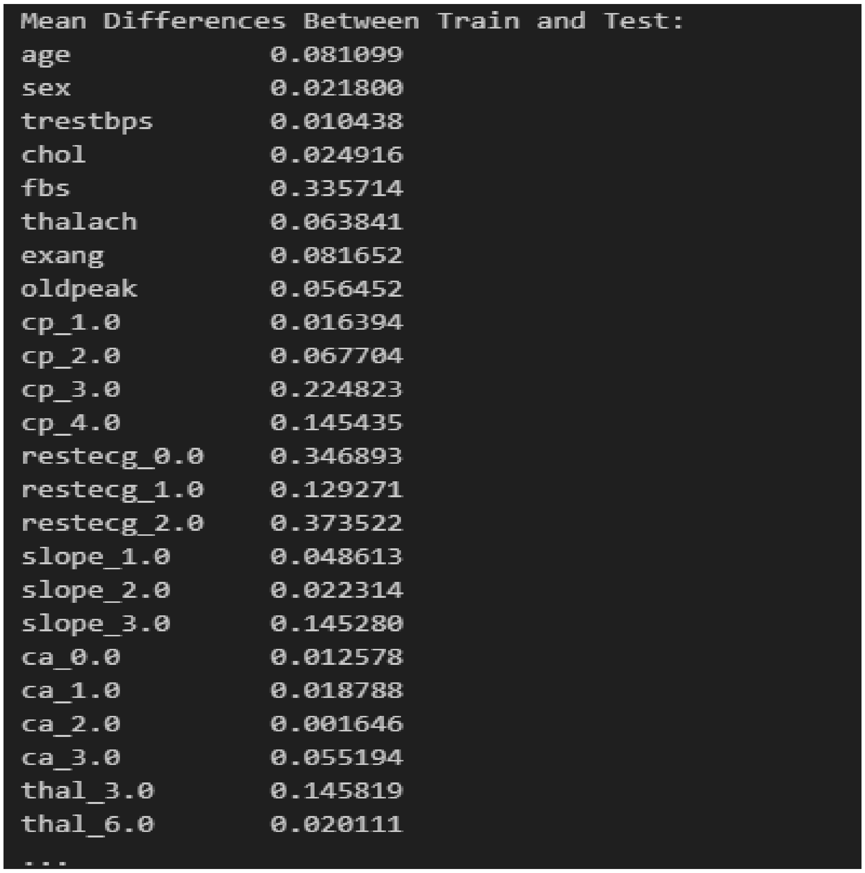
d. Feature Encoding: We applied label encoding to binary categorical variables, and one hot encoding to multiclass variables. These encodings converted categorical data into numerical format required for ML algorithms and analysis.

e. Feature Scaling: Feature Scaling was performed with StandardScaler, which standardized the datasets features.

The order of the preprocessing techniques are important, from the duplicate removal to the feature scaling, each preprocessing step needs to be applied to ensure the integrity of the data

3. Dataset Splitting and Results

80-20 Split: We split the dataset into 80% training and 20% testing, which resulted in very minor differences when we compared our training results to the testing results. We compared the Mean differences between the Train and Test (represented below).



```
Mean Differences Between Train and Test:
age             0.081099
sex             0.021800
trestbps        0.010438
chol            0.024916
fbs             0.335714
thalach         0.063841
exang           0.081652
oldpeak         0.056452
cp_1.0          0.016394
cp_2.0          0.067704
cp_3.0          0.224823
cp_4.0          0.145435
restecg_0.0     0.346893
restecg_1.0     0.129271
restecg_2.0     0.373522
slope_1.0       0.048613
slope_2.0       0.022314
slope_3.0       0.145280
ca_0.0          0.012578
ca_1.0          0.018788
ca_2.0          0.001646
ca_3.0          0.055194
thal_3.0        0.145819
thal_6.0        0.020111
...
```

4. Imbalance Dataset Handling

There were no imbalances in our dataset, this is because when we did the split labels like sex were split evenly. However in datasets where there is an imbalance, it might require resampling or tuning of the weights in the model.

5. Conclusion

In Conclusion, our preprocessing pipeline applied several techniques to prepare the data for model training, testing and evaluation (comparing the testing and the training). Even though the model is not 100% accurate, the difference between the training and testing is very narrow.

6. Contributions:

a. Dhruv Sharma: Worked on splitting the model and training it, as well as tuning the model and displaying the results of the model.

b. Richard Clinger: preface for model and script to make the dataset messy.

c. Brian Hert: Worked on the app.py by enhancing data cleaning script. Added functions for missing values and outlier handling.

d. Created inconsistent data patterns within the rows and Introduced random missing values for messy.py. Tracking and displaying modifications for initial data file.