HEART DISEASE PREDICTION

Gokul Chandrasekaran
Arizona State University
Tempe, AZ, USA

INTRODUCTION

Heart disease is one of the leading causes of early mortality in humans. The common factors among those facing early death is lack of exercise, high cholesterol levels, poor diet and an increasingly sedentary office work culture. One of the major advancements in the medical industry now helps us live longer with the heart conditions if diagnosed and treated early. During diagnosis, the medical professionals are provided with an array of data from which they evaluate the patient for heart conditions. In this paper we look at one of the comprehensive datasets aggregated from patients all over the world from 5 different locations and try to understand the important factors out of the presented data and build a statistical model to make it easier to assess patients in the future,

DATA SET

The data for this project is sourced from 'kaggle' titled "Heart Disease Dataset (Comprehensive)". It has aggregated patient data from multiple heart disease studies across 5 different datasets. It provides 11 parameters and the diagnosis of 1190 patients.

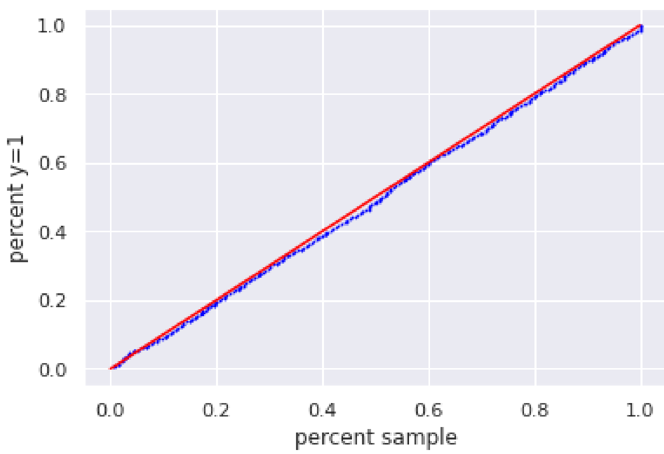
Parameter	Details
Age	A good distribution of people from ages 28-77
Sex	909 males and 281 females
Type of chest pain	Categorized into 1 typical, 2 typical angina, 3 non-anginal pain, 4 asymptomatic
Resting Blood Pressure	Level of blood pressure
Cholesterol	Serum Cholesterol
Fasting Blood Sugar	Blood sugar levels higher or lower than 120mg/dl
Resting ECG	0 in case of normal
Max Heart Rate	Heart rate
Exercise Angina	Reduced flow to heart during exercise
Old Peak	Exercise induced changes in ST curve in ECG
ST Slope	Measured slope of ST curve
Target	Binary Result: Heart Disease

CLASSIFICATION MODELS

The data is run through several models to predict the target result in the best way possible.

SIMPLE DECISION TREE

A decision tree makes it easier to classify a given data set into smaller homogeneous sets and evaluate the importance of different parameters in predicting the results.



The error in prediction is 0.341

Table 2: Variable Importance

age	0.000000
sex	0.077181
chest pain type	0.228055
resting bp s	0.000000
cholesterol	0.020603
fasting blood sugar	0.000000
resting ecg	0.000000
max heart rate	0.050456
exercise angina	0.035478
oldpeak	0.029185
ST slope	0.559041

Though we expect age to be a factor in heart issues, with the data available if there is an abnormal differences with the ST graph from ECG before and after exercise is a very clear indication of a patient having a heart disease. The preexisting level of chest pains are also a strong indicator for a potential disease. So, we see both ST slope and the chest pain type to be the most important parameters.

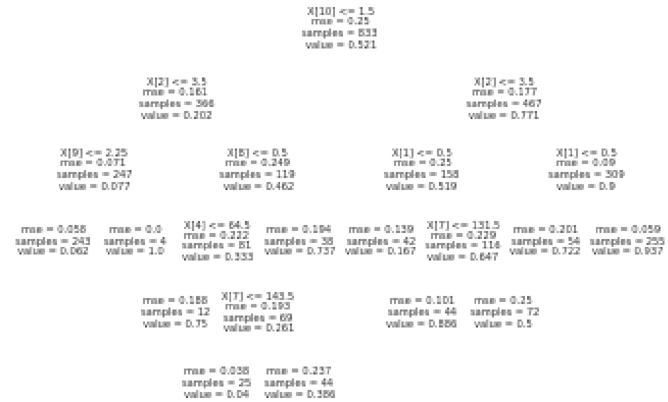
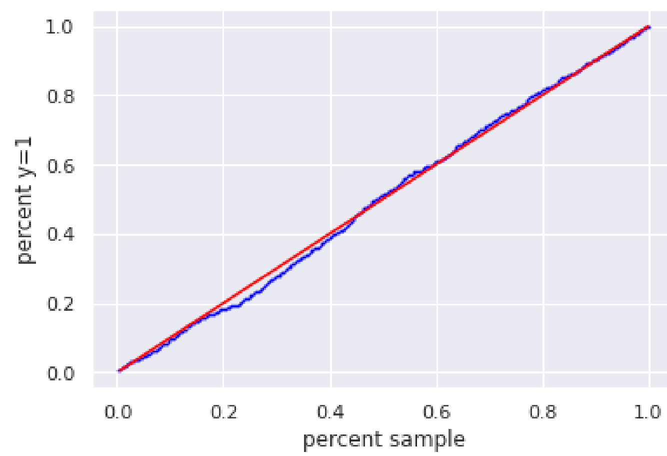
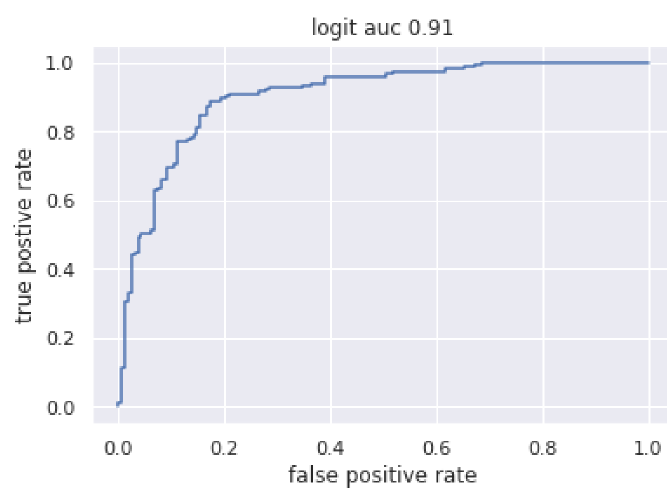


Figure: Decision Tree with 11 end nodes.

SIMPLE LOGIT

A simple logit model converged in 7 iterations for the given data set. With a simple logit model, we have a very high auc value of 0.91.

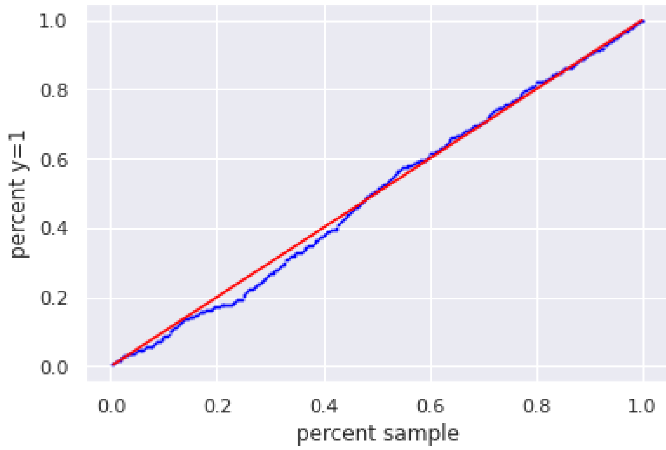




Again we find the ST slope to be a significant factor again. The prediction error is 0.46

MULTINOMIAL LOGISTIC REGRESSION

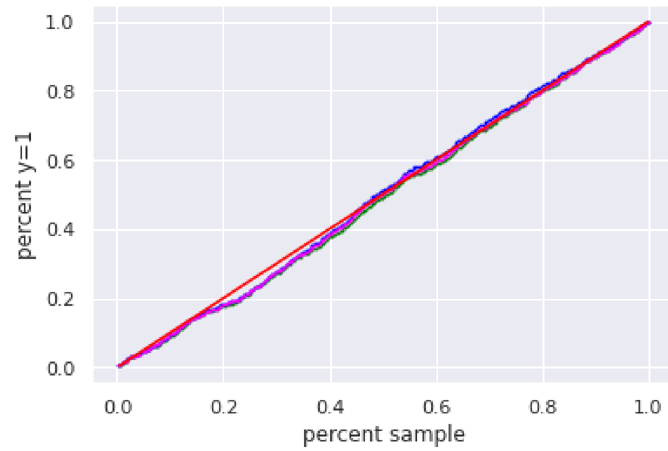
A multinomial logistic regression model yielded auc score of 0.84. It has a lower positivity rate than a simple logit model but has a lower prediction error

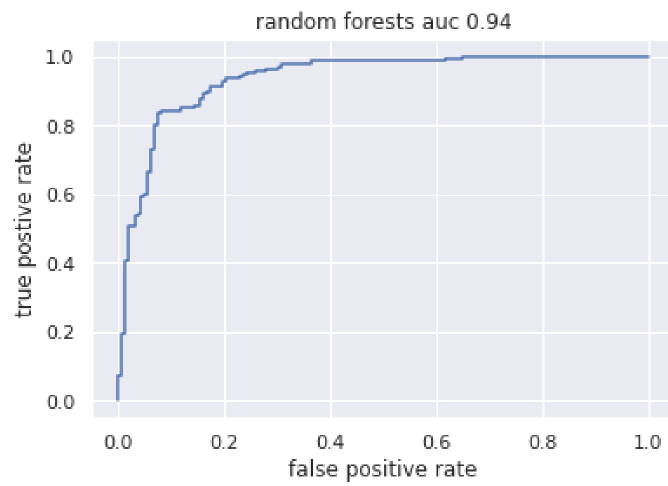


of 0.22. The model was iterated with different l2 penalty values to get the best solution. As expected higher the regularization penalty higher the accuracy.

RANDOM FOREST

A random forest model uses multiple randomized decision trees to classify the system. The random forest method has a higher positivity rate than a

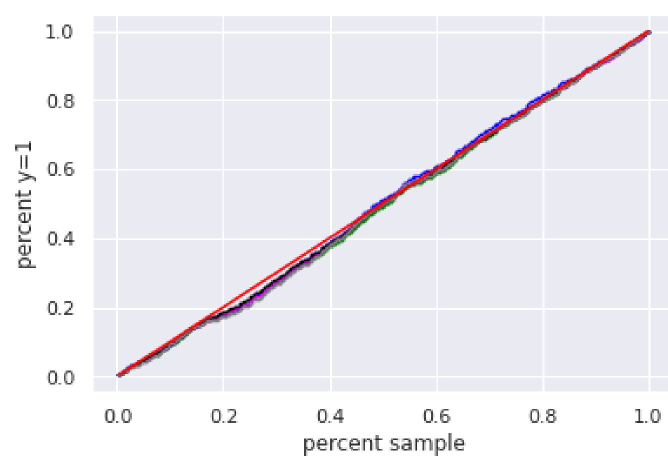


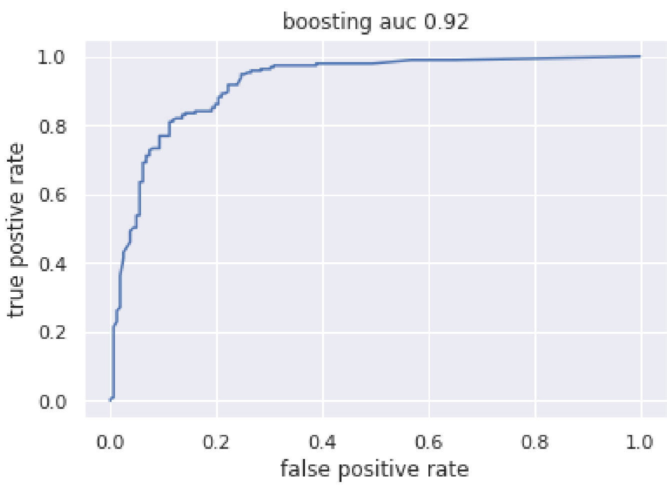


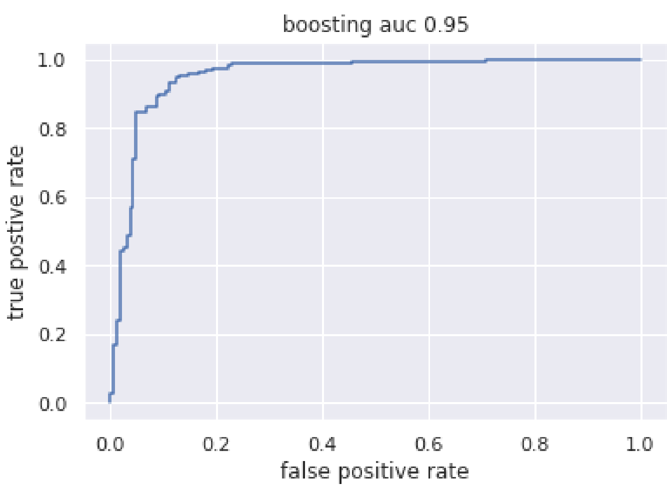
simple decision tree. The accuracy of the model is a little more accurate than all the previous models with a prediction error of 0.12. The feature importance factors tell us the same story as most cases above.

GRADIENT BOOSTING

A gradient boosting model is an ensemble of weak







prediction model. there is a possibility of overfitting leading to higher than expected prediction errors higher than other methods. But there is a higher positivity rate. Even here it is a strong correlation with the ST slope values.

DISCUSSION

The project started with a goal of predicting heart disease with the data of a patient's medical test results. As you expect ECG data before and after a strenuous exercise session definitely is the key indicator. The prediction error for different models

Model	AUC	Prediction Errors	Remarks
Simple Decision Tree	0.94	0.341	The classification is very accurate. but has a higher prediction error
Simple Logit	0.91	0.46	The high positivity rates on the classification data. It also creates a prediction model with highest error.
MuliLogistic Regression	0.84	0.19	He classification rate is very limited
Random Forest	0.94	0.12	Most accurate prediction model.
Gradient Boosting	0.95	0.490	Possible overfitting means it has the most positivity rate in data and a very high error in prediction.

FUTURE WORK

With an interest in learning about how people develop cardiac issues, I should have explored for a more intensive dataset. More learning algorithms can be explored and more intensive work could be done to get better prediction data.

APPENDIX A: REFERENCES

[1] https://www.kaggle.com/sid321axn/heart-statlog-cleveland-hungary-final

[2] Rob-mcculloch.org. 2021. *Machine Learning, STP 598, Spring 2021*. [online] Available at: <http://www.rob-mcculloch.org/2021_ml/webpage/index.html> [Accessed 1 May 2021].

APPENDIX B: CODE SNIPPETS

SIMPLE DECISION TREE:

```
## simple decision tree
nte=len(ytest)
# tree with at most 7 bottom nodes
tmod =
DecisionTreeRegressor(max_leaf_nodes=11)
tmod.fit(Xtrain,ytrain)

## look at in-sample fits
yhat = tmod.predict(Xtest)
c=np.cumsum(ytest[:nte])
p=c/c.iloc[nte-1]
cdt=np.cumsum(yhat[:nte])
pdt=cdt/cdt[nte-1]
plt.scatter(p,pdt,c='blue',s=0.5)
plt.xlabel('percent sample');
plt.ylabel('percent y=1')
plt.plot(ytest,ytest,c='red')
plt.show()
print("number of bottom nodes:
",pd.Series(yhat).nunique())

## variable importance
varimp = tmod.feature_importances_
print('variable importances:',varimp)
print(pd.Series(tmod.feature_importan
ces_,index=x.columns.values))

## plot a tree
tree.plot_tree(tmod)
print('rmse from tree, fit on train,
predict on test: ',myrmse(ytest,yhat))
```

SIMPLE LOGIT:

```
### logit
XX = sm.add_constant(xtr)
lfitM = sm.Logit(ytr, XX).fit()
XXp = sm.add_constant(xte)
phlgt = lfitM.predict(XXp)
```

MULTILOGISTIC REGRESSION:

```
lfitMulti =
LogisticRegression(multi_class='multinom
ial',
solver='lbfgs',max_iter=10000,penalty='l
2',C=0.1)
#lfitMulti =
LogisticRegression(multi_class='multinom
ial',
solver='lbfgs',max_iter=10000,penalty='l
2',C=1)
#lfitMulti =
LogisticRegression(max_iter=10000)
lfitMulti.fit(XX,ytr)
XXp = sm.add_constant(xte)
phlgtm = lfitMulti.predict(XXp)
```

RANDOM FOREST

```
### RANDOM FORESTS

RFM =
RandomForestClassifier(random_state=0,n_jobs=-1,n_es
timators=50,max_features=2,min_samples_split=20,oob
_score=True)

RFM.FIT(XTR,YTR)

PHRF = RFM.PREDICT_PROBA(XTE)[:,1]
```

GRADIENT BOOSTING

```
### GRADIENT BOOSTING

GBM =
GRADIENTBOOSTINGCLASSIFIER(LEARNING_RATE=.01,N_ESTIMA
TORS=100,MAX_DEPTH=4)

GBM.FIT(XTR,YTR)

PHGB = GBM.PREDICT_PROBA(XTE)[:,1]
```