Heart Disease Diagnosis on Medical Data Using Ensemble Learning

BOYANG DUN, ERIC WANG, AND SAGNIK MAJUMDER, Stanford University

We investigate the relevance of medical and demographic information in predicting the presence of heart disease in an individual. We apply a variety of ensemble and deep learning techniques with hyperparameter tuning and feature selection, resulting in a maximum test accuracy of 78%. Model averaging does not significantly improve prediction accuracy, and the same points tend to be misclassified by all the models that don't overfit. This suggests that the errors in our data can mainly be attributed to irreducible error in the problem.

1 INTRODUCTION

Around 1 in every 4 deaths in the United States can be attributed to cardiovascular disease, the likelihood of which only increases with age. Such an occurrence, although highly common, can be mitigated with intelligent forecasts and diagnoses based on related medical data.

The causes and symptoms of heart disease are well known—for instance, medical wisdom holds that diabetic patients[6] and smokers [3] are at greater risk for heart disease—but it is less clear how these causes compare and interact with one another. Therefore, without conducting a blood test, it is difficult for an individual to determine whether they have any kind of heart disease.

This is a significant problem, especially for those without access to affordable medical care. While measures like the Affordable Care Act have somewhat ameliorated this problem, the fact remains that many patients are discouraged from preventative care due to the prohibitively high costs. [5]

This difference may contribute significantly to the strong correlation of life expectancy and income. If these patients had an automated system that could easily and reliably detect heart disease without the expensive equipment needed in the hospital, this would make an enormous contribution to public health.

Our project seeks to understand how different readily measurable features of hospital patients can be used to diagnose heart disease without requiring invasive treatments or the judgment of a medical professional.

We develop an algorithm that can diagnose whether a patient has heart disease with reasonable accuracy given only a readily measurable set of features. Such a program would be enormously beneficial to people without access to easy and affordable medical care. To this end, we compiled a dataset of 899 anonymized patients from three hospitals and transformed it accordingly. We then ran it through a variety of learning models in order to better understand its features and their interactions.

2 RELATED WORK

Extensive research has gone into the problem of predicting the occurrence of heart disease before it happens and understanding the relationship between heart disease and a variety of other health conditions and features. In particular, there is a large body of research suggesting a strong relationship between diabetes and heart disease. However, these studies lack a unified approach to the topic, and we could find few that explored how all of these factors interacted with one another. Other work into the prediction of heart disease tended to use features that weren't part of our model, like insurance filings and blood tests. In general, the field of heart disease diagnosis (for any type of disease) has been much less studied than that of straightforward heart disease prediction, possibly owing to the direct application of latter in insurance bookkeeping.

3 METHODS

3.1 Preprocessing

We compiled a dataset of 899 anonymous patients from three hospitals: Cleveland Clinic, University Hospital of Switzerland, and the Hungarian Institute of Technology. These patients were all suspected of having heart disease, and a little less than 60% of them actually did. These patients were each tested for heart disease via an electrocardiogram and an exercise test, as well as more advanced procedures like X-ray fluoroscopy and blood tests.

We removed X-ray fluoroscopy and blood test information from our data, leaving only features that can be measured without the assistance of a medical professional. We further augmented our data with aggregate metrics from a government study that detailed heart disease frequencies between different age buckets and the two sexes [1]. Following data preprocessing, we implemented four main model types - random forests, logistic regression, support vector machines and neural networks.

3.2 Models

3.2.1 Random Forest. A random forest implements a cluster of decision trees. From a single example, each decision tree outputs a prediction that is then aggregated by majority

vote into the final prediction from the forest. Due to its ability to handle both continuous and categorical variables and its tendency to not overfit the training data, we chose it as a candidate algorithm

3.2.2 Logistic Regression. A logistic regression model tries to find the best hyperplane (if the data has n features, the hyperplane has $n-1$ dimensions) to separate the data along. It attempts to learn the hyperplane by minimizing a cost function. For classification problems like ours, it is standard to use the cross-entropy loss, given by

$$CE(y, \hat{y}) = -\sum_{i=1}^{n} y^{(i)} \log \hat{y}^{(i)},$$

where $y^{(i)}$ is the true label for training example i and $\hat{y}$ is the the probability vector for predictions. The probabilities of each class are generated by applying an activation function on a linear combination of the inputs, and we stick with commonly used sigmoid function, given by

$$\sigma(z) = \frac{1}{1+e^{-z}} \implies P(y=1|x) = \frac{1}{1+e^{-\theta^T x}}.$$

3.2.3 Support Vector Machine. Support vector machines (SVMs) are second planar binary classification models that attempt to maximize the minimum distance of either class from the separating hyperplane. SVMs are non-probabilistic classifiers and simply output the label of the class. They use the hinge loss which is a literal measure of the number of misclassified examples. While SVMs normally only find a hyperplane through the data, they can be used to learn complex non-linearities using the Kernel trick - a method in which the input features are replaced with an implicit mapping into a higher dimensional space. We experimented with two different kernels - the regular linear SVM and the Gaussian radial bias function (RBF) kernel given by

$$K(x, y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right),$$

where σ is a tunable parameter.

3.2.4 Neural Network. Finally, a standard feed-forward neural network captures non-linearities in the data that a logistic regression model fails to. Neural nets are loosely inspired by biological brains; the learning architecture is a series of layers of neurons, where each neuron gets inputs from the previous layer and tries to learn some relevant feature. A neuron is activated by certain linear combinations of the inputs that it receives and it is precisely these wights that it learns. Essentially, a neural net performs several logistic regressions to learn features and uses non-linear activation functions to learn more complex functions in deeper layers. Common choices of activation functions include the sigmoid function, the hyperbolic tangent and the rectified linear unit,

$$\text{ReLU}(x) = \begin{cases} 1 & x \geq 0 \\ 0 & \text{otherwise} \end{cases}.$$

We decided to apply ReLU activation to the hidden layers and sigmoid to the output layer, as is standard practice.

We attempted both L1 and L2 regularization with the neural net. We also used a batch normalization layer, which normalizes the output feature vector of the previous by subtracting the mean and dividing by standard deviation. All of the layers were initialized using Xavier/Glorot initialization in distributions with a standard deviation of a small multiple of

$$\sqrt{\frac{1}{n_{\text{in}} + n_{\text{out}}}} \quad \text{or} \quad \sqrt{\frac{1}{n_{\text{in}}}}$$

where $n_{\text{in}}, n_{\text{out}}$ denote the numbers of input and output layer neurons respectively. We believed that a neural network would outperform logistic regression by capturing non-linear features of the dataset.

4 EXPERIMENTS AND RESULTS

We divided our data along a stratified 4:1 train-test split and performed 5-fold cross validation on our training data for feature selection and hyperparameter tuning. We chose to use 5-fold validation mainly due to resource constraints – some functions, particularly influence measures and feature selection, ran too long for greater values of k.

4.1 Random Forest

The random forest we fit to our dataset performed comparably to our other models.

Model	Train	Dev	Test
Random Forest	84.1	83.0	77.2

Table 1. Random forest model performance

We tuned our model iteratively with hyperparameter selection and a backwards feature search until our dev set accuracy converged. Hyperparameters we tuned include number of trees, maximum height of trees, and minimum number of samples required to be at a leaf node. The most predictive features of the random forest, determined by their average decrease in node impurity, are shown in Figure 1. These include:

- Chest pain
- Max heart rate
- Cholesterol level
- Age
- Sex

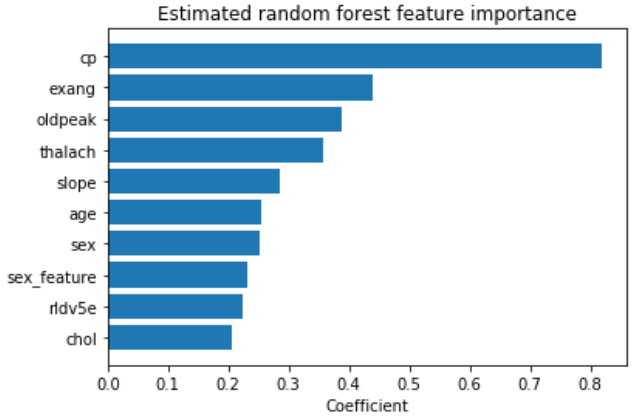

Fig. 1. Random forest feature importances

Fig. 2. Cook's distance, a measure of each point's influence on our parameter estimates. Note the highly influential points at the end.

4.2 Logistic Regression

Logistic regression performs surprisingly well on our data, providing strong evidence that heart disease conditions are linearly separable up to noise.

We used a backward feature search with permutation accuracy tests as our selection criterion. More concretely, we started with a regression model using all the features, then iteratively removed the features that contributed the least to our accuracy until the effect of that feature was statistically significant. The permutation test of statistical significance consisted of randomly permuting the entries of the relevant feature column 100 times, running a regression on each, and counting how many permutations added more accuracy than the feature in question. This allowed us to evaluate how useful each coefficient was without making any assumptions about the underlying data.

We also found that L_2 regularization tended to outperform L_1 regularization in terms of validation error. Using a simple linear search, we established that the optimal λ for L_2 regularization was $\lambda = 7.692$. We considered elastic-net regularization but ultimately did not have the computational resources to attempt it.

With the resulting model, we used Cook's distance estimate to measure the influence of each point on the model. The formula for Cook's distance is

$$D_i = \frac{||\hat{z}_j - \hat{z}_{j(i)}||^2}{ps^2}$$

where $\hat{z}_{j(i)}$ is the value of $X\theta$ when θ is estimated using logistic regression without example i, and $\hat{z}_j$ is the same thing but with example i included in the regression. p and s represent the number of parameters and the mean squared error, respectively.

The distribution of Cook's D along our training set can be visualized in the accompanying figure. We found that removing the two most influential points from our training set increased our validation accuracy by a small but significant amount.

With all of these improvements, we found that we were able to achieve a test accuracy of 75%, a significant increase over a naive logistic regression on all the parameters (which only achieved an accuracy of 65%). Our most important features were age, reported history of heart problems, and reported chest pain.

Model	Train	Dev	Test
Logistic Regression	80.0	78.9	75.0

Table 2. Logistic regression model performance

4.3 Support Vector Machines

We decided to build on our work in logistic regression by running a linear SVM on the same features. Again, we tuned our regularization parameter and found that the optimal validation accuracy came from L_2 regularization with an approximate $\lambda = 1200$. (As linear SVMs tended to converge much slower than logistic regression, time constraints prohibited us from optimizing λ further.) The result was largely the same, however; linear SVMs ended up with the same classification boundary and even misclassified the exact same points on the test set.

We then tried running an SVM with radial basis functions and all the features to test if there was nonlinearity in the heart disease prediction boundary. Despite additional hyperparameter tuning and aggressive regularization, we were

Model	Train	Dev	Test
Linear SVM	80.3	78.9	75.0

Table 3. Linear SVM performance on the data.

unable to match the validation performance of the fully linear models.

Model	Train	Dev	Test
RBF SVM	73.7	71.9	68.9

Table 4. RBF SVM model performances

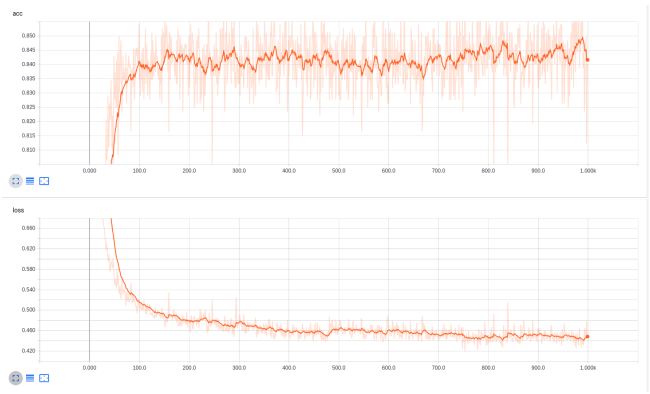

Fig. 3. NN training accuracy and loss against number of epochs

4.4 Neural Network

Finally, we applied a densely connected 3 hidden layer neural network. Unregularized, the neural net grossly overfit the training data, so we tuned the regularization parameters. L2 regularization by itself proved to be insufficient and didn't give us a better result than logistic regression. So, we tuned both the L1 and the L2 regularization parameters and settled for $\lambda_1 = 0.003$ and $\lambda_2 = 0.01$.

We used the Adam optimizer, which computes an exponential moving average of the gradients and their squares, with the decay rate weighted by the default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and updates parameters using bias-corrected gradient measures. We tuned the learning rate α and decided that the default value of 0.001 is what performed best on the validation set.

The Adam optimizer is known to do well with noisy data and data with sparse gradients[4]. We knew that this was the case with our data since the even with regularization, the neural net did only marginally better than logistic regression, so most of our newer gradient must have been zero anyway. This motivated us to also add a batch normalization layer, to minimize covariate shift within the dataset. Adding normalization right after the first layer maximized the accuracy on the development set.

Lastly, we used several variants of the Xavier/Glorot initialization [2] for the weights on each of the layers. This again results in slightly better performance that the standard normal distribution. Xavier initialization makes sure that none of the weights in the network are too small or too large so as to prevent vanishing/exploding gradient issues.

Model	Train	Dev	Test
Neural Net	84.7	83.3	78.3

Table 5. Neural Network performance

We trained our model for a 1000 epochs on all 5 folds and the training set, with an early stopping wait of 200 epochs. The model trains very quickly, usually converging within 100 epochs to close to the best weights. The performance on the test set was still only marginally better than our other models.

4.5 Error Analysis

The RBF SVM model we applied to our data performs significantly worse than our linear SVM, logistic regression, random forest, and neural network models, which suggests that our data is relatively linearly separable. This claim is also supported by the fact that our logistic regression model performs very similarly to our neural network. If the dataset contained major non-linear patterns relative to the presence of heart disease,the neural network model should have performed significantly better.

On the other hand, the fact that our models performed similarly even after iterative hyperparameter tuning and feature selection suggests that either the dataset features inherently aren't that predictive or there exists insufficient data currently to train the models to their optimal performances.

From confusion matrices generated by our models (Figure 3), we see that nearly all diseased patients are classified correctly, but more healthy patients tended to be classified incorrectly. The skew increases as the performance of the model type decreases.

Further, 84% of misclassified examples were shared between the top 4 performing models – random forest, linear SVM, logistic regression, and neural network. By manually investigating these common datapoints, we discovered that there was little distinguishing their features from the correctly classified points—confirming our hypothesis that irreducible error is responsible for the variation and that we would need to collect more features to improve our model.

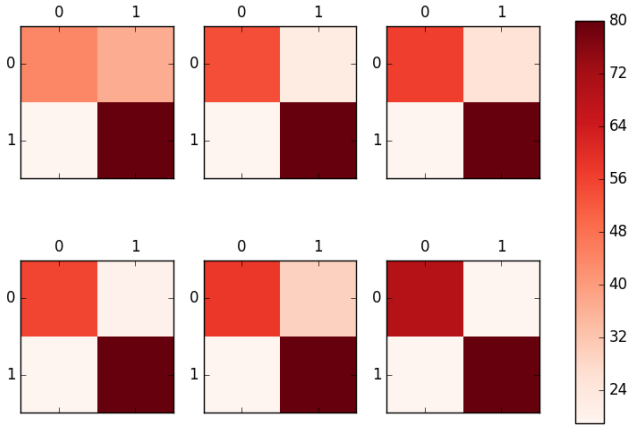

Fig. 4. Confusion matrices in row 1 from left to right correspond to the RBF SVM, linear SVM, and random forest models. Row 2 corresponds to the logistic regression, neural network, and "perfect" classification matrices. 0 corresponds to healthy patients and 1 corresponds to patients with heart disease.

5 CONCLUSION

We reached a peak prediction accuracy of almost 80%, which means significant work still needs to be done in order to reach a performance comparable to that of a medical professional. However, we believe that our initial findings demonstrate the possibility of reaching that level of accuracy, especially by making use of features determined to be most predictive of heart disease within our dataset.

6 FUTURE WORK

Based on our findings, we hypothesize that increased training data will help us reduce the misclassification rate, so we plan on collecting additional, high-quality patient data from diverse sources. We also hope to extend our work to include not only diagnosing heart disease, but also predicting its likelihood of onset in a given period of time in the future. We plan on testing the most predictive features we discovered in our current project to see how well they fare in predicting future heart disease.

7 CONTRIBUTIONS

Boyang Dun - Random forest implementation
Eric Wang - SVM and logistic regression implementation
Sagnik Majumder - Neural network implementation
Joint error analysis, poster creation, and final write-up

REFERENCES

[1] 2015. National Health Interview Survey. (2015). Retrieved Dec 7, 2017 from https://www.cdc.gov/nchs/nhis/nhis_2015_data_release.htm

[2] Y Bengio and X Glorot. 2010. Understanding the difficulty of training deep feed forward neural networks. (01 2010), 249–256.

[3] Gordon M. Burke, Michael Genuardi, Heather Shappell, Ralph B. D'Agostino, and Jared W. Magnani. 2017. Temporal Associations Between Smoking and Cardiovascular Disease, 1971 to 2006 (from the Framingham Heart Study). *The American Journal of Cardiology* 120, 10 (2017), 1787 – 1791. DOI: http://dx.doi.org/https://doi.org/10.1016/j.amjcard.2017.07.087

[4] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). http://arxiv.org/abs/1412.6980

[5] Zosia Kmietowicz. 2008. Vascular Screening of over 40s May Save 2000 Lives a Year. *British Medical Journal* 336, 7647 (April 2008), 737. DOI: http://dx.doi.org/10.1145/1188913.1188915

[6] Federica Barzi Rachel Huxley and Mark Woodward. 2006. Excess Risk Of Fatal Coronary Heart Disease Associated With Diabetes In Men And Women: Meta-Analysis Of 37 Prospective Cohort Studies. *British Medical Journal* 332, 7533 (Jan. 2006), 73–76. DOI: http://dx.doi.org/10.1145/1219092.1219093