**Detection of IoT botnet attacks**

====================

Table of Contents

[Overall Goal/Research Hypothesis 2](#_Toc24114453)

[Research Question 2](#_Toc24114454)

[Previous Contributions 2](#_Toc24114455)

[Preprocessing activities, Features Selection/Engineering 2](#_Toc24114456)

[Training Method 7](#_Toc24114457)

[Logistic Regression Analysis 7](#_Toc24114458)

[Decision Trees 8](#_Toc24114459)

[Random Forest 9](#_Toc24114460)

[Ensemble Method 10](#_Toc24114461)

[Model Execution Time 11](#_Toc24114462)

[Comparison Study 11](#_Toc24114463)

[Summary 11](#_Toc24114464)

[Reference 11](#_Toc24114465)

# Overall Goal/Research Hypothesis

For the final project paper. As the number of Internet of Things (IoT) devices being deployed worldwide has been increasing rapidly. And the traffic volume of IoT-based DDoS attacks has on the rise as well. The needs for detection of IoT botnet attacks and identifying compromised devices have become imperative for mitigating the risks associated with these attacks. In this paper, the goal is to find the anomalies in the dataset and create visualization to get better understanding. And use machine learning techniques to classify packets as either attack or normal.  For that, I have used BoTNet\_IoT data and it contains a total of 117 columns and 983 rows.

The dataset consists of a lot of valuable information. The dataset was created by capturing the raw network traffic data (in pcap format) using port mirroring on the switch through which the organizational traffic typically flows [1]. The dataset consists of the same set of 23 features from five-time windows of the most recent 100ms, 500ms, 1.5sec, 10sec, and 1min. The dataset provided was in CSV format.

# Research Question

1. Detect malicious traffic data using ML methods.
2. Understanding the use of spectral analysis and difference in signal wavelength

# Previous Contributions

There are two groups that have contributed in the past, the work I have referred to. The first group with a team of seven Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Dominik Breitenbacher, Asaf Shabtai, and Yuval Elovici from  Ben-Gurion University of the Negev and  Singapore University of Technology and Design. In that paper, they propose and empirically evaluate a novel network-based anomaly detection method which extracts behavior snapshots of the network and uses deep autoencoders to detect anomalous network traffic emanating from compromised IoT devices

And the second group with Nickolaos Koroniotis, Nour Moustafaa, Elena Sitnikova, Benjamin Turnbull from UNSW Sydney, UNSW Canberra and the University of South Australia. In that paper, the team has evaluated the reliability of the BoT-IoT dataset using different statistical and machine learning methods for forensic purposes compared with the existing datasets. This work provides the baseline for allowing botnet identification across IoT-specific networks.

# Preprocessing activities, Features Selection/Engineering

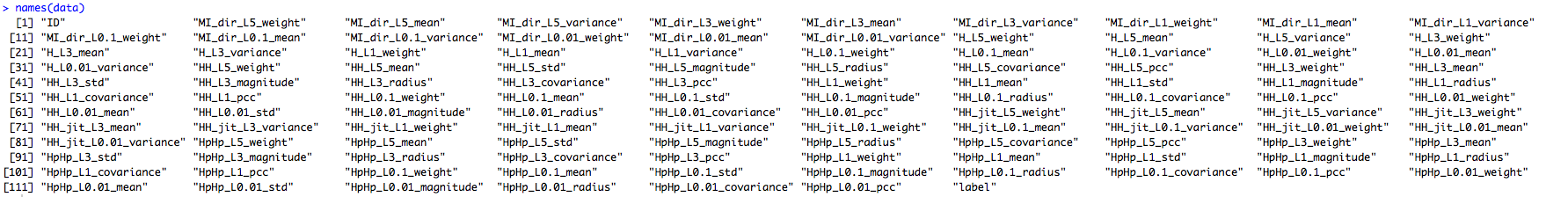
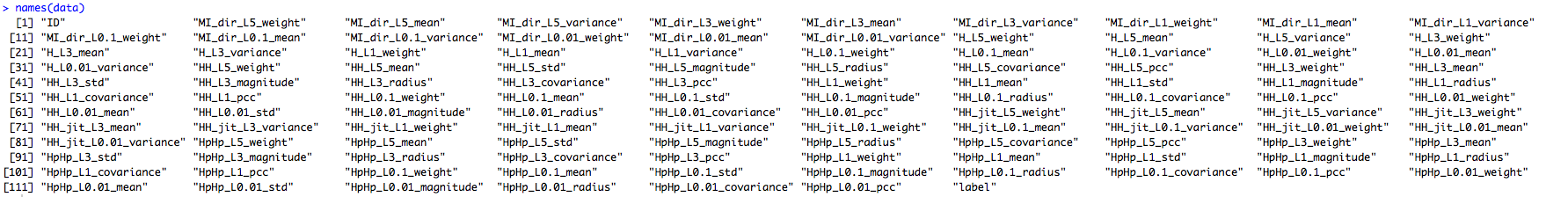
One of the research study goal is to analyze the data, detect and visualize the anomalies in the traffic pattern. The dataset contains a column called label which is binary, to differentiate the two types of traffic. One, normal traffic and the other one is attacked traffic. Hence this dataset requires a classification analysis. I have used R language, Microsoft Excel, and RStudio to complete this project.

I started off by removing all the previous data from the RStudio environment. And set the path where the data is located as a home directory.  And then the CSV data file was loaded. I used some of the following library to plot graphs and to explore insights.

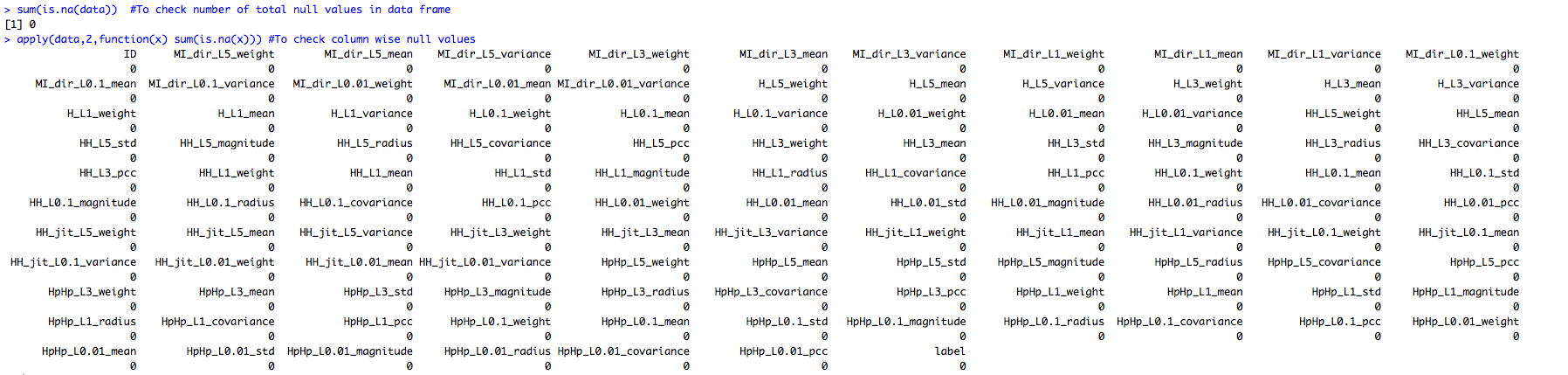
1. reshape2 (It makes it easy to transform data between wide and long formats)
2. ggplot2 (Creates elegant and complex plots)
3. corrplot (To visualize correlation matrix)
4. dplyr (For data manipulation)
5. MASS (Support functions and database for Venables)
6. clusterGeneration (For generating clusters)
7. logistf (Firth's Bias-Reduced Logistic Regression)
8. pscl ()
9. caret (classification and regression training)
10. arm (Data Analysis Using Regression and Multilevel/Hierarchical Models.)
11. standardize (For standardizing the function scale)

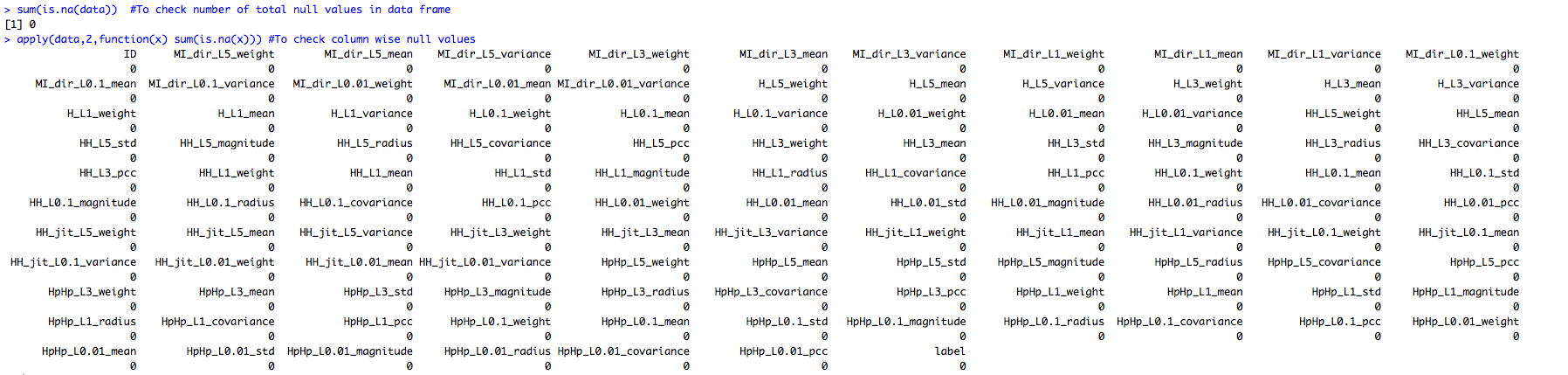
I started off with checking the num of columns and rows in the dataset. The dataset has 983 columns and 117 rows.



Next, I wanted to check the first 5 rows and the last 5 rows of the dataset to see if the data format was uniform. Dude to a large number of columns in the dataset I have decided to skip attaching the screenshot. The below image shows you the names of all the columns in the dataset. 

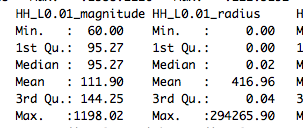
Next, I wanted to check for any missing values. The screenshot shows that the dataset is very clean with no missing values. I have checked the null values for the entire dataset and also double-checked it column-wise.

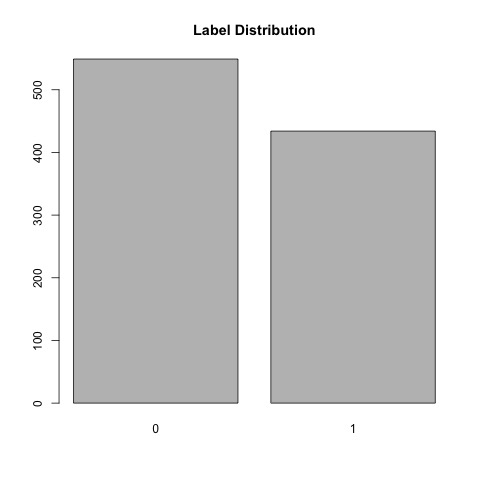




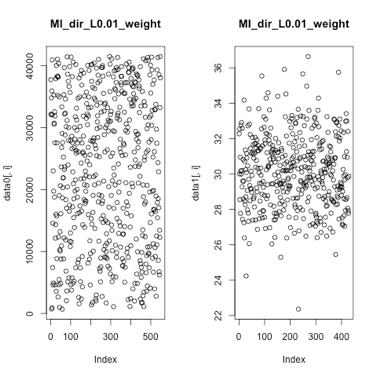
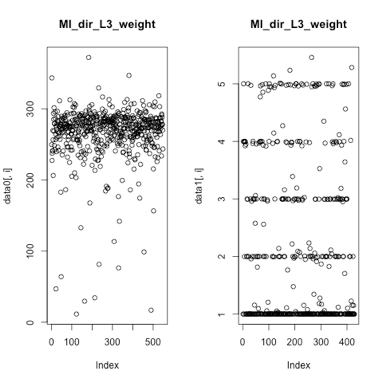
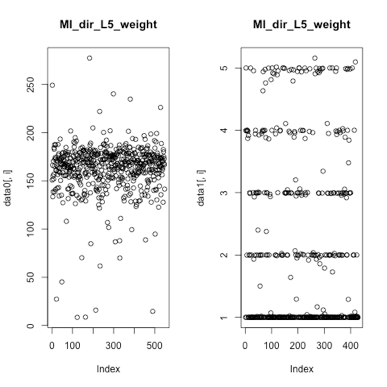
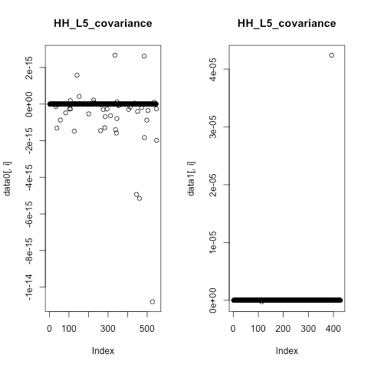
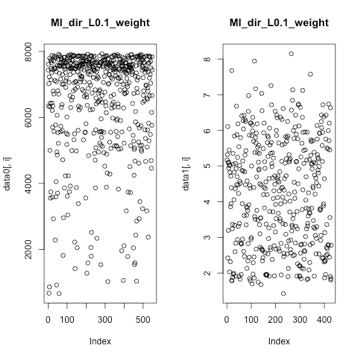
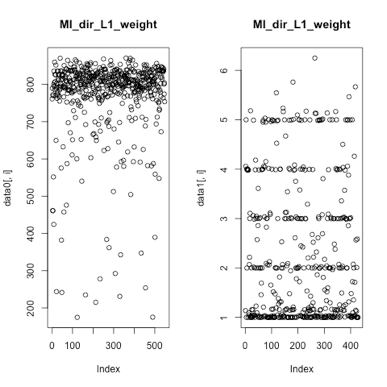


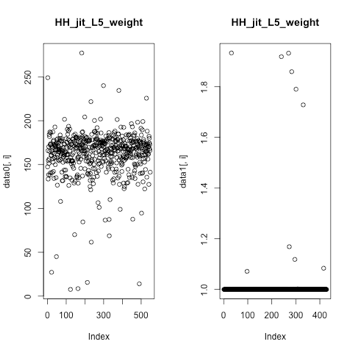
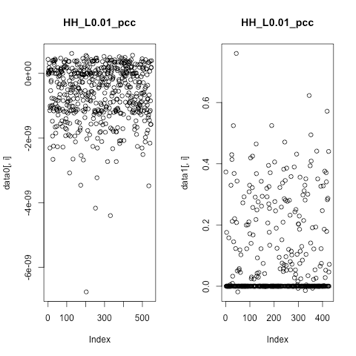
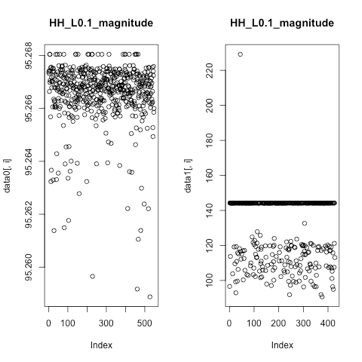
Then I checked for unique values in the ID columns to make sure there are no repeated rows. And there are 983 unique values which means there are no rows repeating. Then checked the structure of the dataset. Which did not give me any alarming insights. After that, I have summarized the numeric values to find more insights. Here I have noticed in quite a few columns the value between the 3rd Quartile and the Max value was very large. That means there are a potential outlier.



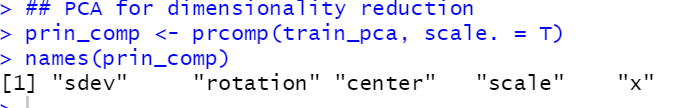
Since it is a classification analysis, I check for class imbalance in the label column by plotting the count of '0' and '1'. There is no class imbalance. 

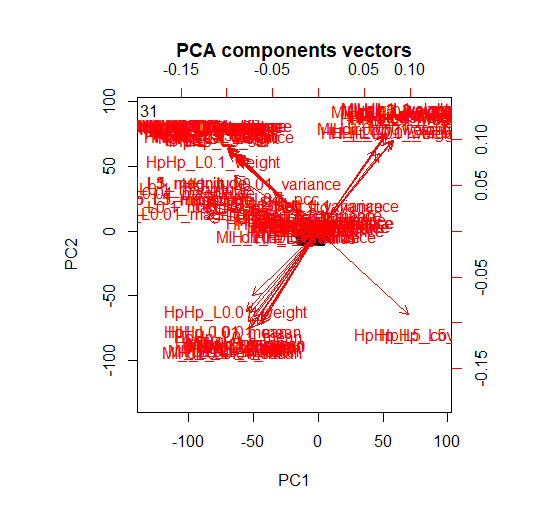
Now to find the anomaly pattern. I have created two new datasets by splitting the label column 0’s into one dataset and 1’s into another dataset. Then we plotted the traffic distribution for each column. As we have a many column. I have attached a few graphs below

****

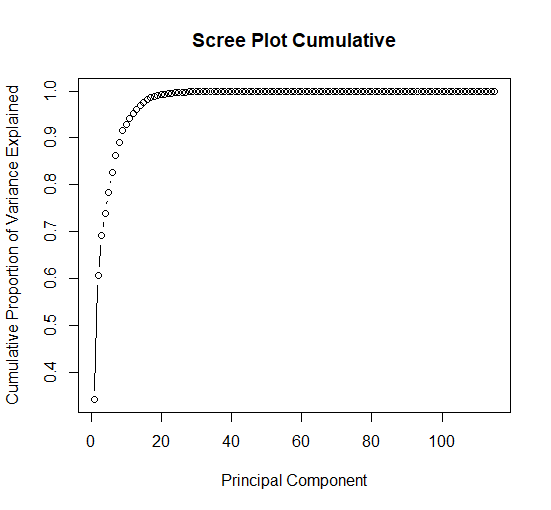
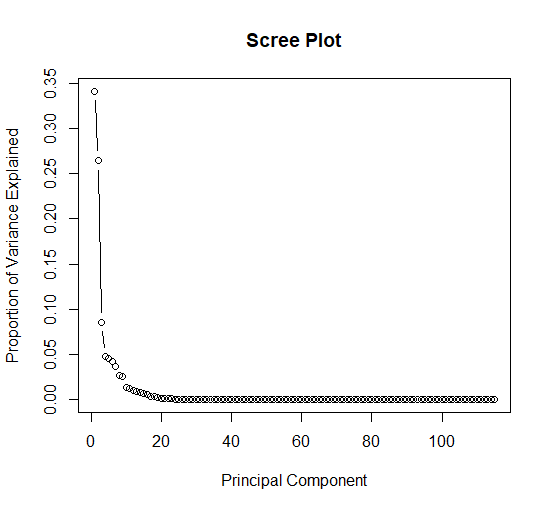


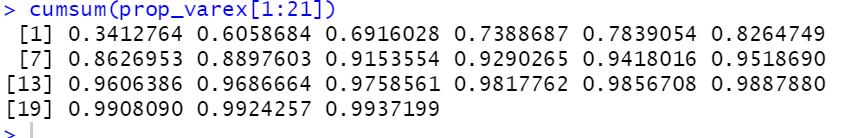
During my earlier analysis as we have doubted outliers in the dataset and also the quite a few columns with the same data which can cause high correlation and the model in predictions. Considering the number of columns in the dataset I have used Principal Component Analysis (PCA) which uses orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables. Next I have created a training set with principal components





Basis the PCA component standard deviation score and we plot the variance explained on scree plot to understand the amount of variance explained by the components.





As we see above the first 20 PCA components are explaining 99% of the variation, we pick up the first 20 PCA components. From this to prepare a dataset and continue into our training models.

# Training Method

The dataset contains a binomial outcome in column ‘label’ with more than one exploratory variable, logistic regression is a powerful statistical way of modeling. Along with Logistic regression I will be presenting other methods such as Decision Tree and Random Forest and an ensembled method. I am going to use 75% data for training and 25% data for testing the model

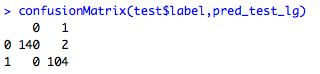
For modeling I have used the training data set that I have created using principal components. Took the first 20 PCA columns and added the target variable to the training dataset. Next, similarly I have converted the test dataset into PCA as well. Now I have my train and test datasets.

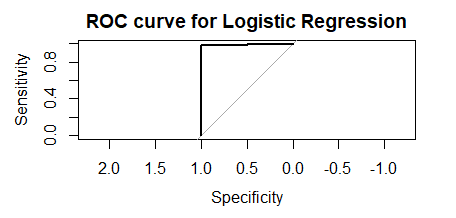
# Logistic Regression Analysis

Having my datasets ready. I have set seed to 1. And fitted the logistic regression model. Here I have used grid search with a parameter of (1,10,100,1000) to find the optimal hyperparameters of a model which results in the most ‘accurate’ predictions.



Once the Logistic model was built. I ran predictions for the test data. And you can see the results below. For the confusion matrix. The model has predicted 140 actual true positives as positives, 2 False Positives as true positives and predicted 104 Actual True Negatives as True Negatives. With an accuracy of 99.19%.





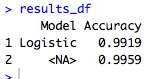
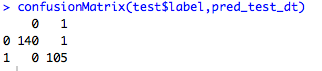
Since the accuracy is high. You can observe the ‘L’ shape in the ROC curve.

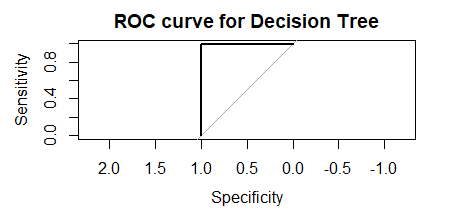
# Decision Trees

Next I used a Decision tree as it is good in predicting the target variable by learning simple decision rules inferred from the data features.



After fitting the decision tree model. And below are the confusion matrix and accuracy. The model has predicted 140 actual true positives as positives, 1 False Positives as true positives and predicted 105 Actual True Negatives as True Negatives. With an accuracy of 99.59%.





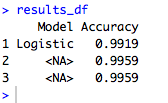
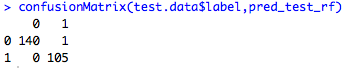
Since the accuracy is high. You can observe the ‘L’ shape in the ROC curve above.

# Random Forest

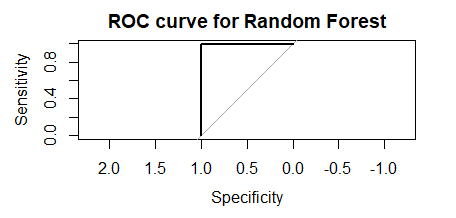
Next I have used Random Forest classifier. This model runs fast and maintains the accuracy for the large proportions of the data. Even if there are more trees, it won’t allow overfitting in the model and also can handle a large data set with higher dimensionality.



And the confusion matrix and the accuracy for the test data is shown below. The model has predicted 140 actual true positives as positives, 1 False Positives as true positives and predicted 105 Actual True Negatives as True Negatives. With an accuracy of 99.59%



Since the accuracy is high. You can observe the ‘L’ shape in the ROC curve below.

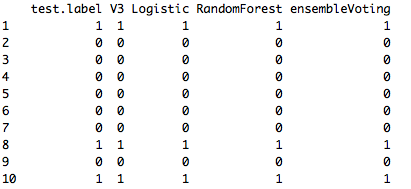


# Ensemble Method

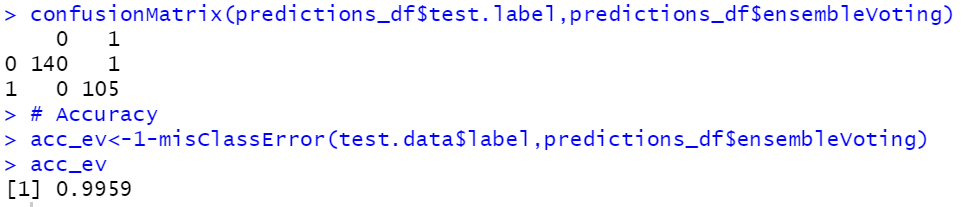
Ensemble is a method of combining the outputs of multiple algorithms, which are known as base learners. We do this to make a more robust system of predictions, where it reduces the misclassifications.

Bagging or Boosting ensemble methods would help in reducing the bias and variance respectively. However, in our case, the difference between train and test accuracies are very small hence did not see the need for applying these practices.

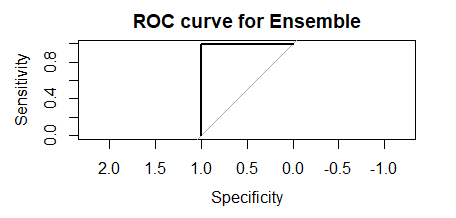
Hence, I have used Majority voting, which is used for classification problems, where we consider the prediction with maximum vote / recommendation from multiple models.



After running the model this method has predicted 140 actual true positives as positives, 1 False Positives as true positives and predicted 105 Actual True Negatives as True Negatives. With an accuracy of 99.59%



Since the accuracy is high. You can observe the ‘L’ shape in the ROC curve below.



# Comparison Study

My analysis results cannot really be compared with the previous research done by the other author. Because in the other research the author has taken just the normal traffic and trained the model used auto encoders whereas we trained the models with a classification data. Also, the other research the author has split the train and test data as 80% and 20% whereas I have split it into 75% and 25%. And the dataset size is very small when compared to the other study.

# Summary

BoTNet\_IoT data was analyzed using different visualizations and models to predict the difference in normal and attacked traffic. Analysis revealed the distribution for the data. Looking at the data there were outliers and high correlation in the data. Hence, we have used PCA is fix these issues and also to get the right composition for the model to learn. Mainly to reduce dimensionality. In order to predict the normal vs attacked traffic models were created using logistic, decision tree, random forest and ensembled methods. And the models could achieve above 98% accuracy. The analysis was done using RStudio IDE, R, Excel.

# Reference

1. <https://arxiv.org/pdf/1805.03409.pdf>
2. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-Iot Dataset
3. <https://en.wikipedia.org/wiki/Principal_component_analysis>
4. <https://www.analyticsvidhya.com/blog/2017/02/introduction-to-ensembling-along-with-implementation-in-r/>