MAI 2017 - Intelligent Decision Support Systems

June 2017

HEART DISEASE DETECTION

Carles Coll Gomila

Daniel García Zapata

Barath Bheeman

Description of the domain of application

Heart - Basic Anatomy.

Heart is one of the most vital parts of the human body. At the size of a closed fist, heart is the hardest working muscle in the body. At an average rate of 80 times a minute, the heart beats about 115,000 times in one day, or about 42 million times in a year. In a 70-year lifetime, an average human heart will beat more than 2.5 billion times. Even when a person is at rest, the heart continuously works hard.

The heart is responsible for circulating blood throughout your body to supply the tissues with oxygen and nutrients. The cardiovascular system is made up of:

1. Four chambers- 2 Atria that receive blood coming back to the heart and 2 ventricles that pumps blood out of the heart.
2. Blood vessels, which are made up of a network of arteries (transport blood from the heart to the body) and veins (transport blood to the heart from the rest of the body)
3. Four valves, to prevent the backward flow of blood.
4. An electrical system that serves as a natural pacemaker and stimulates contraction of the heart muscle.

Heart Disease

Heart disease describes a range of conditions that affect your heart. Diseases under the heart disease umbrella include blood vessel diseases, such as coronary artery disease; heart rhythm problems (arrhythmias); and heart defects you're born with (congenital heart defects), among others.

Coronary heart disease is a common term for the buildup of plaque in the heart's arteries that could lead to heart attack. However, coronary heart disease, or CHD, is a result of coronary artery disease, or CAD, said Edward A. Fisher, M.D., Ph.D., M.P.H. [1]

With coronary artery disease, plaque first grows within the walls of the coronary arteries until the blood flow to the heart's muscle is limited. This is also called ischemia. It may be chronic, narrowing of the coronary artery over time and limiting of the blood supply to part of the muscle. Or it can be acute, resulting from a sudden rupture of a plaque and formation of a thrombus or blood clot. [1]

Congenital heart disease is one of the major causes of death in children as well as adults. But, more often than not, such diseases are not properly diagnosed or diagnosed very late into a patient's life. With a proper automated diagnosis system, most of those lives can be saved.

With the increasing application of artificial intelligence techniques to healthcare research, it has become easier to diagnose people with an elevated level of certainty diseases while also, diminishing the number of visits to hospitals, bad diagnosis and/or the helping doctors make more accurate decisions. This has led to a revolution in healthcare, where doctors can diagnose more effectively and quicker than before.

People with heart disease might develop some typical symptoms such as chest pain, shortness of breath, palpitations and even fatigue. However, the cases where the symptoms are present are few. In many cases the person might not experience any symptom until they have a heart attack [2].

In this work, we propose an intelligent decision support system that aids the doctors at diagnosing correctly the presence or absence of heart disease. The support system provides a clear an instantaneous diagnosis to the medical personnel. To accurately diagnose the patient must take some exams and measurement. Our proposed system consists mainly of:

1) Data collection of different data sources.
2) Pre-processing of the data, which consists of standardizing and normalizing.
3) Training of a deep learning architecture.
4) Graphical user Interface and visualization of the results.

Main identified decisions

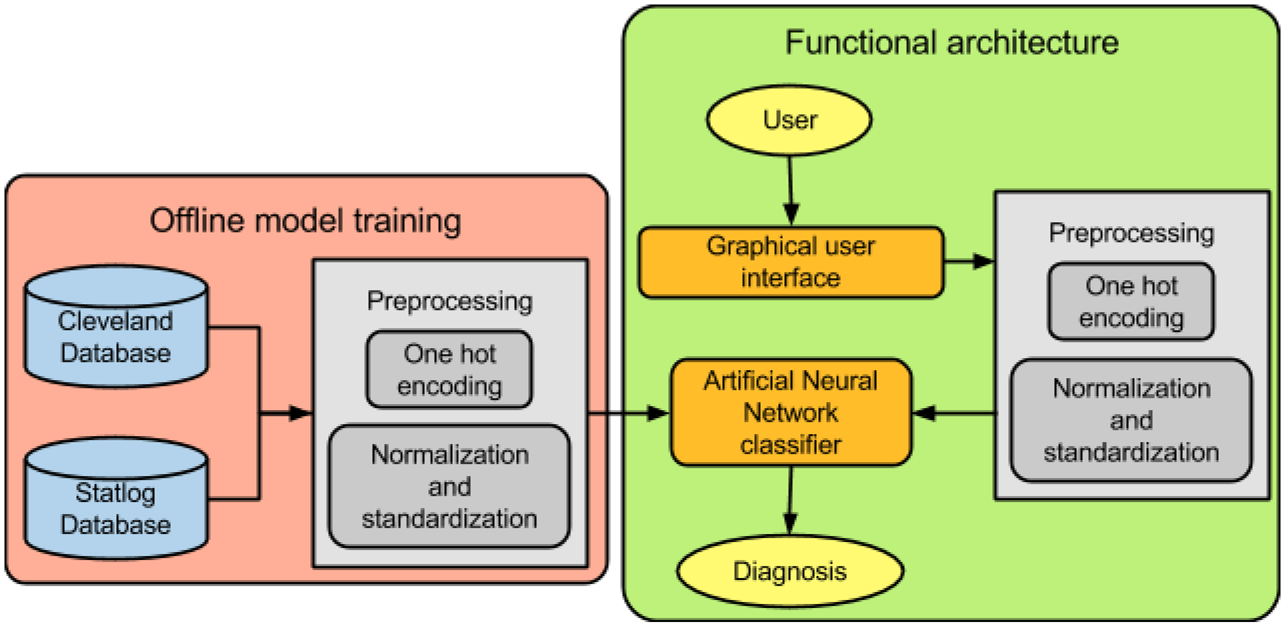
Heart disease can be treated adequately when it is diagnosed in an early stage. Since the obtaining a correct diagnosis is difficult due to complicated components of heart ailments. It is helpful to support the medic staff to give a good analysis on time. The diagnosis procedure can be considered a decision-making process, where the medical personnel must decide based on information available through clinical data in conjunction with the medical staff expertise. In this work, we propose an intelligent decision support system to help doctors diagnose the presence or absence of coronary artery disease.

Heart Defects are basically diagnosed in two stages. These stages are Physical Evaluation and Clinical Evaluation of a patient. During Physical Evaluation stage, a physician gets the signs, symptoms of a patient and also records the measurements like Systolic Blood Pressure, Diastolic Blood Pressure and Heart Beat Rate etc. A patient is suspected to have heart disease based on these signs, symptoms and measurements only. If a patient is suspected of having a disease in the first stage itself only then the doctors order the second stage tests and treatment(Clinical evaluation). But suspecting a disease at first stage is not an easy task, because some patients may have signs and symptoms and some may not have, in which the diagnosis depends on only the experience with the previous cases. In which there is a chance for taking a wrong decision.

If he takes a wrong decision by not suspecting a disease at the first stage then he never orders for the second test, where a disease can detected. In this case even though if a child has a defect it won’t be detected which causes, the patient to enter into the severe condition. Also if he takes a wrong decision by suspecting a child at the first stage itself and orders for the second test then even though the child doesn’t has a defect, he should be clinically tested which causes for more time to know the results and is a cost effective.

These drawbacks can be overcome by using Neural Network Techniques. In the present study a most frequently used Deep Neural Network Model is used to perform the Heart Disease Diagnosis classification based on the signs, symptoms and physical evaluation of a patient Since the Neural Network solutions will not depend on algorithmic solution instead it depends on examples of the previous cases it gives more accurate results than the human diagnosis or diagnosis done with simple machine learning classifiers.

Functional architecture of the IDSS prototype



Dataset

The data that we have used to build the decision support system consists of two different datasets; both have the same attributes, 14 in total.

- Cleveland Heart Disease Dataset, which contains 303 instances. Published in 1988.
- Statlog Heart Dataset, which contains 266 instances. Published in 1993.

The dataset were obtained from the following hospitals:

1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.
2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.
3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.
4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation:Robert Detrano, M.D., Ph.D.

Data pre-processing summary

This step takes data from the Cleveland Heart Disease dataset and Statlog (Heart) dataset, both from UCI website. The Cleveland dataset is divided into four smaller datasets from different hospitals. The first step is to combine these datasets into one. Later we combined this dataset with Statlog (Heart).

Once we have the two datasets combined, we proceed to process categorical variables. Categorical data must be transformed into one hot encoding to be more reliable. Specifically, we create dummy variables encoding the categories from the variable. The next step is to standardize and normalize the data. We chose to normalize the discrete variables and standardize real values.

- Normalization scales all numeric variables in the range [0,1], it also retains the original distribution [3]. We decided to normalize variable 'ca'. This variable describes the number of major vessels colored by fluoroscopy. The main motivation to use normalization is the discrete nature of the variable.

$$x'_k = \frac{x_k - min}{max - min}$$

- Standardization transforms the data to have zero mean and unit variance; this measures how many standard deviations below or above the population mean a score is [3]. We decided to standardize the following variables: 'age', 'restbp', 'chol', 'thalach', and 'oldpeak'.

$$x'_k = \frac{x_k - \mu}{\sigma}$$

Data Validation

The data that we have used to build the decision support system consists of two different datasets that have the same number and types of attributes. One is the Cleveland Heart Disease Dataset, which was published in 1988 and is divided into four smaller datasets from different hospitals. The principal investigator responsible for the data collection at each institution are [5], [6], [7], [8]. The other dataset is the Statlog Heart Dataset, which was published in 1993. The Cleveland dataset contains 303 instances with 14 attributes, and the Statlog dataset contains 266 instances with the exact same attributes. There is a difference in the way the last attribute (the target) is represented: in the Cleveland dataset 0 means absence of heart disease and the other values represent presence of heart disease while in the Statlog dataset 1 represents absence and 2 indicates presence of heart disease. During the preprocessing this discrepancy is resolved. Below the reader can find a table with the description of the 14 attributes and their types.

Attribute	**Value**	**Description**
Age	real	Age of the patient
sex	binary	(1: male, 0: female)
cp	nominal	chest pain type (1: typical angina, 2: atypical angina, 3: non-anginal pain, 4: asymptomatic)
trestbps	real	resting blood pressure (in mm Hg on admission to the hospital)
chol	real	serum cholesterol in mg/dl
fbs	binary	fasting blood sugar > 120 mg/dl (1: true, 0: false)
restecg	nominal	resting electrocardiographic result (0: normal, 1: having ST-T wave abnormality, 2: showing probable or definite left ventricular hypertrophy)
thalach	real	maximum heart rate achieved
exang	binary	exercise induced angina (1: yes, 2: no)

oldpeak	real	ST depression induced by exercise relative to rest
slope	nominal	the slope of the peak exercise relative to rest (1: upsloping, 2: flat, 3: downsloping)
ca	real	number of major vessels (0-3) colored by flouroscopy
thal	nominal	3: normal, 6: fixed defect, 7: reversable defect
num	nominal	Cleveland; diagnosis of heart disease (0: < 50% diameter narrowing, 1: > 50% diameter narrowing) Statlog; 1: absence and 2: presence

The described data has been used to train and test our classifiers, assigning 75% for training and the remaining 25% for testing. The dataset is also quite balanced, with 310 instances belonging to the absence of category and 259 belonging to the presence of disease category.

Flowchart of the data-driven model/s gathering

The flowchart of the proposed IDSS for detection of Heart Disease is straightforward. The whole detection system relies on the classification results obtained from the ANN. Therefore, the system does not have any ensemble of methods and the flowchart is simply a sequence of operations: first the feature vector is built from the information provided by the user through the graphical user interface, and second the feature vector is fed to the prediction module, the deep ANN, which provides the final diagnosis.

The reason why we are not using an ensemble of classifiers is because we think that the performance of the ensemble would not be significantly better than the performance of the ANN. The performance of the ANN is good enough and leaves a very short margin for improvement. This is accentuated by the small amount of data we are working with. Another reason for not using an ensemble of techniques is the gain in computation efficiency we obtain by using a single but powerful classifier.

Methodology

The proposal decision support system for heart disease prediction utilizes a deep architecture. More concretely a feedforward multilayer perceptron architecture of neural network. The system consists of two steps, in the first step 13 clinical features are received as input and then we train the neural network with the training set. done with training data by back-propagation learning algorithm. Specifically, we train the neural network with an stochastic optimization method: adaptive moment estimation (ADAM). The model consists of 3 hidden layers with a relu activation function and an output layer with a softmax activation function. The first hidden layer has 22 neurons, the second has 30 and the third layer has 40.

Model Architecture

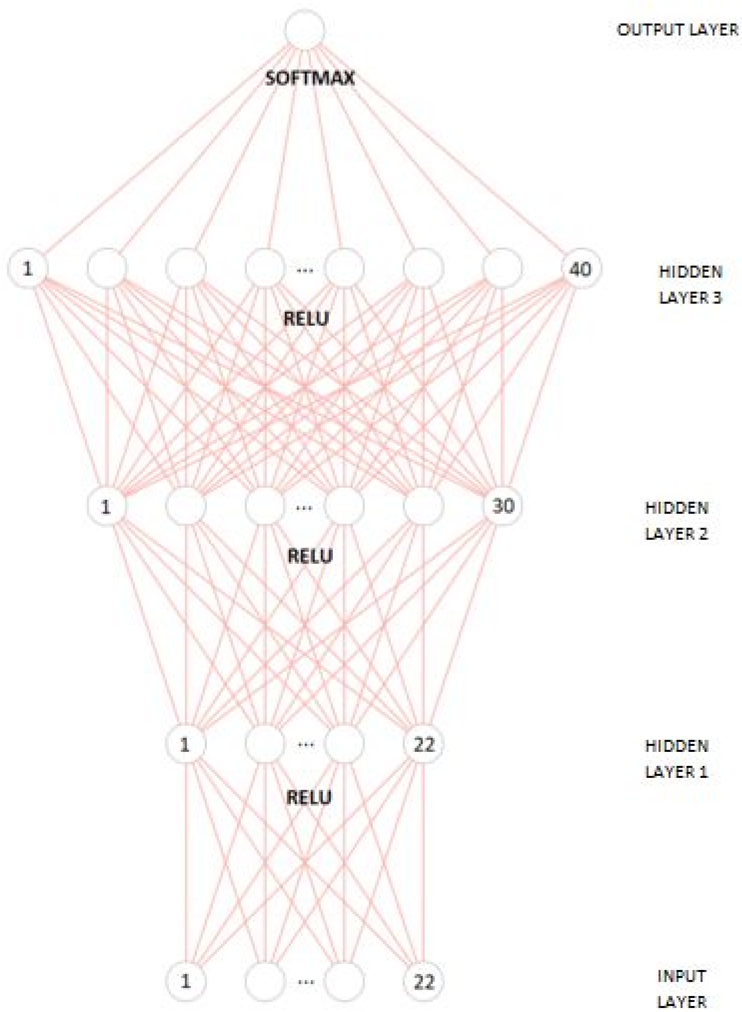
We use a multi-layer perceptrons (MLP), a form of feedforward artificial neural networks which are built by stacking multiple perceptrons [10] on top of each other. The number of neurons in the input layer is fixed by the number of features used, 22 in our case. Finding the right number of hidden layers and the number of units within them, on the other hand, remains one of the unsolved tasks in this research area [11]. Since there is no rule or algorithm to determine the optimal number of hidden layers, we decide to test with one, two and three hidden layers. In most cases, there is no correct way to determine the best configuration for hidden units without training the several neural nets and estimating the generalization error. According to Geman, Bienenstock, and Doursat [12], if you have too few hidden units, you could get a high training error and high generalization error due to underfitting and high statistical bias. If you have too many hidden units, you may get low training error but still have high generalization error due to overfitting and high variance.

Comparison table of relevant artificial neural network tests:

Number of Hidden Layers	Number of Hidden Units	Test Accuracy
1	[20]	0.93
1	[22]	0.923
1	[30]	0.938
2	[22, 20]	0.923
2	[22, 30]	0.940
3	[22, 30, 20]	0.930
3	[22, 30, 30]	0.944
3	[22, 30, 40]	0.916
3	[22,26,30]	0.909

The final architecture is a Deep Neural Network with three hidden layers. The first layer consists of 22 neurons, one for each input. The second layer has 30 neurons and the third hidden layer has 30 hidden neurons. Each hidden layer has a ReLu activation function; we chose this activation to avoid any vanishing gradient problem. Finally, the last layer consist of a softmax activation.

Heart Disease DNN Architecture



Evaluation of results and conclusions

The detection of heart disease in patients given a set of medical measurements is essentially a classification problem. In order to find the best classifier we have tested several well known methods: logistic regression, AdaBoost, support vector machine and a deep neural network.

It is important to note that since we are trying to detect a disease that can cause the death of the patient if he or she does not receive the treatment, we need take into account other performance measures apart from accuracy. The confusion matrix is a great tool for visualizing the performance of the classifiers and it allows us to compute precision and recall measures. In the context of heart disease detection we can consider two different classification errors. First, a patient without heart disease that is classified as having heart disease (also known as type 1 error or false positive) and second, a patient suffering heart disease that is classified as healthy (also known as type 2 error or false negative). From the point of view of the classifier, both types of errors should be minimized in the same way but, from the point of view of the patient, a false negative is much more critical than a false positive, because it puts the life of the patient at risk.

Precision is a kind of measurement that allows us to measure the ratio of type 1 and type 2 errors separately. In our case, we will consider the precision of the type 2 error or negative predictive value (NPV) because it represents the proportion of healthy patient that are classified as healthy. The closer to 1 the better because this means that we are minimizing the number of false negatives, which represent the patients with heart disease that are classified as healthy.

We tested in first place the logistic regression classifier and we got the worst results with an accuracy value of 86.84% and a NPV of 0.85, which indicates that the system won't be able to detect 15% of the patients with heart disease. Then we tried the non-linear ensemble classifier AdaBoost, which turned out to be slightly better than logistic regression, with an accuracy of 89.51% and an NPV of 0.89.

After that, we tried the support vector machine classifier with a radial basis kernel function. But the problem with this configuration is that we have two hyperparameters: the C parameter that is used to balance the classes, and the gamma that is a parameter of the radial basis function. When we have hyperparameters one option is to try different combinations of parameters and try to gain some knowledge about the relation between performance and parameter value. But this approach is not rigorous and can be useless If we have a large number of hyperparameters. In consequence, we searched for an automatic way to optimize these parameters and we finally found the Optunity python library for hyperparameter optimization.

To have an idea about the effectiveness of Optunity, we have compared the performance of a support vector machine classifier without optimization of hyperparameters versus the optimized support vector machine. The obtained results prove the effectiveness of the optimization of hyperparameters, but we must say that given the small amount of data and the very low number of hyperparameters we have, we believe the same performance could be achieved by manually trying several combinations of parameters. The performance of the optimized support vector machine classifier was even better than AdaBoost with an accuracy of 92.30% and an NPV of 0.93.

Finally, for the last classifier we tested a fully connected deep neural network. For its implementation we used the Tensorflow library because it provides many interesting features for the construction of neural networks and visualization of results. The final neural network we have built has three hidden layers, the input layer has 22 nodes and the output layer two. The first hidden layer has 22 neurons, the second hidden layer has 30 nodes and the third hidden layer has 40 nodes. The network has a learning rate of 0.001, is trained for 1000 epochs and the batch size is 100. With this structure and hyperparameters we managed to get a 95.10% accuracy and an NPV of 0.94.

After comparing the performance of the different classifiers we can conclude that the best method to diagnose heart disease based on the available features is the fully connected feed-forward deep neural network, therefore it is the model implemented in the final Decision Support System. However, we must add that even though the neural network shows clearly the best performance, its performance has some variability with respect to the initialization of the weights, while the other techniques are much more consistent with their performance. The results of the different classifiers are all gathered in the table below. The Negative Predictive Value and Positive Predictive Value are computed from the confusion matrix.

	Logistic Regression	AdaBoost	SVM	deep NN
Accuracy	85.31%	89.51%	92.31%	95.10%
Negative Predictive Value	0.84	0.89	0.93	0.94
Positive Predictive Value	0.87	0.91	0.91	0.96

The Cleveland and Statlog datasets are old datasets (late 1988 and 1993 respectively) and have been used in several research works. Therefore we provide a comparison between the performance of the techniques we have tested in this project against the results of the older research works. It is important to note that the previous works always use only one of the two datasets so we have retested our models with the datasets separately.

Dataset	**Technique**	**Accuracy**
Statlog	Ensemble of 2 ANN with 1 hidden layer	81.92%
	SVM with RBF kernel	83.70%
	Naive Bayes	84.50%
	k-NN	82.90%
	Our Logistic Regression	82.35%
	Our AdaBoost	88.23
	Our SVM	82.35%
	Our ANN	77.90%
Cleveland	C4.5 [9]	77.56%

	Naive Bayes [9]	83.50%
	SVM [9]	84.12%
	Our Logistic Regression	86.66%
	Our AdaBoost	85.33%
	Our SVM	86.66%
	Our ANN	80%

As the table shows, the accuracy drops in general when using only one of the two datasets. Even the models that we have tested have much worse accuracy, which can be explained by the fact that we now have about half of the original number of samples. Other interesting things to mentions are that the AdaBoost classifier gives very goods results with the Statlog dataset compared to the other techniques, and that our ANN has the worst accuracy. The reason for the bad results of the ANN is that we are using the same model architecture that we use with the two datasets which is optimal for that particular setting, and with such a significant reduction of samples it proves to be useless.

Future work and improvements

The system we have developed provides highly accurate results (94%). But it still remains to be seen that whether the system will work for larger quantities of data, since we the data set we have is very small (569 rows). Another very useful feature would be to enhance the Graphic User Interface (GUI) already presented to the doctors. The GUI will not only display the result of the prediction, but also some insights of the patient. Identify the features that are more important for the prediction such as high cholesterol, etc.

Software tools

The Intelligent decision support system developed in this project is fully implemented in python and uses several relevant libraries that add functionalities. We have chosen the programming language Python because it is a high level language that makes the development of software applications very simple and also it has very complete and powerful libraries in the field of machine learning and data science. The drawback of python is its efficiency, which can be much lower than other lower level programming languages like C. However, efficiency is not an important issue in our project because we do not have a huge amount of data and the number of dimensions of the data is also low, so all training is achieved very fast.

An important library used in this system is scikit-learn because it is used to create, train, and test Logistic Regression, AdaBoost and SVM classifiers. For the creation, training and testing of the deep ANN we use TensorFlow because it is a very powerful library for developing complex Machine Learning applications.

For the preprocessing of data we use pandas library because it makes all the preprocessing much easier once the data is loaded in a dataframe. The library provides a function to load a csv file into a dataframe, it allows to easily apply operations to a whole column, which is useful when normalizing and standardizing the data, and finally it provides a function to generates dummy features from a categorical feature. The library Optunity has been used to find the optimal hyperparameters of the SVM with RBF model.

Finally, we use ipywidgets and IPython libraries to create and support the GUI of the system. Through this GUI the user can enter the different measurements from the patient into the system using sliders and dropdowns, run the decision system and see the resulting decision. Below is a screenshot of the GUI.

Tasks assignment and responsibilities among group members. Gantt diagram with tasks planning

Gantt diagram for the project was generated and maintained using the software "mindview". The detailed pdf is attached in the project folder. Below is the attached screenshot (images are available in the directory) of the list of tasks and the Gantt chart of the plan:

	i	Tas...	Task Name	Duration	Start	End	Predecessors	Completion	Resources
1			**Heart Disease Detecti...**	**36 days**	**4/20/2017**	**6/8/2017**		**97%**	
2			**⊟ 1. Planning**	**18 days**	**4/20/2017**	**5/15/2017**		**100%**	**Barath, Carles, Daniel**
3			1.1 Project Schedul...	6 days	4/20/2017	4/27/2017		100%	Barath
4			1.2 Model Selection	6 days	4/28/2017	5/5/2017	3	100%	Carles
5			1.3 Tool Selection	6 days	5/8/2017	5/15/2017	4	100%	Daniel
6			1.4 UI Design	5 days	4/28/2017	5/4/2017	3	100%	Barath
7			**⊟ 2. Simple Classifiers**	**16 days**	**5/15/2017**	**6/5/2017**	**2**	**100%**	
8			2.1 Data Pre-proce...	4 days	5/18/2017	5/23/2017		100%	Carles
9			2.2 Decision Tree	3 days	5/24/2017	5/26/2017	8	100%	Carles
10			2.3 Random Forest	4 days	5/29/2017	6/1/2017	9	100%	Carles
11			2.4 Logistic Regres...	2 days	6/2/2017	6/5/2017	10	100%	Carles
12			2.5 Support Vector...	4 days	5/24/2017	5/29/2017	8	100%	Barath
13			**⊟ 3. Neural Network**	**13 days**	**5/18/2017**	**6/5/2017**		**100%**	**Daniel**
14			3.1 Data Processing	3 days	5/18/2017	5/22/2017		100%	Daniel
15			3.2 Tensor Flow	7 days	5/23/2017	5/31/2017	14	100%	Daniel
16			4. UI	10 days	5/10/2017	5/23/2017	6	80%	Barath
17			5. UI - Program Linking	3 days	6/1/2017	6/5/2017	16	100%	Barath, Daniel
18			6. Report	2 days	6/2/2017	6/6/2017		100%	All

Figure: List of tasks, duration, and resources.

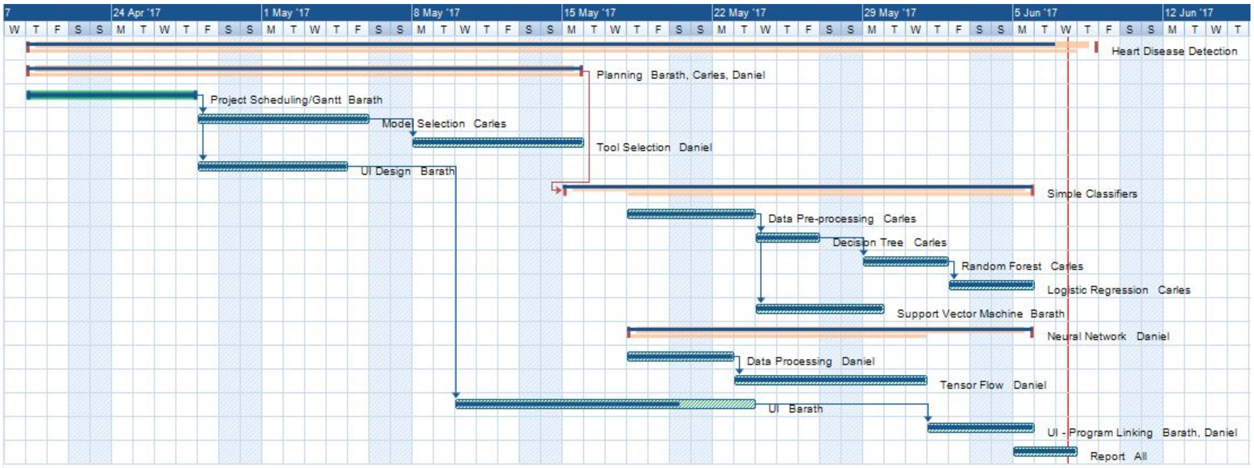


Figure: Gantt Chart of the project timeline and contributing resources.

Bibliography and Acknowledgement

[1] American Heart Association, http://www.heart.org/HEARTORG/, (2014)

[2] R. Das, I. Turkoglu and A. Sengur, "Effective diagnosis of heart disease through neural networks ensembles", Expert Systems with Applications, vol. 36, no. 4, (2004), pp. 7675-7680.

[3] A. K. Jain, K. Nandakumar, and A. Ross, “Score normalization in multimodal biometric systems," Pattern Recognition , 2005.

[4] Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[5] Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.

[6] University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.

[7] University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.

[8] V.A. Medical Center, Long Beach and Cleveland Clinic Foundation:Robert Detrano, M.D., Ph.D.

[9] D. Chaki, A. Das and M. I. Zaber, “A comparison of three discrete methods for classification of heart disease data”, Bangladesh J. Sci. Ind. Res. 50(4), 293-296, 2015.

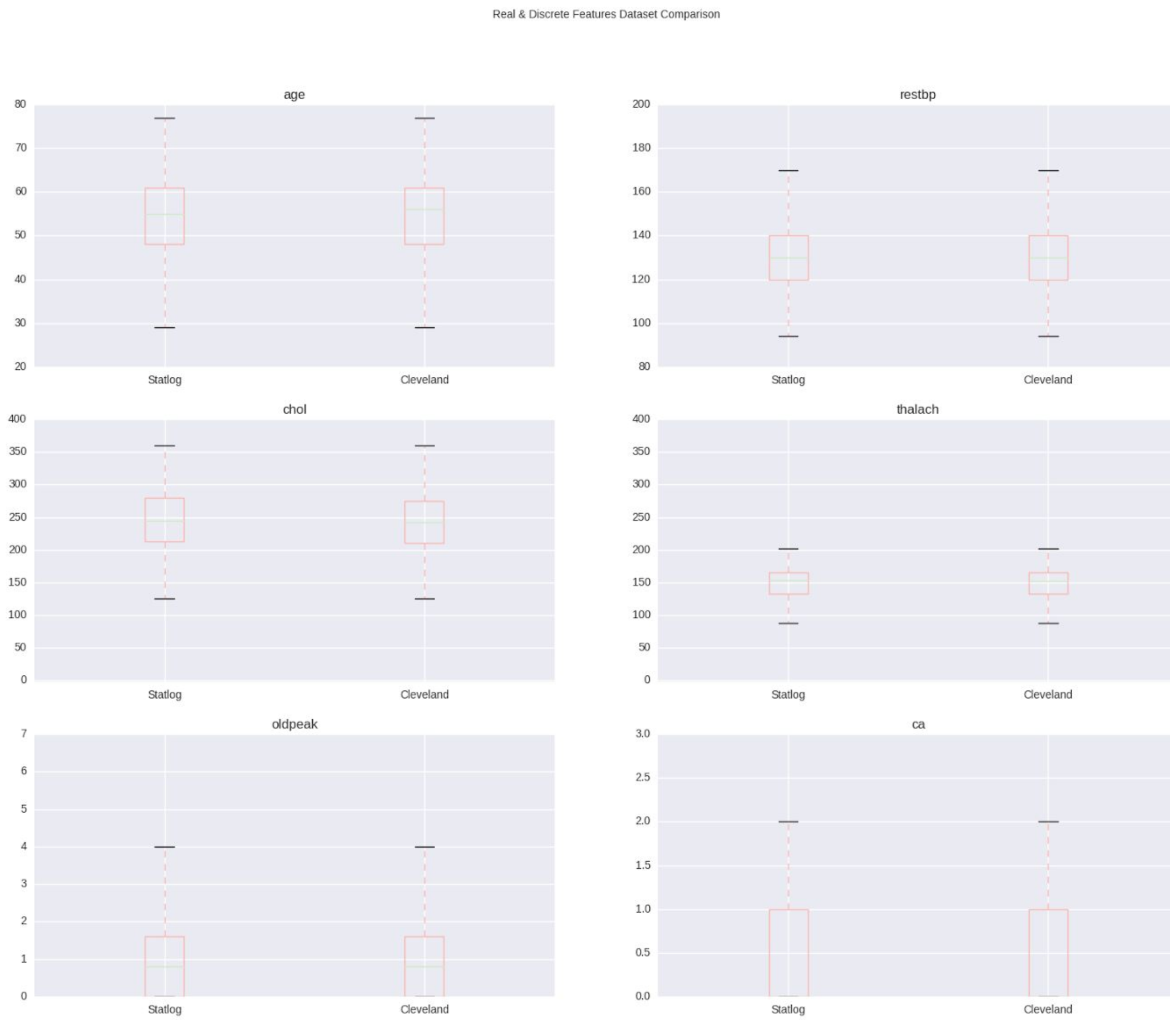
[10] Frank Rosenblatt. “The perceptron: A probabilistic model for information storage and organization in the brain.” In: Psychological review 65.6 (1958), p. 386.

[11] Shuxiang Xu and Ling Chen. “A novel approach for determining the optimal number of hidden layer neurons for FNN’s and its application in data mining”. In: (2008).

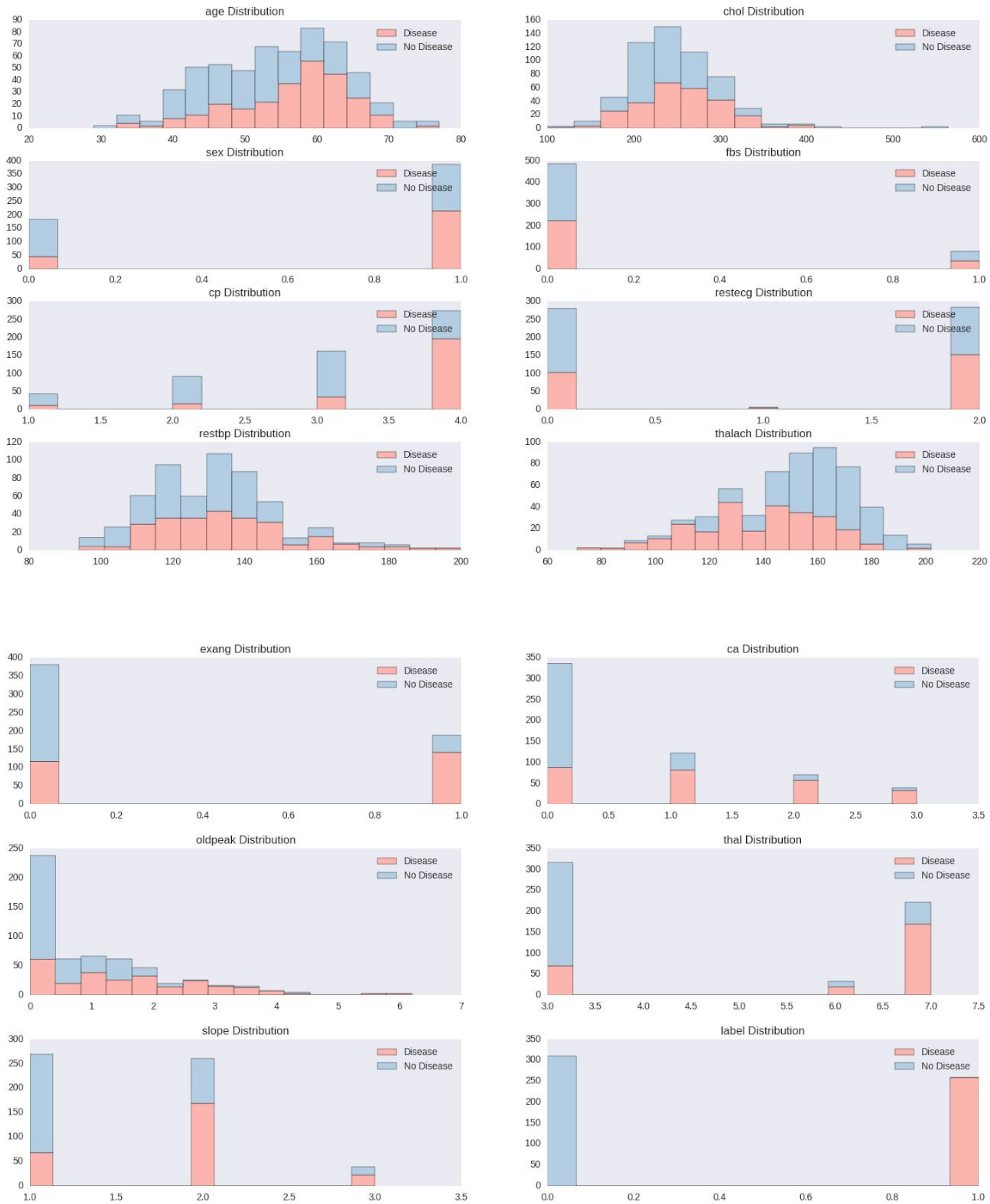
[12] S. Geman, E. Bienenstock, and R. Doursat. “Neural Networks and the Bias/Variance Dilemma”. In: Neural Computation (1992), pp. 4, 1–58.

Appendix

Feature histogram comparison of Cleveland and Statlog dataset.



Numeric features comparison between Cleveland and Statlog datasets.



Feature histograms comparing presence and absence of disease for the conjunction of Cleveland and Statlog datasets.