Heart Disease Prediction and Analysis

Project Group 25
Carmen Bentley (cnaiken)
Hasham Mukhtar (hmukhta)
Anthony Polakiewicz (apolaki)

1 Background

Heart disease is a major health problem, with many potential causes. A popular domain in ML research, automated predictive diagnosis is being applied to flag high-risk patients through analysis of symptoms and contributing factors. Hospitals and medical facilities typically have access to large sets of data regarding these symptoms and contributing factors, thus collecting and mining such datasets can be significantly beneficial to this area of research. The UCI Machine Learning Repository contains a particular dataset regarding heart disease diagnosis, with data collected in the United States, Hungary, and Switzerland.

Previous work has explored this particular domain of predictive diagnosis concerning heart disease, as well as the aforementioned dataset; however, much of the previous work only considered the data collected from Cleveland, Ohio. Various predictive machine learning models have been explored in previous work. For instance, Palaniappan and Awang were able to report an accuracy of 0.89 when using a Naive Bayes model [1]. Additionally, the work completed by Medhekar, Bote, and Deshmukh, explored the applications of Naive Bayes, Decision Tree and Artificial Neural Network models when predicting heart disease, reporting accuracy scores of 0.86, 0.89 and 0.85 respectively for the models [2].

The work mentioned above does not explicitly discuss any hyperparameter exploration for the models. Therefore, to increase accuracy, we implemented hyperparameter tuning, as well as, a more accurate mean estimation for missing values. Additionally, as previous works focused only on data from Ohio, we extended analysis to all locations included in the UCI Heart Disease dataset.

Research efforts in identifying best prediction models work to aid doctors in detecting patients at risk of heart disease for respective locations. Results aim to trigger next steps in preventative care and heighten awareness of known and newly identified contributing factors which could lead to a positive diagnosis.

2 Methods

In approaching the problem statement, we constructed predictive models based on the following four classifiers: (1)Decision Trees (2)SVM (3)Logistic Regression and (4)XGBoost. Each model was applied to all locations and performance was then analyzed. In comparing the results, we found the most appropriate classifier for each location based on prediction accuracy. Additionally, we compared surface level statistics between datasets to identify factors such as which gender and age ranges exhibit higher percentages of heart disease.

2.1 Prediction Techniques

Further explanation of applied classifiers can be found below.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

2.1.1 Decision Trees

Decision trees are a predictive model widely used in statistics, data mining and machine learning. This model incorporates a white-box approach taking in observations about an object and producing its target value or classification. Decision trees are simple structures for classifying data building upon partitioning rules and impurity metrics. Although decision trees can be sensitive to changes in data, it requires little data preparation, contains a form of built in feature selection placing more decisive features toward the top of the tree, and are simple to understand or interpret. For these reasons, we believe it is an appropriate model for comparison.

2.1.2 SVM

Support vector machines (SVMs) have been commonly used for many machine learning tasks. The primary task of an SVM is to find a decision boundary which correctly separates the dataset into different classes. Additionally, the decision boundary should maximize the separation or margin between the classes. SVMs are quite adept when classifying data that is linearly separable. For data that is non-linearly separable, a kernel transformation is often employed, mapping the original dataset to a kernel space that is linearly separable. There are several kernel functions typically used, such as a Polynomial or a Radial Basis Function (RBF). Considering that the task is binary classification, and all of the features are numerically represented, we believe SVM to be an appropriate model[5].

2.1.3 Logistic Regression

Logistic regressions are a predictive model which can be thought of as an extension of the linear regression model. Logistic regressions are often used for categorical classification, and are often considered for binary classification tasks. The logistic regression model is similar to a linear regression model; however, parameters are estimated using maximum likelihood. Thus, the output of the logistic function is probabilistic prediction about the class of the input data. Logistic regression models can be applied to datasets which include any of the nominal, ordinal, interval, or ratio data types, and thus was considered an appropriate model for anaysis[4].

2.1.4 XGBoost

Extreme Gradient Boosting is a recent ensemble machine learning algorithm and library that enhances speed and performance. Using a gradient boosting framework, this method is similar to adaboost in that it implements an ensemble of decision trees to predict target labels. Taking advantage of algorithmic enhancements such as LASSO and Ridge penalty regularization to prevent overfitting, increased sparsity awareness and iterative cross-validation, this method reduces training time and increases prediction accuracy. Additionally, the open-source XGBoost libraries offer significant hardware/software adjustments that aid in yielding results using less computational resources in the shortest amount of time. We chose to implement this method as research confirms it is good for small to medium datasets with a tabular structure [3].

2.2 Training and Testing Sets

We broke up each dataset into training and testing data by using the Holdout method with an 80% training and 20% testing split. In combination with this method, we implemented stratified sampling to alleviate any biases or unbalance in the datasets and further enhance our results.

2.3 Summary Statistics

To observe surface level data across the datasets, we looked at what gender and age ranges were most affected by heart disease. Figures and further analysis of these can be found in the Results section of this report that help visualize these statistics.

2.4 Hyperparameter Tuning

In contrast to previous works, we tuned hyperparameters using Grid Search in addition to choosing the best parameters based on evaluation of multiple value combinations. This helped to produce better results for each classifier. Specific hyperparamters can be viewed in section 3.3.5 of this report.

3 Plan & Experiment

3.1 Dataset

The UCI Heart Disease data contains four sets based on statistics gathered from Ohio, Virginia, Hungary, and Switzerland. This data can be found in the UC Irvine Machine Learning Repository and accessible via the link below, navigating to the data folder link at top of the page to download.

`https://archive.ics.uci.edu/ml/datasets/Heart+Disease`

Sample size varied per location with the Cleveland site having 303 instances, Long Beach 200, Hungary 294 and Switzerland 123. Each set consisted of 76 attributes ranging from categorical, binary, integer, and real values. Although 76 attributes were gathered, only 14 were used during analysis. The attribute reduction was based on previous publications and their work in filtering irrelevant attributes. This subset contains information about a patient's age, sex, blood pressure, blood vessel anatomy and variation in heart rate and can be seen in Table 1: Features of datasets.

Table 1: Features of datasets

Features	
Name	Description
Age	Age of patient
Sex	Gender of patient (0: female, 1: male)
CP	Chest Pain Type - 1: typical angina, 2: atypical angina, 3: non-anginal pain, 4: asymptomatic
Trestbps	Resting blood pressure in mmHg (unit)
Chol	Serum cholestoral in mg/dl (unit)
Fbs	Fasting blood sugar > 120 mg/dl (unit) (0: false, 1: true)
RestEcg	Resting electrocardiographic - 0: normal, 1: ST-T wave abnormality, 2: left ventricular hypertrophy
Thalach	Maximum heart rate achieved
Exang	Exercise induced angina (0: no, 1: yes)
Oldpeak	ST depression induced by exercise relative to rest
Slope	The slope of the peak exercise ST segment
Ca	Number of major vessels: (0 - 3) colored by flourosopy
Thal	Thalassemia - 3: normal, 6: fixed defect, 7: reversible defect
Num	Diagnosis of heart disease - 0: no heart disease & 1,2,3,4: heart disease

3.2 Hypotheses & Questions

The following are questions we wanted to answer with our experiment:

- What prediction model would most help doctors predict heart disease in each location?
- Does tuning hyperparameters create better test results than using default parameters in a classifier?
- How do the locations compare in terms of how heart disease affect age and gender?

The following are hypotheses we had:

- Based on our research, we believed XGBoost will give the most accurate results.
- Given the way we handled missing values, we believed that the accuracies of all our models will be higher than those we have seen in previous works.

3.3 Experimental Design

3.3.1 Software Used

Language:	Python, version 3.
Python packages:	pandas, matplotlib, seaborn, xgboost which has import XGBClassifier, and sklearn which contains imports StandardScalar, train_ test_ split, DecisionTreeClassifier, Logistic Regression, SVC, metric and GridSearchCV.

3.3.2 Summary Statistics

Binary values from the features 'sex' and 'num' were mapped to properly display the data in graphs. For 'sex', 0 and 1 values were mapped to 'female' and 'male' respectively. For 'num', 0 values were mapped to 0 indicating no heart disease while all other values: 1,2,3,4 were mapped to 1 indicating heart disease. From these new mappings, visualizations using the matplotlib and seaborn python packages were created to find the statistics mentioned in the Methods section of this report.

3.3.3 Data Preprocessing

After summary statistics were gathered, 'sex' records were reverted back to their original binary values. Next, we handled missing values which is described in the Missing Values section below. The data was then split into training and test data using the python train_ test_ split package. When stratified sampling was applied in our tests, we would set the stratify parameter in this package to the y feature of our data. Then we standardized the data with the StandardScaler package. Additionally, we created lists of hyperparameters to use for each classifier package. The list of hyperparameters and their names are in the hyperparameter section below.

3.3.4 Handling Missing Values

To handle missing data, we chose to produce subset estimations. Subsets were first divided into groups based on their sex, male or female, further clustered based on age ranges [20-29],[30-39],[40-49],[50-59],[60-69],[70-79]. The amount of missing data varied. In sets for Virginia, Switzerland, and Hungary it was necessary to remove fields Ca and Thal as more than seventy percent of there entries were missing. In the sets belonging to Europe, fields missing more than forty percent were also removed. The remainder of missing data was replaced with the subset mean value for continuous values and the majority value for discrete values.

3.3.5 Hyperparameters Used

Table 2 shows all the hyperparameters we used for each classifier.

Table 2: Hyperparameter List

Hyperparameters			
SVM	Decision Tree	Logistic Regression	XGBoost
C = { 0.1, 0.5, 1, 5, 10 } kernel = { 'linear', 'poly', 'rbf','sigmoid' } gamma = { 'auto', 0.0001, 0.001, 0.01, 1 } degree = { 1, 2, 3, 4, 5 } coef0 = { 0, 0.0001, 0.001, 0.01, 1 }	criterion = { 'gini', 'entropy'}	C = { 0.1, 0.5, 1, 5, 10 } penalty = { 'l1', 'l2' }	learning_ rate = { 0.05, 0.01, 0.3, 0.5 } max_ depth = { 3, 5, 7, 10 }

3.3.6 Data Processing

For each dataset, we applied the classifiers discussed in the Methods section. To tune the hyperparameters, we used the GridSearchCV package passing the model and a list of corresponding parameters. GridsearchCV can also take in different cv values, changing the number of folds used when testing all the hyperparameters. For each model we found the accuracy, recall, precision, and f-measure values using the metric python package. Finally, we found the optimal hyperparameters for each model by using GridSearchCV's best_ params_ method.

Using these methods, we created different combinations of each prediction model using the following setups:

- no hyperparameter tuning (no GridSearchCV) with and without stratified sampling
- hyperparameter tuning using cv value 3 with and without stratified sampling
- hyperparameter tuning using cv value 5 with and without stratified sampling

By comparing each version of the prediction models, the best combination of parameters for each location was determined.

3.3.7 Challenges

Existence of missing values posed a prominent challenge. Data for both Hungary and Switzerland contained an abundance of missing data, and for two features, the majority of entries were missing. For this reason, features missing data for more that 40% of data instances were removed in order to avoid any skewing.

Data gathered from Switzerland was very unbalanced. It had the least number of records, of which approximately 90% showed true for heart disease. When the models were evaluated on this set, extremely high accuracy resulted. Because of these issues, this set was removed from our experiment.

4 Results

4.1 Summary Statistics

Graphs showing summary statistics of Cleveland, Long Beach and Hungary datasets can be seen in Figures 1, 2 and 3 respectively. Orange indicates heart disease and blue indicates no heart disease. The chart on the left of each figure represents a comparison of patients with heart disease based on age while the chart on the right shows a comparison based on sex.

Figure 1: Cleveland Statistics

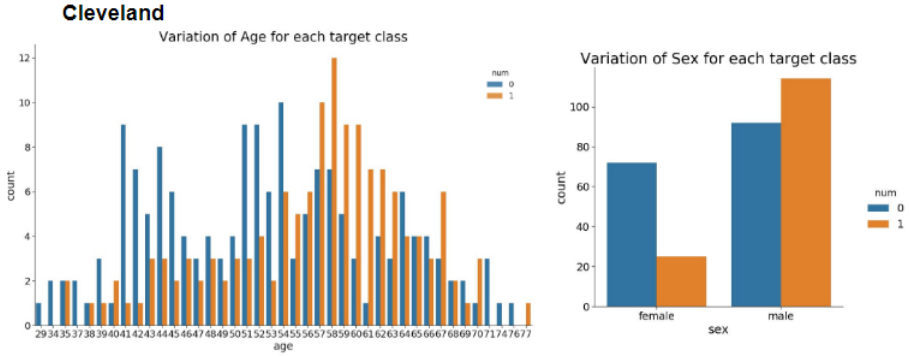
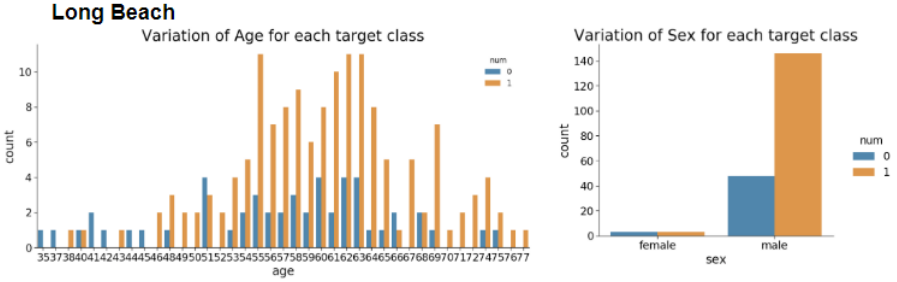


Figure 2: Long Beach Statistics



4.2 Cleveland

The best prediction model for this location was XGBoost as it had the highest accuracy compared to the other models. These results came from the combination of using hyperparameter tuning with cv value 3 and no stratified sampling. The optimal hyperparameters for this model and the results of the rest of the models can be seen in Table 3.

Figure 3: Hungary Statistics

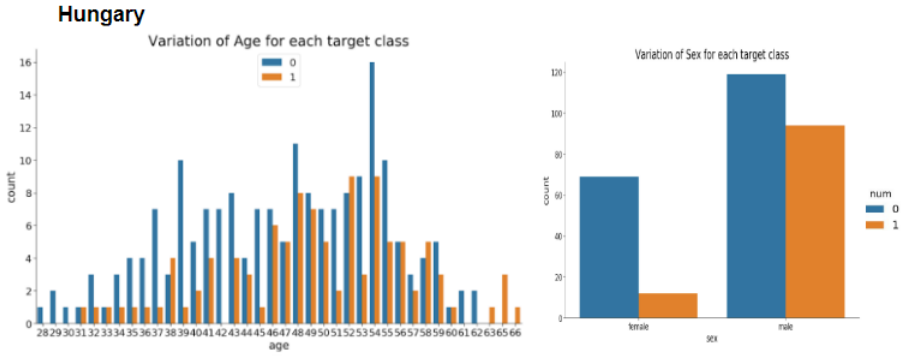


Table 3: Cleveland Data Results

Prediction Technique Comparison				
Measure	SVM	Logistic Regression	Decision Classifier	XGBoost
Accuracy	0.86885	0.85246	0.73771	0.88525
Recall	0.84615	0.80769	0.76923	0.80769
Precision	0.84615	0.84	0.66667	0.91304
F - Measure	0.84615	0.82353	0.71429	0.85714
Optimal Parameters	C = 1 coef = 0.01 degree = 3 gamma = 'auto' kernel = 'poly'	C = 0.5 penalty = 'l2'	criterion = 'entropy'	learning_ rate = 0.05 max_ depth = 3

4.3 Long Beach

The best prediction model, Logistic Regression, used hyperparameter tuning using cv value 5 without stratified sampling. The results of each classifier and hyperparameter for the Long Beach dataset can be seen below in Table 4. Because each of the classifiers produced an accuracy of 0.7, it was necessary determine the best model based on recall. Considering our goal to label patients with heart disease, evaluation on high recall provides the model with the highest rate of true positives.

Table 4: Long Beach Data Results

Prediction Technique Comparison				
Measure	SVM	Logistic Regression	Decision Classifier	XGBoost
Accuracy	0.7	0.7	0.7	0.7
Recall	0.1	1.0	0.89286	0.92857
Precision	0.7	0.7	0.73529	0.72222
F - Measure	0.82353	0.82353	0.80645	0.8125
Optimal Parameters	C = 0.1 coef = 1 degree = 1 gamma = 1 kernel = 'sigmoid'	C = 0.1 penalty = 'l1'	criterion = 'gini'	learning_ rate = 0.01 max_ depth = 7

4.4 Hungary

The best prediction model for the Hungary dataset was SVM. The hyperparameters for the model were tuned using 3-fold cross validation and using stratified sampling. The results for each classifier and the optimal hyperparameters for the Hungary dataset can be seen below in Table 5.

Table 5: Hungary Data Results

Prediction Technique Comparison				
Measure	SVM	Logistic Regression	Decision Classifier	XGBoost
Accuracy	0.91525	0.89831	0.81356	0.84746
Recall	0.90476	0.85714	0.71429	0.71429
Precision	0.86363	0.85714	0.75	0.83333
F - Measure	0.88372	0.85714	0.73170	0.77
Optimal Parameters	C = 0.5 coef = 1 degree = 1 gamma = 'auto' kernel = 'sigmoid'	C = 0.1 penalty = 'l2'	criterion = 'entropy'	learning_ rate = 0.05 max_ depth = 3

4.5 Critical Evaluation

Our biggest question for our experiment was which model best predicts heart disease in each location. From our results, we found that Logistic Regression was the best prediction model for Long Beach, XGBoost was the best prediction model for Cleveland, and SVM was the best prediction model for Hungary. In order to determine if tuning hyperparameters would give better test results, we used a grid search method for determining optimal parameters compared to default models. The results for each dataset used hyperparameters to get their best prediction model, suggesting that the hyperparameter tuning successfully optimized the models. The results also suggested that by handling missing values using subset mean clustering, based on age range, prediction accuracy values were comparable to previous work regarding the Cleveland dataset. Also, due to the method of handling missing data, predictive models were able to be built for the Hungary and Long Beach datasets, which were missing a fair amount of data initially. Models predicting on the Hungary dataset performed well relatively speaking; however, it is clear that the Long Beach dataset may require further exploration.

Initially we hypothesized that the XGBoost model would perform with the best prediction accuracy metrics. However, this was only true for the Cleveland dataset. One possible answer for this is that the Long Beach set is smaller than the Cleveland one. While XGBoost is supposed to work better for smaller datasets, maybe the Long Beach dataset was too small for it to perform well. Additionally, the Long Beach and Hungary datasets were missing more data initially than the Cleveland set, which could have affected performance.

In terms of our summary statistics, we wanted to see how the locations compared. We noticed that both American locations followed a similar pattern in that heart disease is affecting older people in their mid 50's to early 60's. The Hungary data has a slightly smaller age range that is affected; mid 40's to mid 50's. This is probably because the dataset is much smaller than the other two and the people who were recorded may have just been younger in age. Also, while it can be seen that all locations had the same pattern in that males accumulated more instances afflicted by heart disease than females, the datasets were significantly unbalanced in regard to gender. This was why age was the better option for imputing missing data.

5 Conclusion

5.1 Lessons Learned

Throughout this project, we encountered many obstacles which required pause. Considering the wealth of data available, it was difficult to select a well structured set that could offer both interesting and relevant results all members were excited to explore. Many sets could only provide partial

answers to proposed questions, while others were malformed. The heart disease set, although small in comparison, offered reliable and previously analyzed data.

Initially, we had anticipated applying the same set of parameters for model evaluation on each dataset. After discussion amongst the group and with course teaching assists, we realized that, since there was no need for cross comparison over locations, it was possible to apply different parameter mixtures to each set. This gave us the opportunity to observe how various combinations of tuned hyperparameters, sampling and validation techniques could improve upon model accuracy.

While evaluating models for each location, two of the three models were easily selected using accuracy. Unfortunately, the Long Beach set showed equal accuracy across all models. It was unclear at first how to determine which was the best fit. After research into selecting a model that would yield best results given the context of medical diagnosis, it was determined that using recall as a source of comparison would determine the model which returned the highest rate of true positive cases. Here we learned that the problem circumstances should be considered when deciding ones approach for model evaluation.

As a general observation about the problem itself, it was our assumption that as locations vary, especially across countries, trends would vary accordingly. For example, American patients would show a higher rate of heart disease over patients in Hungary. However, this was not the case and heart disease showed the same general tendencies over all sites. This shows the power in data analysis and its ability to discover patterns to possibly support or disprove misconceptions.

5.2 Future Work

Data augmentation is suggested for any future work with this experiment. This can be done by continued data collection for each site, combining similar datasets for models analysis, or apply techniques for data augmentation thus increasing records in each set.

Only four classifiers were applied in this experiment, others could be evaluated. Although previous works surveyed Naive Bayes and Neural Networks, hyperparameter tuning and enhanced handling of missing values was not applied. Therefore, future observations should combine these classifiers/techniques with our adjustments as the previous results could be improved.

Also, due to time constraints, we were unable to discover which of the 14 features most contributed to heart disease. Future work can be done to determine what these features are and possibly do an analysis of any anomalous data from each dataset.

6 References

[1] Palaniappan, Sellappan, and Awang, Rafiah. "Intelligent Heart Disease Prediction System Using Data Mining Techniques." Conference: IEEE/ACS International Conference on Computer Systems and Applications, March. 2008, https://ieeexplore.ieee.org/abstract/document/4493524.

[2] Medhekar, D., Bote, M., and Deshmukh, A. "Heart Disease Predictcion System using Naive Bayes." International Journal of Enhanced Research in Science Technology and Engineering, March. 2013, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.378.9860&rep=rep1&type=pdf

[3] Chen, Tiangi, and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System - Researchgate.net." Research Gate, Conference: the 22nd ACM SIGKDD International Conference, Aug. 2016, https://www.researchgate.net/publication/310824798_XGBoost_A_Scalable_Tree_Boosting_System.

[4] Bewick, V., Cheek, L. and Ball, J. "Statistics review 14: Logistic regression." Crit Care 9, 112 (2005) doi:10.1186/cc3045 https://ccforum.biomedcentral.com/articles/10.1186/cc3045#citeas.

[5] Huang, Shujun et al. "Applications of Support Vector Machine (SVM) Learning in Cancer Genomics." Cancer genomics and proteomics vol. 15,1 (2018): 41-51. doi:10.21873/cgp.20063 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5822181/.

7 Appendix

7.1 Project Repository

The following link provides access to the project repository including source code and datasets.

https://github.ncsu.edu/cnaiken/CSC_522_Final_Project

7.2 Teammate Division

Carmen Bentley:

Applied all techniques mentioned in Method section to Long Beach dataset. Also handled missing values in all datasets.

Hasham Mukhtar:

Applied all techniques mentioned in Method section to Cleveland dataset.

Anthony Polakiewicz:

Applied all techniques mentioned in Method section to Switzerland and Hungary datasets.

All:

All members worked on report. Each member assigned sections to work on. Added any findings, graphs, and figures related to their work to appropriate sections.

7.3 Revisions

- Added to future work - Finding most important features that contributes to heart disease. Referenced previous publication to identify irrelevant features to be removed from each dataset before analysis.
- Removed analysis of anomalies and focused time on finding most appropriate combination of parameters to enhance prediction results for each dataset location.