**Data Mining and Analysis for Managers**

**(MGMT 635 – Group Assignment #2)**

**Pima Indian Diabetes Database**

* **Diabetes diagnosis using classification algorithms**

July 27th, 2018



**Group #4**

**Group Members: Bikalpa Pahari (bp359)**

**Kalpit Khamar (kkk0482)**

**Mona Nandy (mrd2)**

**Sahana Umashankar (su83)**

**Instructor: Dantong Yu and Shibo Yao**

Table of Contents

[Introduction 3](#_Toc520457299)

[Description of Data 3](#_Toc520457300)

[Data Mining Methods 5](#_Toc520457301)

[I. Neural Networks 5](#_Toc520457302)

[II. Decision Tree 11](#_Toc520457303)

[III. Logistic Regression 16](#_Toc520457304)

[Testing and Analysis comparing all three algorithms 1](#_Toc520457305)9

[Weka Tool Results 2](#_Toc520457306)1

[Result and conclusion 2](#_Toc520457307)1

[References: 2](#_Toc520457308)2

# Introduction

Diabetes is metabolic conditions in which body can’t produce enough insulin or the body’s cells do not respond to insulin properly. Effects of diabetes have a more fatal and serious impact on women than on men. The impact of diabetes on women is unique because it can affect both mothers and their unborn children. Heart-attacks, miscarriages, kidney failure, baby born with birth defects are some of the implications of this disease on women.

This report focuses on diabetes recorded in pregnant women. In this paper, we have used data mining algorithms on a pre-existing diabetes dataset to make prediction of whether diabetes is diagnosed in a patient or not. We used a structured approach, CRISP-DM from understanding the problem to testing and evaluation. The algorithms used are: neural networks, decision tree and logistic regression. Results from these three algorithms have been compared and presented. To confirm the best accuracy, two tools have been used: MATLAB and Weka.

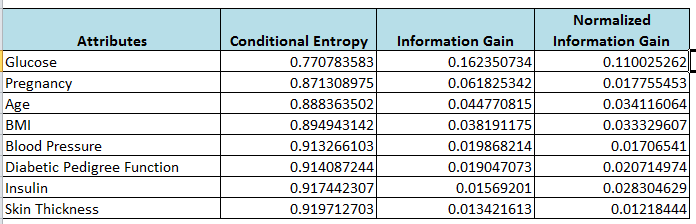
# Description of Data

For our project, we have used a diabetes dataset, which originally comes from National Institute of Diabetes and Digestive and Kidney Diseases. The dataset is collected from Kaggle UCI ML Repository (<https://www.kaggle.com/uciml/pima-indians-diabetes-database>). Based on the source, there were several constraints taken into account on selection of these instances from a larger database. Specifically, all patients are females and they are at least 21 years old of Pima Indian heritage. The data consists of 9 attributes and their descriptions are below (descriptions are taken from Kaggle):

* *Pregnancies*: Number of times pregnant
* *Glucose*: Plasma glucose concentration a 2 hours in an oral glucose tolerance test
* *BloodPressure*: Diastolic blood pressure (mm Hg)
* *SkinThickness*: Triceps skin fold thickness (mm)
* *Insulin*: 2-Hour serum insulin (mu U/ml)
* *BMI*: Body mass index (weight in kg/(height in m)^2)
* *DiabetesPedigreeFunction*: Diabetes pedigree function
* *Age*: Age (years)
* *Outcome*: Class variable (0 or 1)

The first 8 attributes are integers values describing each individual patient. The last attribute ‘Outcome’ is a target value in Boolean (0 or 1). The value of ‘0’ is given for the diagnosis of diabetes being positive for a patient. The value of ‘1’ is given for the diagnosis of non-diabetes for a patient. Among all 8, there are 1 categorical and 7 continuous attributes.

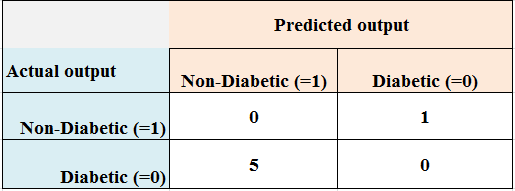
The dataset contains records for 768 female patients. Among these, 500 patients are diagnosed as being diabetic and rest 268 are diagnosed are non-diabetic. We will use information gain for each attributes to prioritize the attributes. Following is the summary of the conditional entropy, information gain, and normalized information gain for each eight attributes, in the order of highest to lowest information gain:



Detail of the calculations and summary of all algorithms are included in this workbook.



As part of the data understanding, we also learnt about the cost matrix guidelines provided in determining if our prediction is correct or incorrect, then how much weight can be assigned to the cost. In summary, if the predictions are correct (diabetic or non-diabetic), the count is ‘0’. If predictions for non-diabetic patients are ‘diabetic’, then a count of 1 is assigned to each prediction. And the most risky scenarios, where predictions for ‘diabetic’ patients are ‘non-diabetic’, then a count of 5 are assigned to those predictions. Following is the summary of the cost matrix for our project:



The total count of the cost points is calculated for each neural network trials. The lower the total cost points, the better our neural network would be. For our project, we divided dataset into training set (80%) and testing set (20%). Training dataset has 614 records and we held 154 records for test dataset for the evaluation of our model.

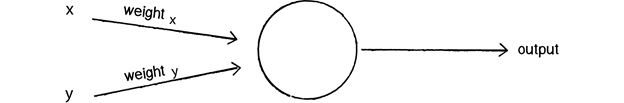
# Data Mining Methods

In this project, our main goal was to evaluate different data mining methods to find the best accuracy. We used two tools MATLAB and Weka to predict the outcome and evaluate the accuracy. We also used three different data mining methods using both tools: neural networks, decision tree and logistic regression. Each of these methods is described below in general with the brief process description, results and evaluations.

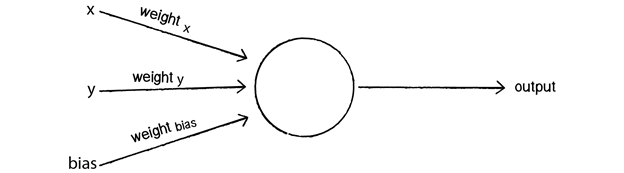
## Neural Networks

Neural Network algorithm is an implementation of the popular and adaptable neural network architecture for machine learning. Input data is passed through input layer and obtain result by activating the output layer. The algorithm works by testing each possible state of the input attribute against each possible state of the predictable attribute, and calculating probabilities for each combination based on the training data. You can use these probabilities for both classification or regression tasks, to predict an outcome based on some input attributes. A neural network can also be used for association analysis. In order to get the accurate result from neural network, the model can be trained and tested several times by changing the number of layers and neurons.

Neural network method learns information from the data by calculating and modifying the weights that are associated with neural networks.



We can see how there are two inputs (x and y), a weight for each input (weight x and weight y), as well as a processing neuron that generates the output.



0 \* weight for x = 0

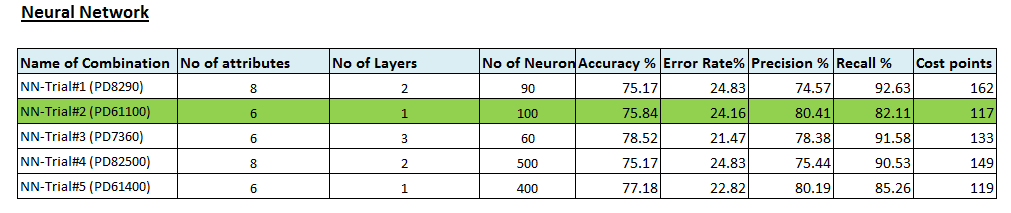
0 \* weight for y = 0

1 \* weight for bias = weight for bias

Why we chose Neural Network:

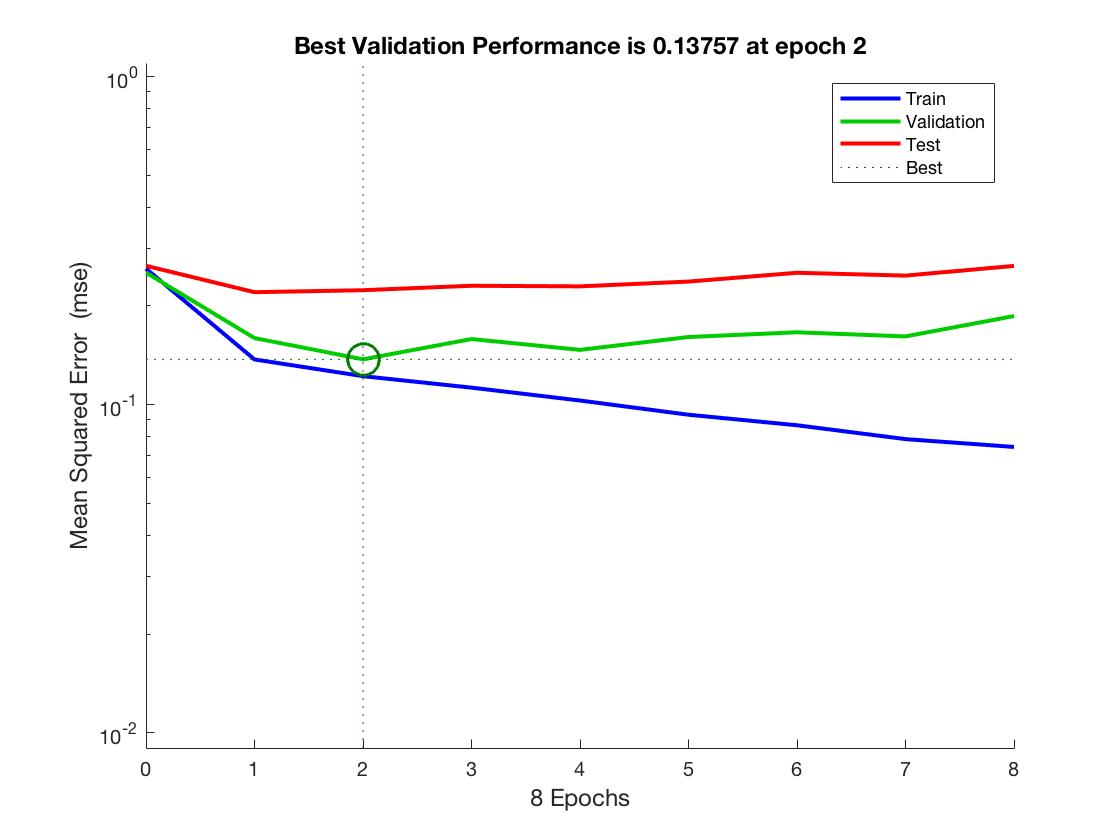
* A neural network model supports regression, association, and classification analysis, Therefore, the meaning of each prediction might be different.
* We can also query the model itself, to review the correlations that were found and retrieve related statistics.
* Neural Network is nonlinear model that is easy to use and understand compared to statistical methods.
* Neural Network with Back propagation (BP) learning algorithm is widely used in solving various classifications and forecasting problems.

Using neural network in MATLAB, we tried 5 different parameter combinations: number of attributes, number of layers and number of neurons as shown in the table below. The attribute selection was mostly based on the value of information gain (the higher the value, the higher the priority of that attribute). Besides number of attributes, we experimented with different number of layers and number of neurons to see which combination generates the output with best performance.

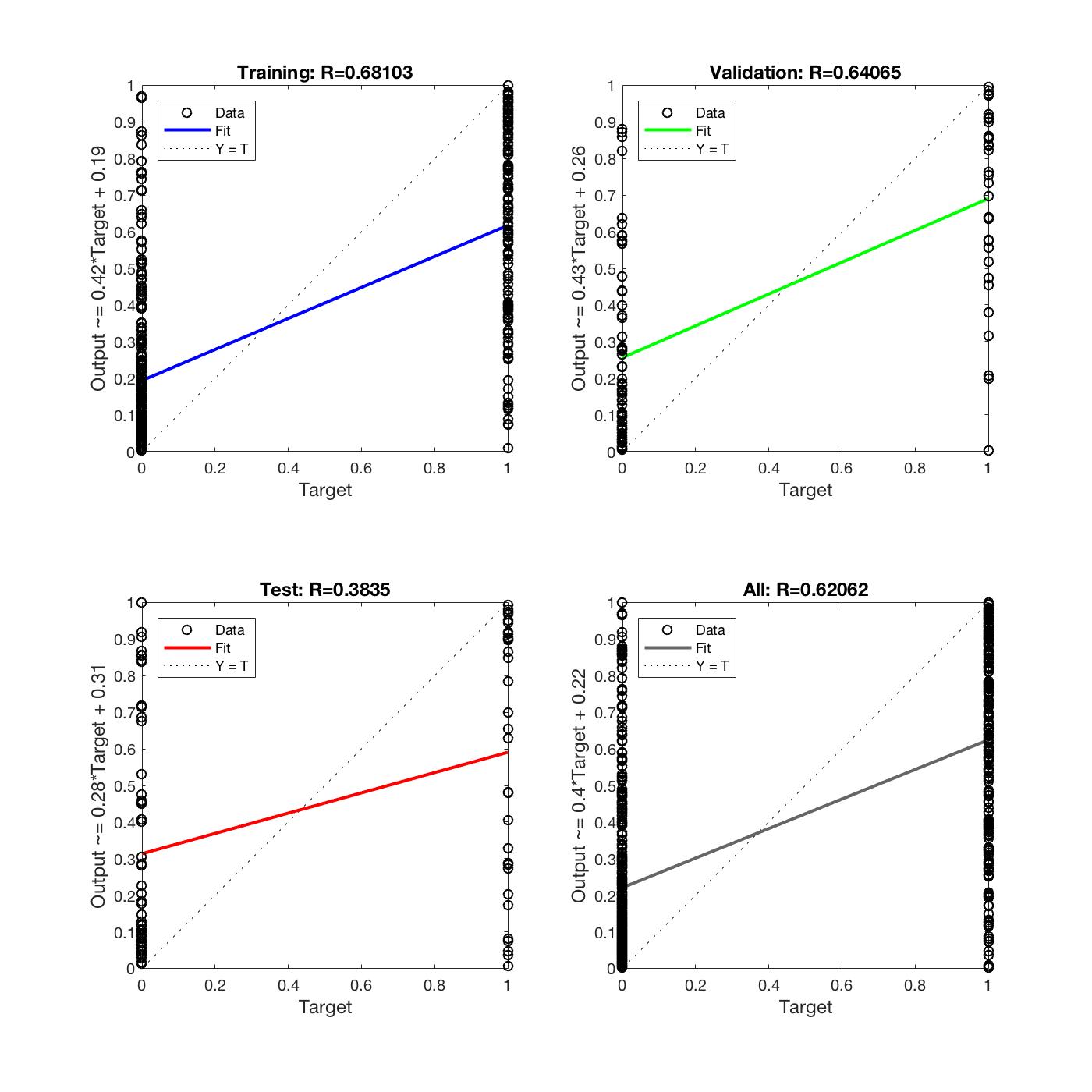


This table contains the combinations of each trial as well as the performance and calculated results which are discussed below.

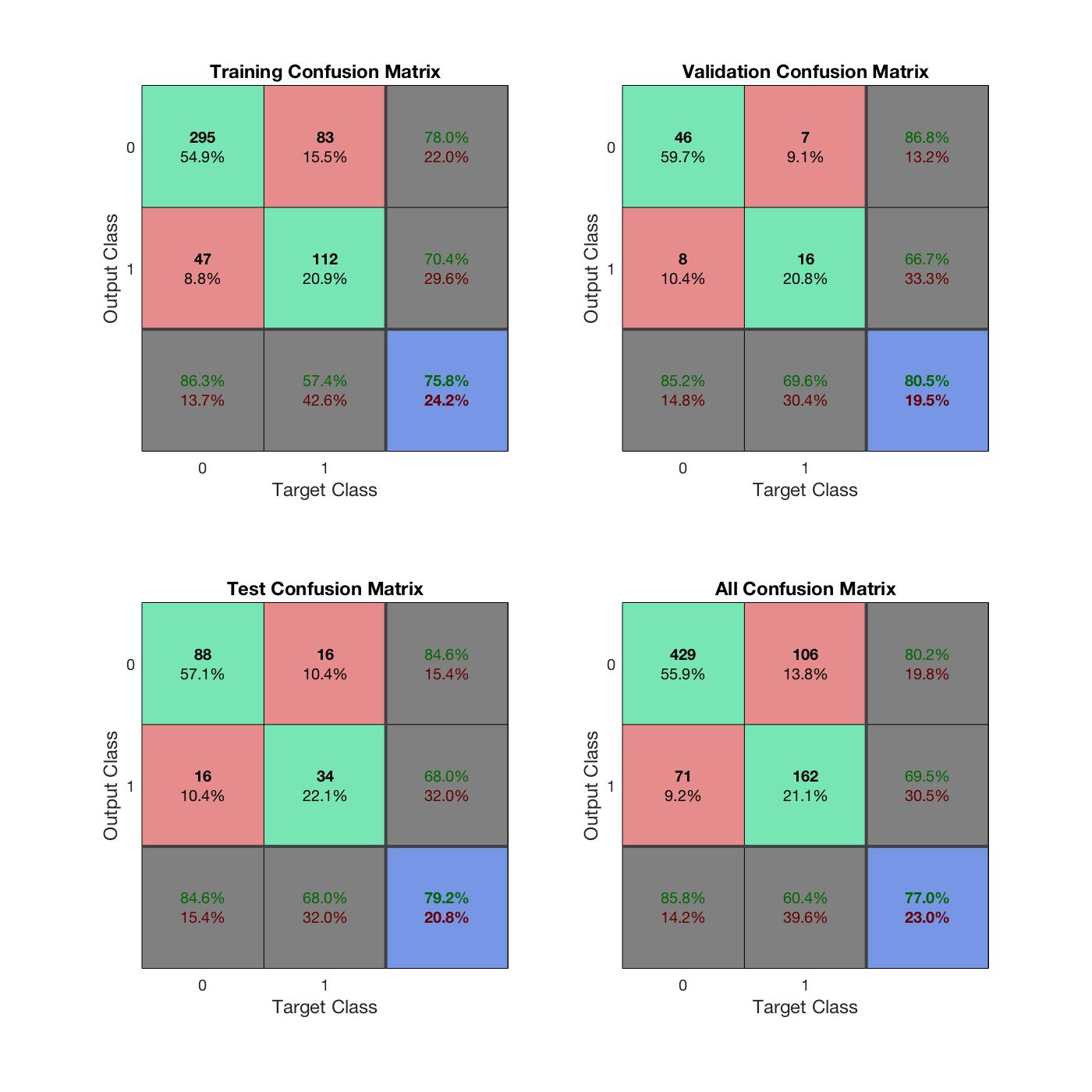
For our best neural network, we looked at the performance and validation graph as well. This is our best performance from epoch with the lowest validation error.



The second graph is the regression model created after training the data. The blue line shows the declining error on training data. The green line indicates the validation error and training stops when the validation error stops decreasing. The red line indicates the error on the test data. It basically says how well this neural network will generalize to new data outside of this given dataset.

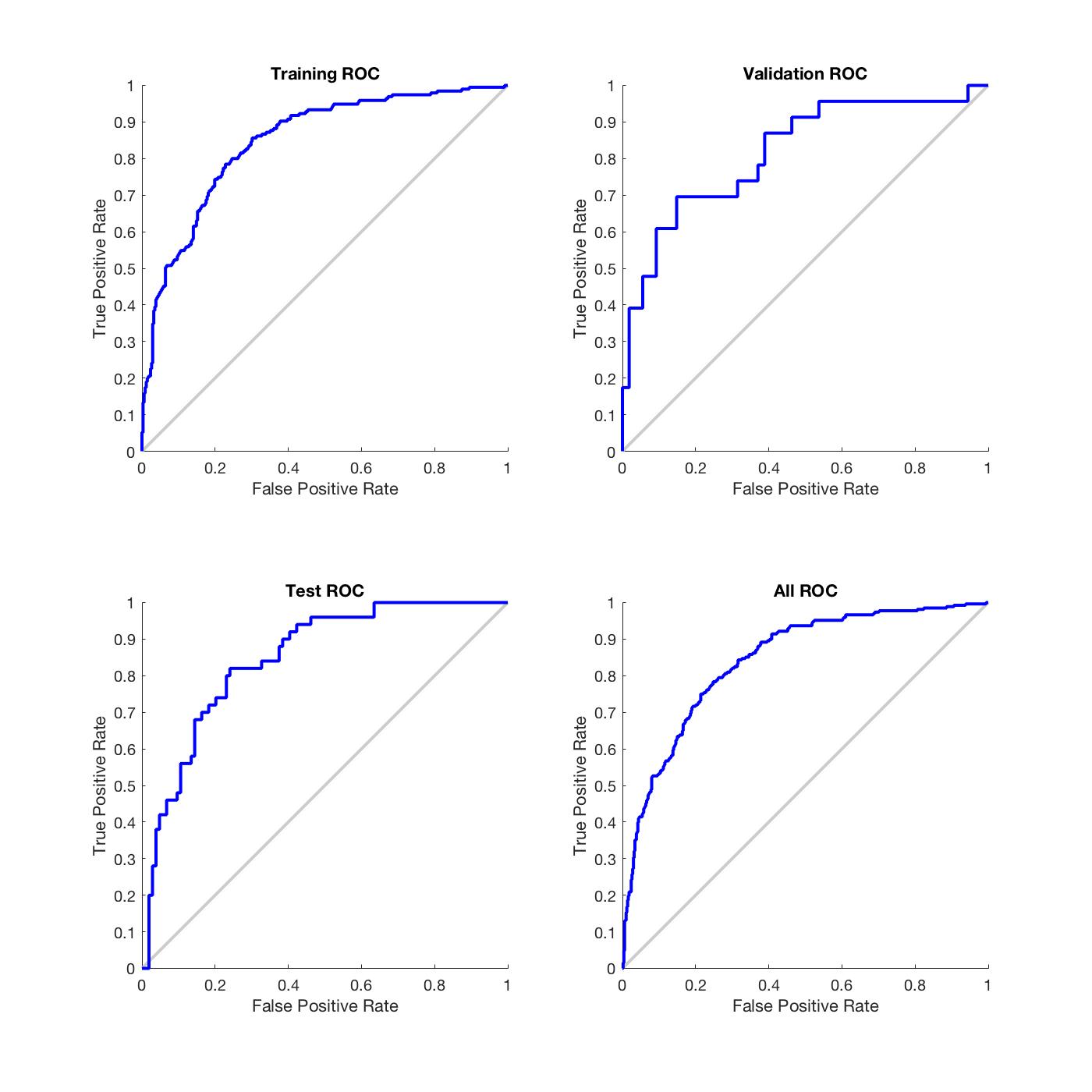


Classification of non-diabetic and diabetic patients of Diabetes dataset is shown in confusion matrix using the neural network pattern recognition tool (nprtool).



Proper classifications of patients are shown in green squares and improper classifications of patients are shown in red squares.  Total percentages of proper and improper classifications are shown in blue square.

The Receiver operating characteristic plot shows the percentage of True positive rate we get as a function of how many false positive rates we are willing to accept.



The accuracy % and confusion matrix values were minimally different between default and calculated values. Hence we conclude that the neural network algorithm fits the dataset.

## Decision Tree

Decision Tree belongs to the category of supervised learning algorithms. It is used to create a training model which can be used to predict class or label of target variables by learning decision rules inferred from training data. Decision Tree tries to solve the problem using a tree representation. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. In a decision tree

* An internal node represents an attribute
* Leaf node represents the class label
* Branch represents an outcome of the test

Decision tree algorithms usually work top-down, by choosing an attribute at each step that best splits the set of items. Different algorithms use different metrics for measuring “best”. These general measure the homogeneity of the target variable within the subsets. The popular attribute selection measures are:

* Information gain
* Gini index.

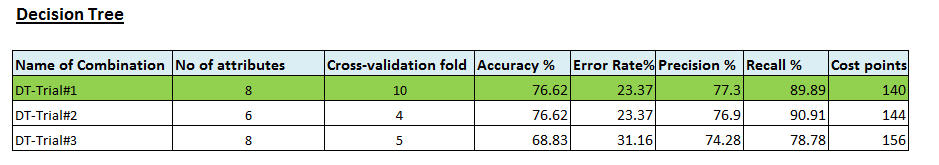
Suppose there are two classes P (positive) and N (negative) where P contains p training examples and N contains n training examples

*I(p, n) = -pp+nlog2pp+n-np+nlog2np+n*

Assume that using attribute A, a set S will be partitioned into subsets {S1, S2, …, Sv}. If Si contains pi examples of P and ni examples of N, the entropy, or the expected information needed to classify objects in all subtrees Si is

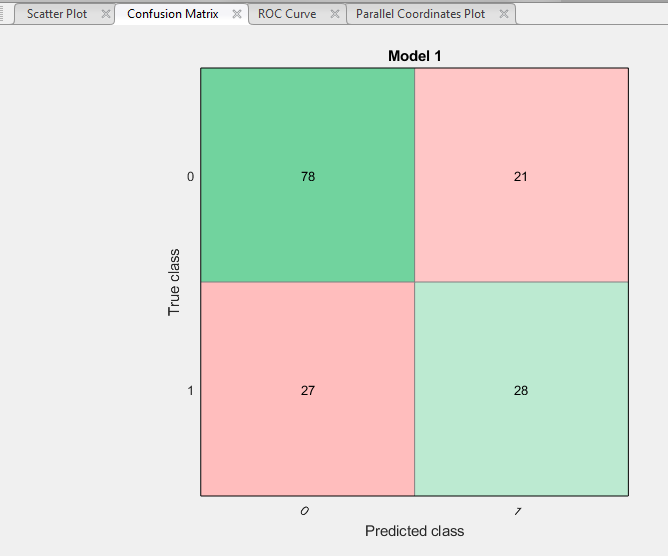
*E(A) = ∑pi+nip+n I((pi, ni)vi=1 Gain(A)=I(p, n) – E(A)*

Using decision tree in MATLAB, we tried 3 trials of different parameter combinations: number of attributes and cross validation fold as shown in the table below. The attribute selection was based on the value of information gain (the higher the value, the higher the priority of that attribute). Besides number of attributes, we experimented with cross validation folds. This method provides a good estimate of the prediction accuracy of the final model trained with all the data. It requires many fits but it uses all the data efficiently. This method is recommended for small data sets.



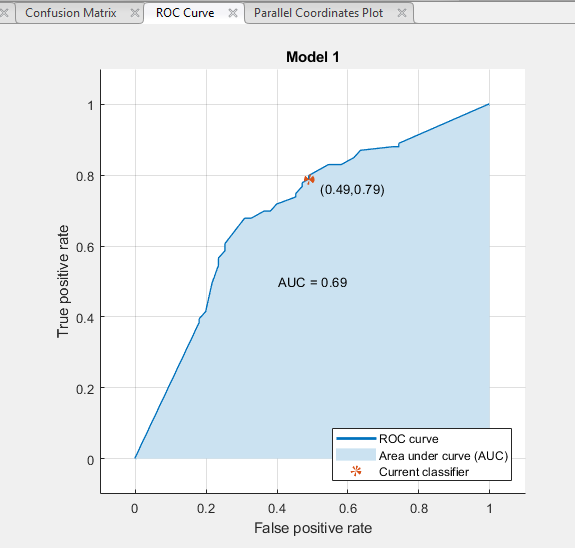
This table contains the combinations of each trial as well as the performance and calculated results which we is discussed below.

For our best decision tree model, we looked at the performance and validation graph as well. Classification of non-diabetic and diabetic patients of Diabetes dataset is shown in confusion matrix using the decision tree classification learner app. Following is the default confusion matrix for the best trial for decision tree algorithm in MATLAB.



After training the model, on Classification Learner tab, confusion matrix can be generated. This matrix helps us identify the areas where the classifier has performed poorly. Rows in the matrix represent true class and columns show the predicted class. The diagonal cells show where the true class and predicted class match. If these cells are green, the classifier has performed well and classified observations of this true class correctly.

The Receiver operating characteristic plot shows the percentage of True positive rate we get as a function of how many false positive rates we are willing to accept. Following is the default ROC curve for the best trial for decision tree algorithm in MATLAB.

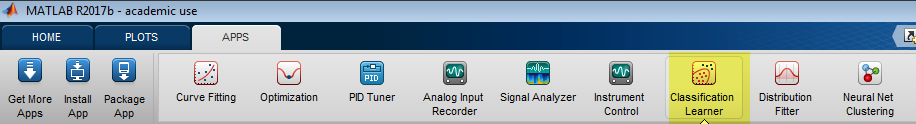


The accuracy % and confusion matrix values were significantly different between default and calculated values. We cannot conclude that decision tree algorithm is stable for this dataset.

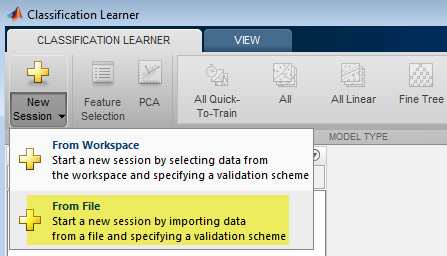
**Decision Tree and Logistic Regression Algorithm – Process step by step**

Both decision tree and logistic regression algorithms have similar steps to train data and test data by generating the predicted output. For our project, we divided dataset into training set and testing set. Training dataset has 614 records and we held 154 records for test dataset for the evaluation of our model. In the trial displayed below, we will train the model on 614 data records and use the trained model on 154 records to check the accuracy.

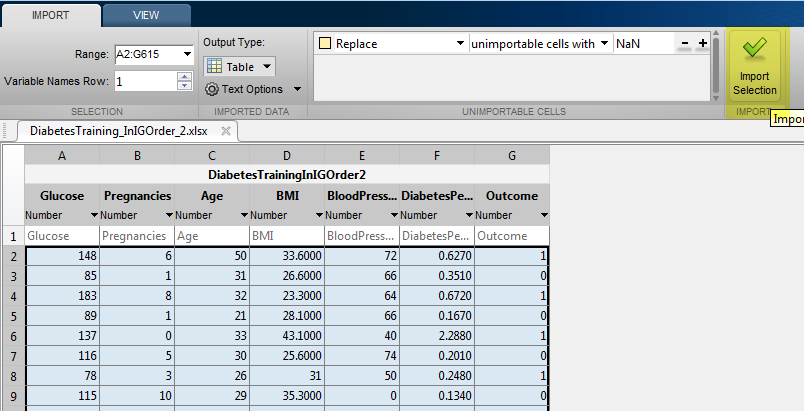
* In MATLAB, first we selected the app ‘Classification Learner’



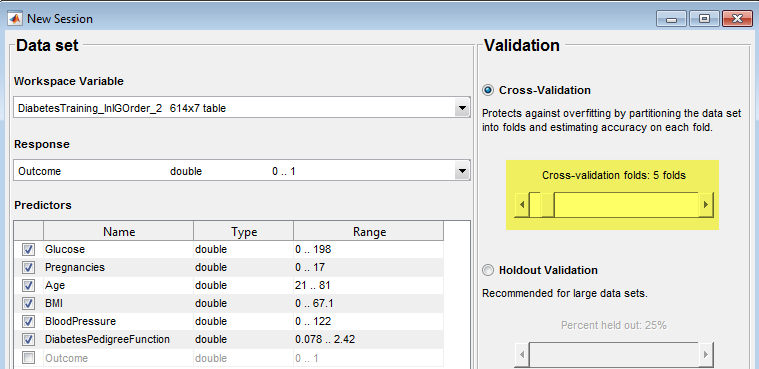
* In classification learner, start a new session and select ‘From file’ option to select the training dataset to import



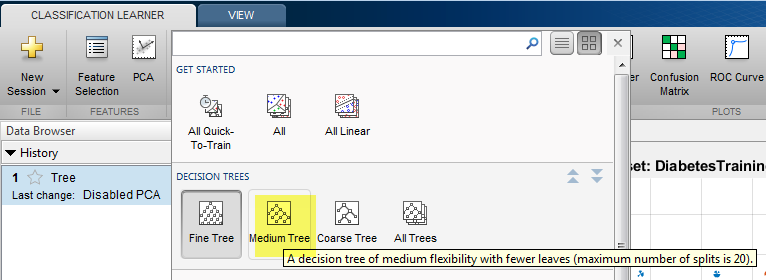
* After selecting a training dataset as shown below, select ‘Import Selection’. This option imports the selected dataset into the MATAB workspace



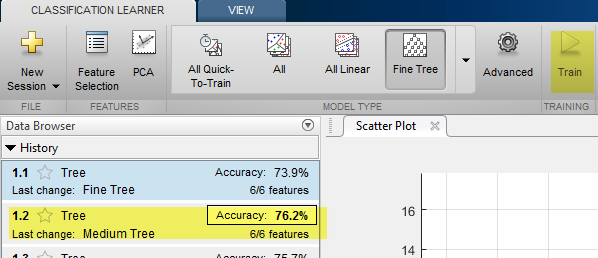
* For parameter selection, make sure that only attributes are selected as predictors and target attributes are unselected in the training process. We can also change the cross-validation fold by moving the slider.



* On starting new session, we can select the type of decision tree, we plan to train. In following screenshot, Medium Tree has fewer leaves with maximum number of splits is 20.



* Once you click on ‘Train’, it will train each model selected and provide the accuracy %.



* Evaluating the training model can be done by using the following code on testing dataset, where ‘TreeMedium01’ is the trained model name and ‘DiabetesTestInIGOrder’ is the testing dataset. The predicted values are provided as Boolean.

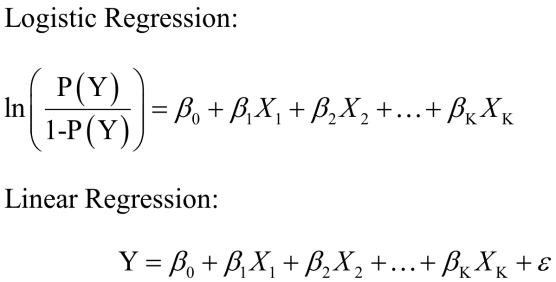
>> yfit = TreeMedium01.predictFcn(DiabetesTestInIGOrder)

As mentioned earlier, similar step by step process is used for Logistic Regression.

Attaching Matlab files trial of Decision tree trials 

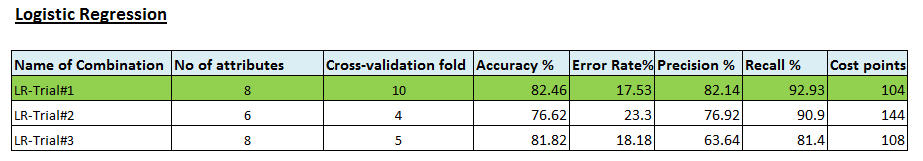
## Logistic Regression

Logistic regression extends the idea of linear regression to the situation where the outcome is situational. It is mostly used in structured model that are useful to explain or predict and mainly the focus is on binary classification. When you run a linear regression with a binary response variable to predict and obtain an output, the coefficient interpretation does not make sense. One of the assumptions of the linear regression is that there is a linear relationship between X and Y, and when Y is categorical it cannot have linear relationship with X, using the logarithmic transformation, we make that relationship appear linear. The below equation shows the difference between logistic and linear regression.



The logistic regression assumes that the outputs must be discrete.

For this method, we did three trials with different number of attributes and different cross-validation fold to improve the accuracy. Here is the summary of logistic regression trials and results.



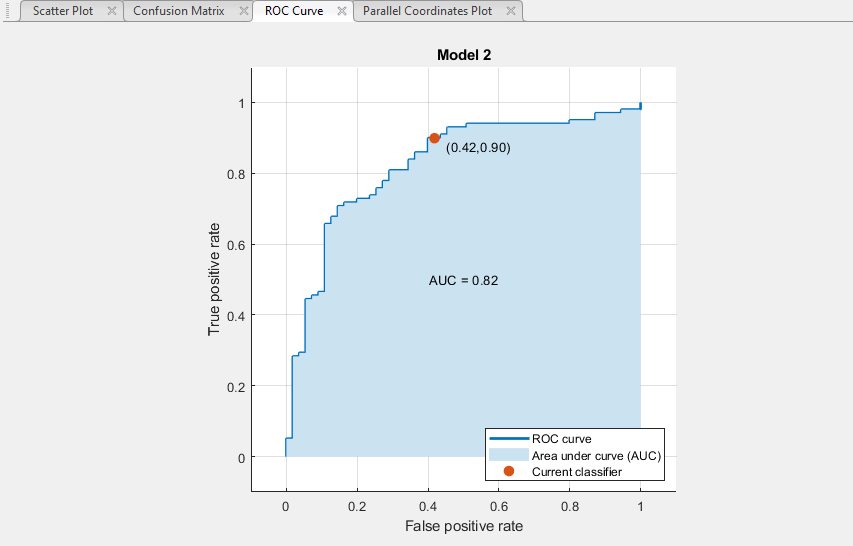
This table contains the combinations of each trial as well as the performance and calculated results which we is discussed below.

For our best logistic regression model, we looked at the performance and validation graph as well. Classification of non-diabetic and diabetic patients of Diabetes dataset is shown in confusion matrix using the decision tree classification learner app.



After training the model, on Classification Learner tab, confusion matrix can be generated. This matrix helps us identify the areas where the classifier has performed poorly. Rows in the matrix represent true class and columns show the predicted class. The diagonal cells show where the true class and predicted class match. If these cells are green, the classifier has performed well and classified observations of this true class correctly.

The Receiver operating characteristic plot shows the percentage of True positive rate we get as a function of how many false positive rates we are willing to accept.



The accuracy % and confusion matrix values were matching different between default and calculated values. We conclude that logistic regression algorithm is stable for this dataset.

Attaching Matlab files for three trial of Logistic regression

****

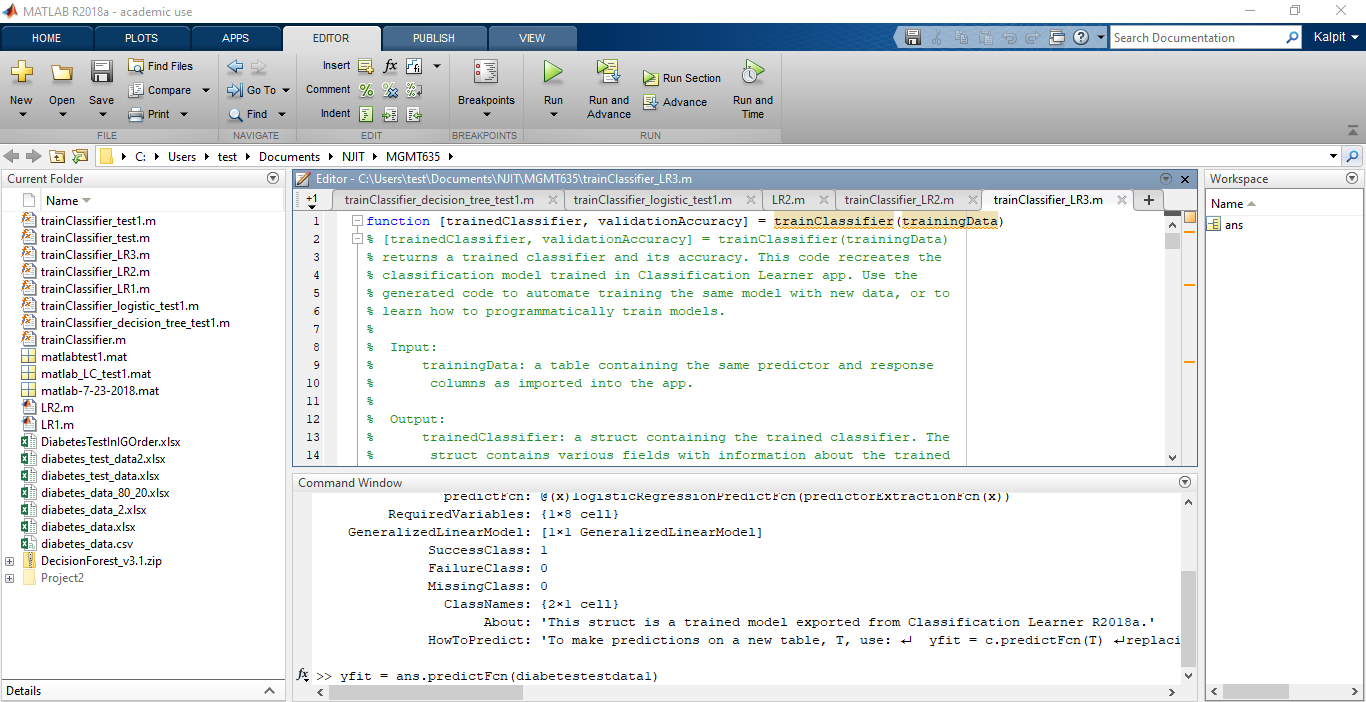
trainClassifier\_LR1.m contain 8 attributes: Glucose, Pregnancy, Age, BMI, Blood Pressure, Diabetic Pedigree Function, Insulin, Skin Thickness. Cross Validation Fold 10

trainClassifier\_LR2.m contain 6 attributes: Glucose, Pregnancy, Age, BMI, Blood Pressure, Diabetic Pedigree Function. Cross Validation Fold 4

trainClassifier\_LR3.m contain 8 attributes Glucose, Pregnancy, Age, BMI, Blood Pressure, Diabetic Pedigree Function, Insulin, Skin Thickness. Cross Validation Fold 5

Command to find predicted value for each modified Matlab file is specify here for two dataset instance: yfit = ans.predictFcn(diabetestestdata1) and yfit = ans.predictFcn(diabetestestdata2)

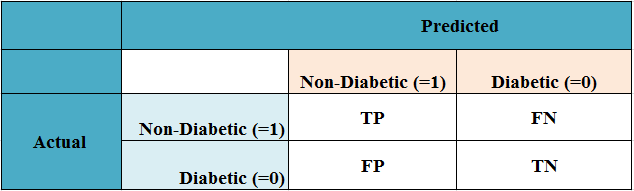
All three Matlab files use test dataset attached here 



# Testing and Analysis comparing all three algorithms

As a phase of CRISP-DM, we evaluated the model for accuracy and generality. This is a critical and challenging step. This is to test how good is our model and the parameters used. Data analysts need to understand the data mining objectives, and how it will all fit in with the business objectives. In order to properly analyze the results, confusion matrix is used.

**Confusion matrix:** The confusion matrix is a two by two table that contains four outcomes produced by a binary classifier. Various measures, such as error-rate, accuracy, sensitivity, and precision, are derived from the confusion matrix.



**Fig: Confusion Matrix**

**Error-rate:** Error rate (ERR) is calculated as the number of all incorrect predictions divided by the total number of the dataset. The best error rate is 0.0, whereas the worst is 1.0.

**Error rate: (FP+FN)/ (TP+FN+FP+TN)**

**Accuracy:** Accuracy (ACC) is calculated as the number of all correct predictions divided by the total number of the dataset. The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – ERR.

**Accuracy: (TP+TN)/ (TP+FN+FP+TN)**

**Precision:** Precision (PREC) is calculated as the number of correct positive predictions divided by the total number of positive predictions. It is also called positive predictive value (PPV). The best precision is 1.0, whereas the worst is 0.0.

**Precision : TP/ (TP+FP)**

**Recall:** Recall or Sensitivity (SN) is calculated as the number of correct positive predictions divided by the total number of positives. It is also called Sensitivity (SN) or true positive rate (TPR). The best sensitivity is 1.0, whereas the worst is 0.0.

**Recall/Sensitivity: TP/ (TP+FN)**

For our project, using all three methods: neural network, decision tree and logistic regression in MATLAB, we simulated the prediction model on testing “Holdout Input” dataset. With each trial, we repeated this step to capture the predicted output. We rounded the predicted output values (154) into 0’s and 1’s for neural network and for other two methods, decision tree and logistic regression, the output were Booleans. We added predicted results and actual results (which are from given dataset) side by side to conclude the classification type. Here are the classification guidelines that we used:

|  |  |
| --- | --- |
| **Classification Type** | **(Predicted, Actual)** |
| True Positive (TP) | (0, 0) |
| True Negative (TN) | (1, 1) |
| False Positive (FP) | (0, 1) |
| False Negative (FN) | (1, 0) |

Based on the count of each classification type, we calculated **cost points, accuracy%, precision%, error rate% and recall%**. The worksheet contains the detail information about each trial (combinations) we analyzed. Our goal was to identify the best neural network with respective parameter combinations (attributes, layers and neurons). Here are the mathematical calculations we used to derive these performance indicators:

* + - Accuracy % = (TP + TN) \* 100 / (TP + TN + FN + FP)
    - Precision % = (TP \* 100) / (TP + FP)
    - Error Rate % = (FP + FN) \* 100 / (TP+TN+FN+FP)
    - Recall (sensitivity) % = TP \* 100 / (TP+FN)

where TP, TN, FP FN indicates the count of these classification type output. We summarized the result, which is described in result section.

# Weka Tool Results

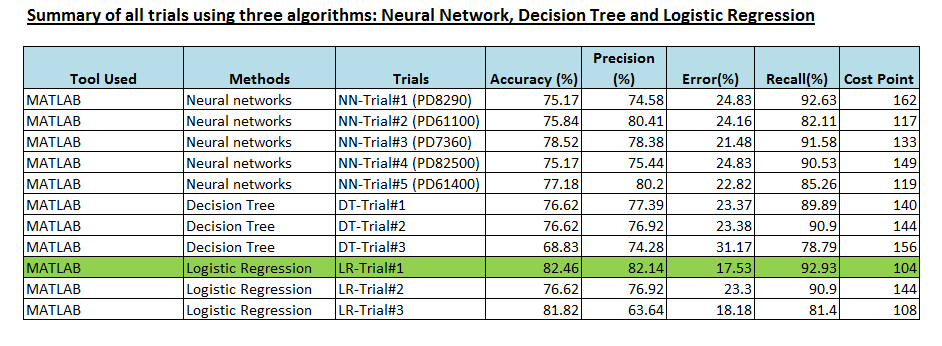
We implemented various algorithms in Weka tools to compare the results with the ones in MATLAB. We used the same dataset and achieved the results that were almost similar to the results from MATLAB. We have attached the results from WEKA tool with explanation below.



# Result and conclusion

Application of data mining algorithms on full scale implementation is cost effective, less error-prone and less time consuming. Application of data mining algorithms in health sector is gaining momentum. It requires more research and exploration. It has high potential to cater to diversifying needs in the health sector, mainly in the identification and prevention of various diseases such as diabetes.

In this project, we were presented with a classification problem of diagnosis of diabetic patients, especially PIMA Indian females. We have trained and tested the diabetic dataset with 80% training and 20% test data split. We were tasked to train and test Supervised learning models for diabetic prediction. The dataset consisted of all the female patients’ information with target value as Boolean (1=diabetic or 0=non-diabetic). We followed CRISP-DM methodology to complete this data mining project. We built our models using three different algorithms: Neural network, decision tree and logistic regression in MATLAB and WEKA tools. We did 5 trials of neural network, 3 trials of decision tree and 3 trials of logistic regression. We performed error based analysis and compared the performance between all three algorithms mentioned. We selected our best result based on the lowest cost point with optimal accuracy and precision rate. Thus, **logistic regression** (trial #1) algorithm has the best accuracy (82.46%) and cost point of 104, thus achieving the objective of this study.



# References:

<http://www.diabetes.org/living-with-diabetes/treatment-and-care/women/>

<https://www.kaggle.com/uciml/pima-indians-diabetes-database>

<http://www.saedsayad.com/decision_tree.htm>

<https://en.wikipedia.org/wiki/Decision_tree_learning>