Predicting Two-Year Patient Survival from Echocardiogram Data (small dataset)

Raghavendra Prasad Savada

An echocardiogram (ECHO) uses ultrasound technology to see how blood moves through the heart. It shows the shape, texture, movement of heart valves, and the size of heart. An ECHO may be performed to assess a variety of heart conditions, such as heart murmurs, damage to heart muscle in patients who have had a heart attack, and infections in the heart (Reference).

In this project, I built a classification model using echocardiogram data to predict whether a patient will survive at least two years after a heart-attack. The dataset is obtained from UCI machine learning repository. There are 110 observations and 13 variables in the dataset. Out of 110 patients, 40 patients who had heart attack are still alive. One patient survived 57 months after a heart-attack. Among the patients who are still alive, the maximum months a patient has survived is 40 months. The heart-attack is most prevalent in the age group of 60-65. The youngest person to have a heart-attack was 35 years old.

In data preprocessing step, I converted missing values from ? to np.nan, and changed the columns to appropriate datatype. Then 39 patients who are still alive but had heart attacks less than 2 years ago were removed from the dataset because we cannot determine whether they are going to survive at least 2 years or not. A target variable has been created, where patients who (has) survived at least for 24 months were labelled as 1 and others were labelled 0. The target variable is imbalanced (label 0: 19 instances and label 1: 52 instances). A dataframe of 6 important variables was created to be used as features in training (variables that directly encode target variables or meaningless variables were not included).

There were some missing values in 3 of the features, which were imputed using column mean. Although, there were some outliers in each of the features, I did not remove them as I do not know whether they represent a data error or signify an important trend. The features were scaled (standardized) as feature like age-at-heart-attack takes large values (range: 35-80) and may overshadow the effect of features like fractional-shortening that takes small values (range: 0.036-0.61).

To build a predictive model, first I split the data into train and test set (75:25%). Then I fit an imputer and scaler using training set and transform both train and test set using the same imputer and scaler. First, I trained a simple logistic regression model, which also served as a benchmark model. I tuned the hyper-parameters of this model using cross-validation technique. Since the dataset is imbalanced, I used F1 score instead of accuracy as an evaluation metric during cross-validation. I think, F1 score of both classes is important in this case, so I used f1_macro, which gives same weightage to F1 scores of both classes. The logistic regression model with tuned hyper-parameters yielded an f1_macro score of 0.49. It correctly predicted 4 out of 7 label 0s (recall 57%) and 6 out of 11 label 1s (recall 55%) in the held-out test data.

Next, to find out which algorithm works best for this problem, I tested several algorithms. I tried algorithms like naive bayes, decision tree, support vector machine, K-nearest neighbor, and ensemble methods like bagging, random forest, boosting and deep learning algorithm like MLP (Multi-Layer Perceptron). I could test several algorithms as the dataset was very small. Among them, decision tree performed the best. The decision tree model with tuned hyper-parameters yielded an f1_macro score of 0.60. It correctly predicted 4 out of 7 label 0s (recall 57%) and 7 out of 11 label 1s (recall 64%) in the held-out test data. One of the major limitations in this project is that the dataset is very small and imbalanced. I have discussed the possible ways to further improve the model performance in jupyter notebook (echocardiogram_predict_survival.ipynb). The wall-motion-index is the most influential feature followed by epss in this model as indicated by feature importance score.

Finally, I labelled the unseen data in the file echocardiogram.test using logistic regression and decision tree model I built.