Assignment 1: Forecasting

Joshua Bainbridge

250869629

Description of the Selected Forecasting Problem

Forecasting problem: evaluate a patient's medical history in order to predict the likelihood the patient currently has or is likely to contract heart disease.

Heart disease is the most common cause of death in the world. The ability to forecast the potential of an individual have or contracting heart disease could be a valuable resource for physicians, hospitals and patients.

This is classification forecasting problem. The final result should be able to accurately classify a patient as either having (or at risk of heart disease) or not having (or at risk of) heart disease base on a give set of independent cardiac related medical attributes.

Description of Available Data

The data set being used to build a forecasting algorithm was provided in a Kaggle competition posting [https://www.kaggle.com/fedesoriano/heart-failure-prediction]. The dataset is made of 920 individual patient medical information. Each patient in the data set is described by two background attributes, nine medical attributes with some relationship to heart failure and a single classification attribute denoting whether or not the patient has heart disease. All data is being used to complete this assignment.

Background Attributes

The background attributes for this data set are age and sex. These attributes are considered background attributes because they do not relate directly to medical conditions but are clinically relevant when determining the likelihood a patient has heart disease.

Medical Attributes

Medical attributes are attributes which describe medical conditions which have been shown to have a relationship with heart disease in patients. These attributes include chest pain type, resting blood pressure, cholesterol levels, fasting blood pressure, resting ECG, maximum heart rate, exercise angina, exercise relative to rest (old peak) and exercise induced increments in heart rate (ST slope). These attributes are used in the dataset because individually each of them is often found in patients with heart disease, however patients present with unique variations and levels of each of these conditions.

Numerical Attributes

Resting blood pressure, cholesterol levels, fasting blood pressure, maximum heart rate and old peak attributes are given as numerical values. After examining each attribute no large outliers or excessively large or small values were found. Therefore the data did not need to be normalized prior to being used in the three forecasting algorithms.

Ordinal Attributes

Sex, chest pain type, resting ECG, exercise angina and ST slope are all given in the data set as ordinal attributes. Prior to being used in a forecasting algorithm the data was vectorized. The table below details the results of the vectorizing process.

Table 1. Vectorized Data from Given Ordinal Data

Sex	Vectorized	Chest Pain Type	Vectorized	Resting ECG	Vectorized	ST Slope	Vectorized	Exercise Angina	Vectorized
M	0	ATA	0	Normal	0	Down	0	N	0
F	1	NAP	1	ST	1	Flat	1	Y	1
		ASY	2	LVH	2	Up	2		
		TA	3						

Classification Attribute

The final attribute is a heart disease attribute. This attribute represents the dependent variable in this forecasting problem. In the given data set the attribute is given as a binary value. The value is either zero, the patient does not have heart disease or one, the patient has heart disease.

Short Overview of the Selected Algorithms

Multiple Linear Regression

Multiple linear regression is a variation of standard linear regressions that uses multiple independent variables to predict the outcome of a single dependent variable. Multiple linear regression predicts the value of the dependent variable based on a summation of the weighted values of all independent variables. As the algorithm is training the values of each weight being multiplied with each independent variable is individually changed.

Support Vector Machine

Support vector machine is a linear regression algorithm that defines the amount of error that the model is allowed to have and define the fit that adheres to it. Using the SVM over MLR increases the robustness of the machine learning model.

Neural Network:

Neural networks are machine learning algorithms modeled after the neurological structure of a brain. The network takes independent inputs into perceptron along with weight values. The combination of the input and weight value determines whether or not the perceptron will fire in the next layer. Through the training process the weights are adjusted until the network can accurately fit the data.

Specifics About How Algorithms were Applied and the Evaluation Procedure

Preprocessing Steps

The program begins by reading in the data and separating the attributes into two data frames. The independent attributes are store in the X data frame and the Y data frame contains the single dependent attribute.

Once the data set is split into dependent and independent data, it is split into training and test sets. The split used in this assignment is the hold-out method. The data split of the data used in this assignment the split of data is 80% (736 data points) are used as training data and 20% (184 data points) are used at testing data.

Training steps

Once the data set is separated into independent and dependent variables and further split into training and test data, the training data is used in the algorithm. In the python code creates and then fits each model to the randomly selected training data sets to create a model to be tested.

Multiple linear regression (MLR): applied using the *sklearn.learn_model* library using *LinearRegressionl* in python.

Support vector machine (SVM): applied using the *sklearn.svm* library in python. Multiple kernels were experimented during the training and testing of the program. The polynomial kernel was selected for use in the final program.

Neural network: applied using the *sklearn.neural_network* library using *MLPClassifier* in python. Multiple solvers were experimented during the training and testing of the program. The 'lbfgs' solver was selected for use in the final program.

Evaluation

Evaluation Method

The forecasting problem is a classification problem with a binary final output. As a result using accuracy and evaluation methods sure as RMSE and MAE does not yield very useful information. Instead, to evaluate the accuracy of each model, the model's classification of a patient in the testing data set was comparted to the known classification of the patient. With this information a table containing the attributes, true positive, true negative, false positive and false negative. The accuracy of the models is then calculated based on number of correctly classified samples divided by the total number of samples. Furthermore, the program also evaluates the accuracy of the models when classifying a patient as positive or negative to see if one is more commonly miss classified, if it turns out the model is not accurate.

Additional Evaluation Step for MLR and SVM

The results of MLR and SVM both yield a non discretized value. To properly evaluate these results the data was discretized. Results that were ± 0.3 of 1 were classified at a positive (patient has heart disease), and values ± 0.3 of 0 were classified at a negative (patient has heart disease). Using this method also resulted in data being left unclassified.

Comparison of Results Obtained with Different Algorithms

Each model was tested five times. The accuracy was calculated and averaged over each of the trails. This was done to attempt to remove some of the dependencies related to the training, testing data split, as the cross-validation method was not used in this assignment. Examples of the results outputted by the program can be found in Appendix A.

Conclusion

Multiple Linear Regression

Multiple linear regression had the lowest accuracy of all three algorithms tested. The final accuracy averaged over all five trials was 32% with an average of 56 out of 184 data points being unclassified (UC). The highest accuracy achieved using the multiple linear regression algorithm was 34%. The lowest number of unclassified data points was 50.

With an accuracy that is lower than simply randomly diagnosing a patient, it is clear using a linear model does not accurately model the system. It is likely the relationship between the independent attributes and a patient having or not having heart disease in not a linear relationship.

Support Vector Machine

The support vector machine algorithm performed second best out of the three algorithms tested. The final accuracy averaged over all five trials was 55% with an average of 43 out of 184 data points being unclassified. The highest accuracy achieved using the support vector machine algorithm was 74%. The lowest number of unclassified data points was 37.

The support vector machine performed slightly better than flipping a coin to decide the final result, with fewer unclassified patients. However the performance what not close enough to be considered a solution to the assignment problem. Using the polynomial model to fit data is not a close enough fit to accurately model the data.

Neural Network

The neural network performed the best out of the three algorithms tested. The final accuracy averaged over all five trials was 93% with an average of 0 out of 184 data points being unclassified. The highest accuracy achieved using a neural network was 100%. The lowest number of unclassified data points was 0.

The neural network outperformed both MLR and SVM algorithms. Allowing the neural network architecture of adjusting perceptron was the method that could best fit the given data. The resulting 93% accuracy is still not a high enough level to be considered a solution to the assignment problem. However the neural network is them best algorithm to used when forecasting heart disease in patients.

Results

The final results are shown in the table below.

Table 2. Final Results

Experiment	MLR					
	TP	TN	FP	FN	Accuracy	UC
1	7	46	69	3	29%	59
2	62	1	6	48	34%	67
3	12	49	71	2	33%	50
4	5	58	65	4	34%	52
5	7	45	62	20	28%	50
				Average	32%	56
Experiment	SVM					
	TP	TN	FP	FN	Accuracy	UC
1	71	55	17	4	68%	37
2	87	50	0	0	74%	47
3	11	53	62	5	35%	53
4	11	57	70	3	37%	43
5	75	37	9	30	61%	33
				Average	55%	43
Experiment	NN					
	TP	TN	FP	FN	Accuracy	UC
1	86	70	18	10	85%	0
2	109	75	0	0	100%	0
3	120	64	0	0	100%	0
4	104	80	0	0	100%	0
5	85	64	21	14	81%	0
				Average	93%	0

Appendix A

Multiple Linear Regression Example Results

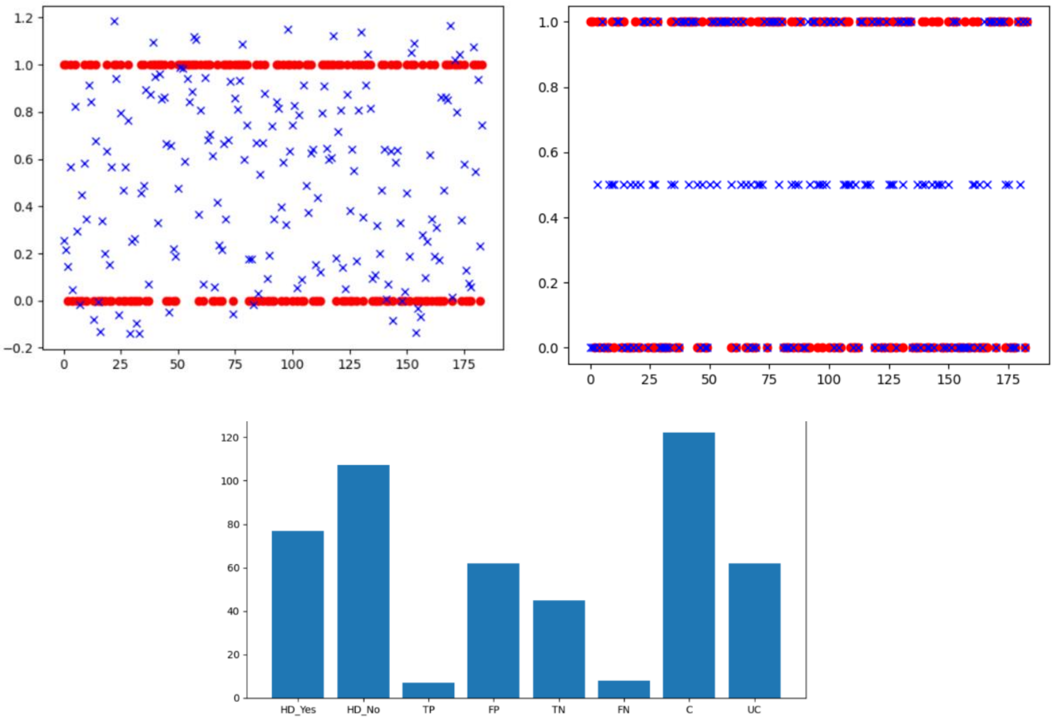

Figure 1. MLR Results: Trial 4. Initial Results (top left), Result using Min-Max grouping (top right), Final results (bottom)

Support Vector Regression Example Results

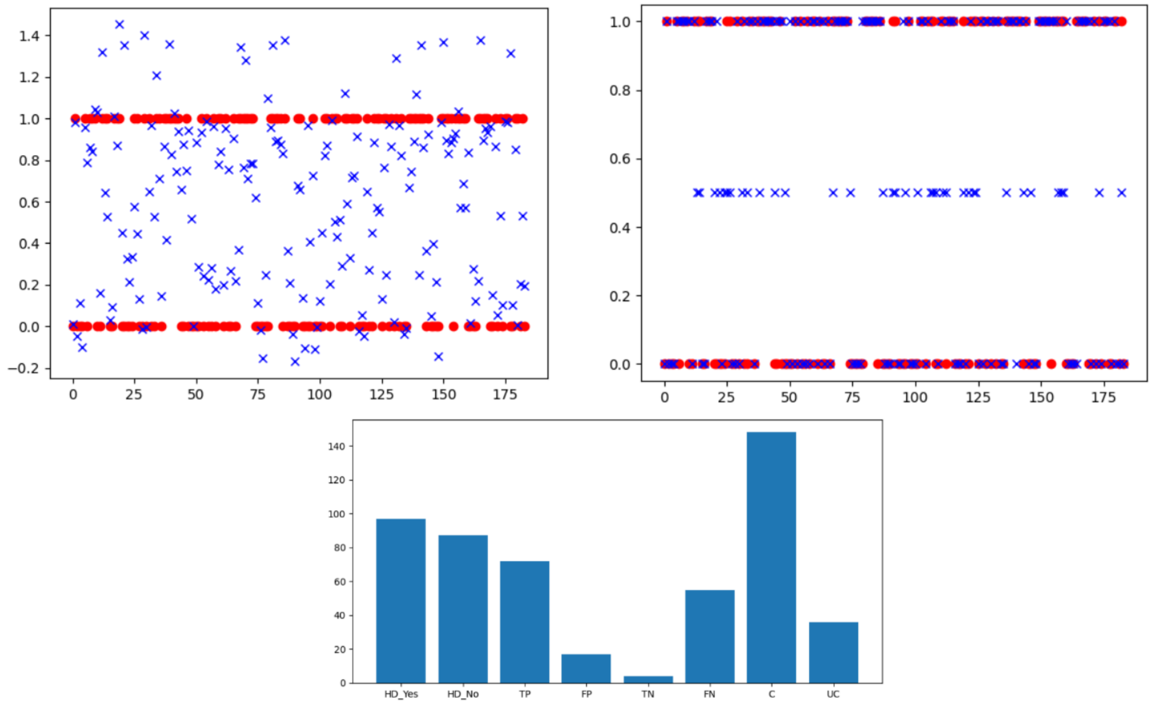

Figure 2. SVM Results: Trial 2. Initial Results (top left), Result using Min-Max grouping (top right), Final results (bottom)

Neural Network Example Results

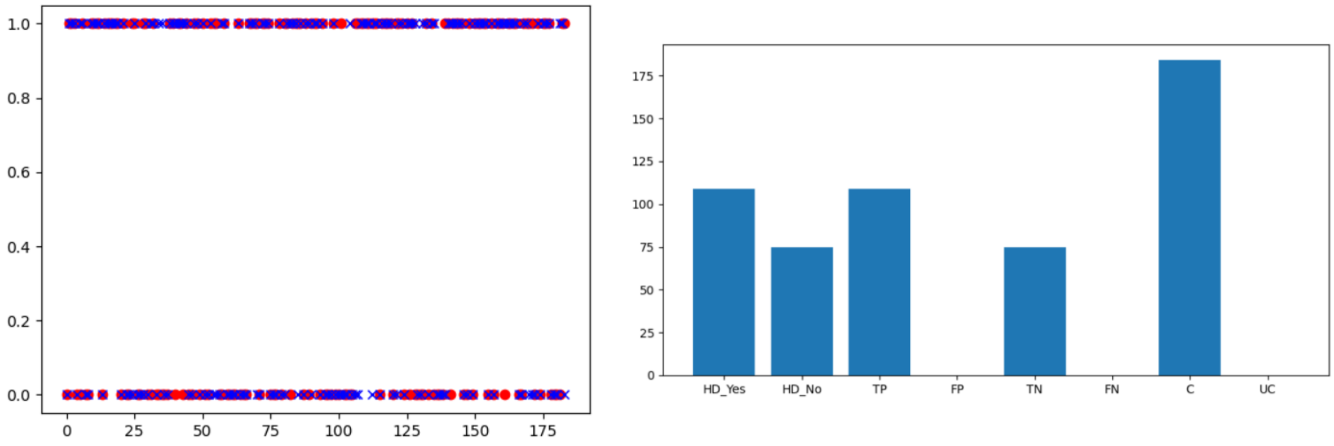

Figure 3. Neural Network final results