**Heart Disease Detection: Using Ensemble Voting**

Vishal V Patel

Department of Computer Science

COMP 4600-204: Computing for Health and Medicine

Professor Wenjin Zhou

April 29, 2022

**Project Description**

My group and I chose to detect heart disease in a dataset we retrieved from the Kaggle database. While trying to search for a problem, heart disease was a common theme from my group members and therefore, we retrieved a dataset which contains one thousand samples of patient data with over thirteen unique features upon which we can evaluate. Heart disease is the leading cause of death in America because a person dies every thirty-six seconds from cardiovascular failure. Doing the research after we chose our problem led us to conclude that we can build a model from the existing dataset from Kaggle to predict whether a patient has a reversible or unreversible heart disease. The problem we want to solve is our model should be able to detect whether or not a patient has a reversible or unreversible heart disease. This detection would be able to save lives if heart disease is detected earlier on and treated.

**Dataset**

The dataset we used is from the Kaggle database and it is called “heart.csv.” Our decision of target value was changed from the original binary classification of heart disease/no heart disease to the reversible/irreversible/no disease. Below is a chart of all the columns of data that we retrieved and once we printed the dataset, there was severe imbalance in the dataset. We used the SMOTE class to manage this severe class imbalance which created pseudo data for all of the underrepresented classes in the dataset. This dataset has fourteen different attributes(13 features and one target).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Age | Sex | Cp | Trestcps | Chol | Fbs | Restecg |
| Thalach | Exang | Oldpeak | Slope | Ca | Thal | Target |

**Related work**

We referenced two different papers for this project that really guided us for most of the code that we had to write. It gave us an indication as to if the methodology we were implementing was correct or not. The first paper we used is “Using Machine Leaning to Predict Heart Disease” by Nikhil Bora and he used two different datasets to compare heart disease detection. The first dataset had the same attributes as ours and the second dataset had twelve attributes. He used random forest, logistic regression, Naïve bayes, SVM, KNN, and Extreme Gradient boosting to make his prediction. He really goes in depth as to what the features mean and explains the steps taken to understand the data and preprocess the data which helped me understand what the dataset is presenting to me. In the end, he got two different sets of results for both datasets and he concluded that the first dataset gave him a test accuracy of 92% with SVM and the second dataset gave him a high accuracy of 94% using Random Forest. All the other models performed well too but these stood out the most with the high accuracies.

In my personal opinion, the second paper we referenced did not help me much because it was comparing results to Tufts and the dataset they used was a bit different from ours. I referenced the paper by Nikhil all the way because I got a much better understanding about the dataset through that. Even till now, I took a look at the second paper at best a few times and it did have a few things that helped me but it was not anything new I discovered.

**Methodology and Results**

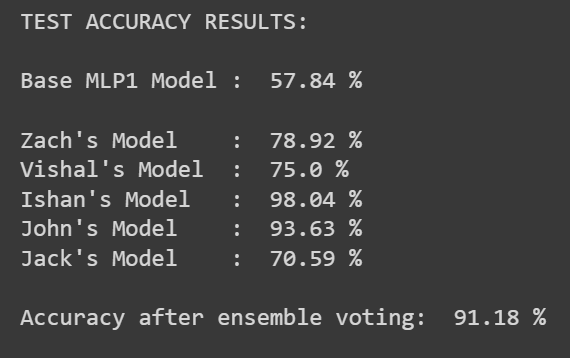
The key point in our project is we wanted to use ensemble voting which would make the overall result easier to produce. We first imported our dependencies and then printed the data. Then we went right into preprocessing the data which means we set our target, stratified our train, test, split, normalized the data, and used oversampling to manage the imbalance. We removed invalid results in the preprocessing and using stratify will allow the distribution of data be smooth in the train, test, split. We used a normalizer to normalize the data and there is a method called smote which helps with imbalanced data and it was a game changed for us when we found out it existed.

A picture containing icon

Description automatically generatedA picture containing bar chart

Description automatically generatedOnce preprocessing was done, we printed the graphs for before and after resampling the data. As you can see, before the data had major imbalance but once we resampled the data

using smote, it was managed with ease. Then we defined a function that produces train, test, and validate results for each of our models. This was done using k-fold cross validation which means the data was split evenly between five models and then it would output the results to us.

This is when we all had our own individual models that we created with tuned hyper parameters and we all made our models from a base model that was a fully connected ANN (MPL). The model was simple, define it with the mlp classifier and then fit it to xtrain and ytrain and then predict our results with the function that we made above. Then we made a helped function for the ensemble voting. This function will combine arrays into arrays of tuples which is useful for ensemble voting and it will return the array with the most common vote. The next cell is creating an ensemble array to combine all of our predictions and print it. Note, this is not making the prediction but we are just creating the array to be used in the next cell for the predictions.

The next cell will now produce an array with a vote to see the average accuracy of our models. The final cell we printed our results for all the models and what the accuracy was for ensemble voting. As you can from the results, ensemble voting gave us a 91% accuracy which is very promising for heart prediction failure.

The results are exceptionally good and I will explain what they mean. With ensemble voting, since there are three models that did not perform well and two model that did perform well, ensemble voting will be able to counteract the accuracies and balance it out. This is the key significance for our project versus the papers we referenced. We show that ensemble voting can give us a good accuracy as long as our models are performing well.

**Discussion**

A few things to discuss about the code itself is that we did not have this accuracy in the beginning. There are a few things that we found out throughout doing the project that helped us write the code efficiently and produce the results smoothly as well. The smote method we did not discover it until last week and before that we were trying to manage the imbalance a completely unique way. Zach had a keras model that did not work with all the MLP models so he had to make his own MLP model to collaborate with ours. We went back and forth to make his model work but, in the end, we could not make it work and he just made a MLP model. His own work is still within our google colab notebook because he still spent time coding it.

Our group misunderstood the target “thal” for our heart disease dataset. We were initially instructed to stay away from binary classification and we saw that “thal” was the next column over in our dataset and it mentioned reversible and nonreversible heart disease so we assumed this was our target for the entirety. While presenting in class on Tuesday April 26, a fellow classmate and professor Zhou mentioned something about not comparing datasets and how were we sure we beat the accuracy of Tufts Medical center. We were suspicious of this and upon investigating after our presentation, we found out that “thal” is a blood disease and not a heart disease. Therefore, we retract our statements of beating Tufts because we cannot compare these models and results since they are two different diseases. Overall, the problem is still the same and we did meet the requirements of classifying a disease but the interpretation of the results cannot be compared to Tufts and Nikhil Bora. Even though we did not predict heart disease detection, we still managed to predict a blood disease prediction and the disease is know as Thal.

**Teamwork**

All the work done evenly and everything was split so that one person was not doing all the work. I added some comments into the code written and made my own MLP model. I also proofread the presentations and made sure everything was the correct font and meticulously organized. Jonathan wrote the base code for the file itself and everyone including myself worked off of that. Ishan did some visualizations and his own MLP model. Jack did his own MLP model and some other comments and bug fixes. Zach did his own model as well but he had more understanding of machine learning than anyone else so he played a big part in helping us understand some of the things within our file. Everyone did their own slides in a shared google presentation and we met every couple days to go over what was due and finalized everything as a group. The final submission was finalized together and we combined all our code into one file and made sure everyone had their part done.

**Future work**

We could take a different approach to this problem by using and coding a completely different architecture for the code to achieve better results. Ensembling different network architectures and trying different sampling techniques can make this project more efficient with better results. This is in theory but these are a few things that can be done to achieve a potential higher accuracy than Nikhil Bora. We could read the data more efficiently so we do not encounter a problem where our target column was misunderstood. I will focus on this more than anyone else because not recognizing the target column was a misunderstanding.

**Conclusion**

We will not be comparing our results to either of papers mentioned in my report because we did not predict heart disease predictions but a blood disease known as “Thal.” We set out to build models to predict heart disease and we found a 91% accuracy using ensemble voting for a blood disease. I am really upset that we did not recognize the correct target column but we were able to achieve a remarkably high accuracy for the blood disease and this is an accomplishment in my books. Like I mentioned above, reading, and analyzing the data more efficiently would have been in our benefits so we can understand and chose the correct target. Ensembling different network architectures and trying different sampling techniques would allow us to distribute the data more efficiently. We could have achieved a high accuracy if this were actually for heart disease and in that case, we would have beaten Tufts dataset by 7%. This is not the case so overall; our models perform very well and ensemble voting gave us the accuracy that we were looking for.

Work cited

<https://scholarworks.calstate.edu/downloads/nc580s739?locale=en>

Fahmy AS, Rowin EJ, Manning WJ, Maron MS, Nezafat R. Machine Learning for Predicting Heart Failure Progression in Hypertrophic Cardiomyopathy. Front Cardiovasc Med. 2021 May 13;8:647857. doi: 10.3389/fcvm.2021.647857. PMID: 34055932; PMCID: PMC8155292.