Heart Disease dataset

Elena Markoska

This document contains the highlights of the experimentation with the processed.cleveland.data dataset, which targets heart disease. This document is accompanied by a Jupyter notebook containing the full account of all the experimentation.

## Data exploration and preprocessing

The dataset contains 302 rows of patients’ data spread across 13 columns describing a variety of medical measurements used as features within this dataset. The 14th column is a diagnosis, representing heart disease (values 1-4) or no heart disease (value 0).

Preprocessing of the data includes:

1. Adding column names
2. Replacing numerical values with categorical names for easier understanding.
3. Replacing missing values (represented as ?) in the dataset with the most frequently used categorical value in the respective feature.
4. A correlation analysis with the intention to perform a dimensionality reduction. However, using a heatmap it was shown that none of the features were correlated and thus none of them were removed.
5. Dummy variables were created.
6. A conversion in the diagnosis field was made describing values 1-4 -> 1 (heart disease).

## Classification model and optimization of hyperparameters

We have chosen to make a predictive model that classifies patients into categories signifying whether they have a heart disease or not. The initial train of thought considered all types of approaches, however given the size of the dataset, we opted to make initial experiments using sklearn and opted to attempt deep learning techniques in future experimentation. We tried a Random Forest (RF) classifier, Naïve Bayes, and Support Vector Machine. All further experiments were performed using RF.

A subset of the initial metrics that the RF classifier was evaluated on were**: acc: 0.738, precision: 0.758. recall: 0.738.** We use this model as our baseline model for future experiments.

Evidently, these numbers could be further improved. A potential approach for improvement is to optimize the hyperparameters of the RF classifier. A complete analysis would search through all 20 hyperparameters as implemented in RF’s source code. However, these searches take a considerable amount of time. We thus chose a subset of RF’s hyperparameters and performed a grid search to find the most suitable ones. The hyperparameters we focused on were number of decision trees, max number of features before a node is split, max number of levels in each tree, whether or not we replace sampled data points.

Naturally, the goal is also not to try and find hyperparameters which will fit the data so perfectly that it will become useless on unseen data points (overfitting), but instead to try and find the right hyperparameters while still maintain the generalising power. We're thus going to use cross validation during this grid search, which does not on its own prevent overfitting, however it does give us a very good idea about the model's performance on unseen data.

First, we used a randomized grid search, achieving the following results: **acc: 0.77, precision: 0.78, recall: 0.77**. The randomized grid search allowed us to know more specifically where we should focus our grid search. Afterwards, using a targeted grid search, we achieved the following results: **acc: 0.852, precision: 0.853, recall: 0.852**. The best obtained RF classifier was obtained using the following values for the RF hyperparameters: 'bootstrap': True, 'max\_depth': 10, 'max\_features': 'sqrt', 'n\_estimators': 100.

Once we achieved a suitable RF classifier, it was of interest to advance our understanding of the model. Thus, we focused on model interpretability using permutation importance, to try and evaluate which features contribute the most towards deciding on a diagnosis. After the model was fitted, the values of every column were shuffled to try to evaluate how much they affect the predictions. If a random shuffle on a column doesn't affect the outcome, then it's likely that specific column doesn't play a large role in the final diagnosis. A partial view of this examination is presented on Figure 1.

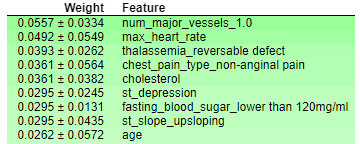


Figure 1. Features which most affect a diagnosis

Further, we wanted to understand how the model works on new data. That is, when a new patient is admitted and we intend to use our model to make a diagnosis, we’d like to know why the model predicted the outcome as it did. We thus used SHAP values (Shapley Additive explanations) to examine how the model classifies new patients. Figure 2 presents an example. The score of this specific patient is 0.78, rather close to 1, implying they very likely have a heart disease. Factors that contribute towards how high the score is are the ST depression, their thalassemia (highest value), their max heart rate of 145. Factors that help the patient are the fact that their chest pain type is a typical angina.

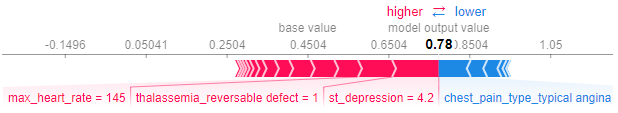


Figure 2. Model interpretability for a single patient

## Summary and Outlook

In this task, we experimented with a heart disease dataset. We chose a Random Forest classifier and focused on optimizing its hyperparameters. We used cross validation to examine the model’s performance on unseen data. We used permutation importance and SHAP values to further increase understanding of the model. It is expected that further improvement of all performance metrics will be achieved with the increase of the size of the dataset.