Machine Learning for Cyber Security: Intrusion Detection

Springboard DSC:
Capstone Project 2
Matthew Wong

Introduction

- Cyber-attacks are increasing as more data is being processed
- Many companies are targeted
- We can use data science to detect future attacks

Overview

- This project will focus on using machine learning and data science on a cyber security problem: predicting network attacks.

- **Client**: Online company looking for a data science consultant to assist them with their web security problems.

- **Goal**: To help build a system for intrusion detection

- **Dataset**: The dataset was created by the IXIA PerfectStorm toll in the Cyber Range Lap of the Australian Centre for Cyber Security. Its a hybrid of real modern activities and synthetic contemporary attack behaviors.

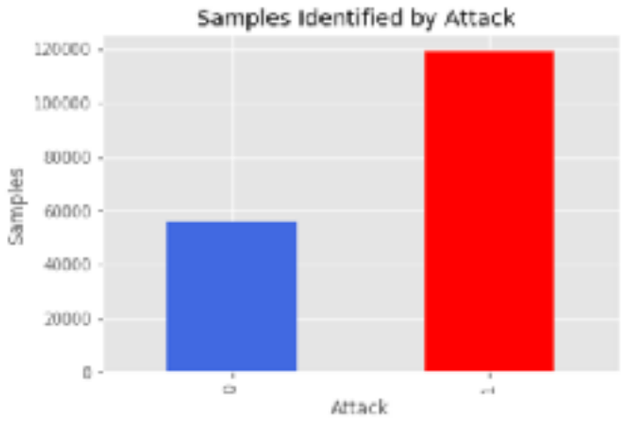
Data Acquisition and Wrangling

- Using Python and pandas we are able to wrangle and to some initial exploration on the data
    - .info()
    - .describe()
    - .shape()
- Data Splitting
    - Splitting the data is used to train and validate algorithms.
    - The UNSW-NB15 dataset is divided into a 60/40 train/test ratio
- Data Size
    - 49 Features
    - 175,341 connections in the training dataset
    - 82332 connections in the testing dataset

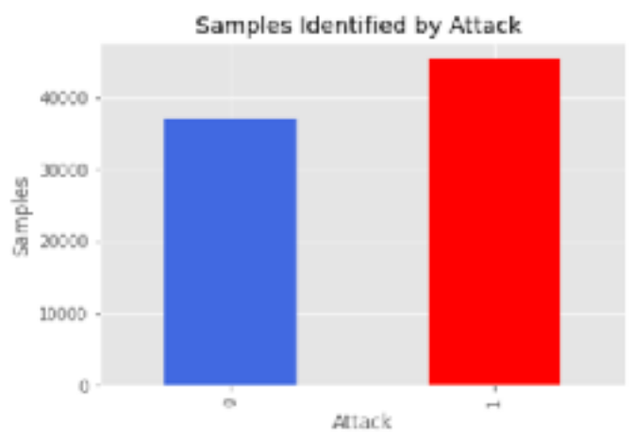
Exploratory Data Analysis

- Visualizations were done using Maplotlib

Normal vs anomaly (Training)



Normal vs anomaly (Testing)



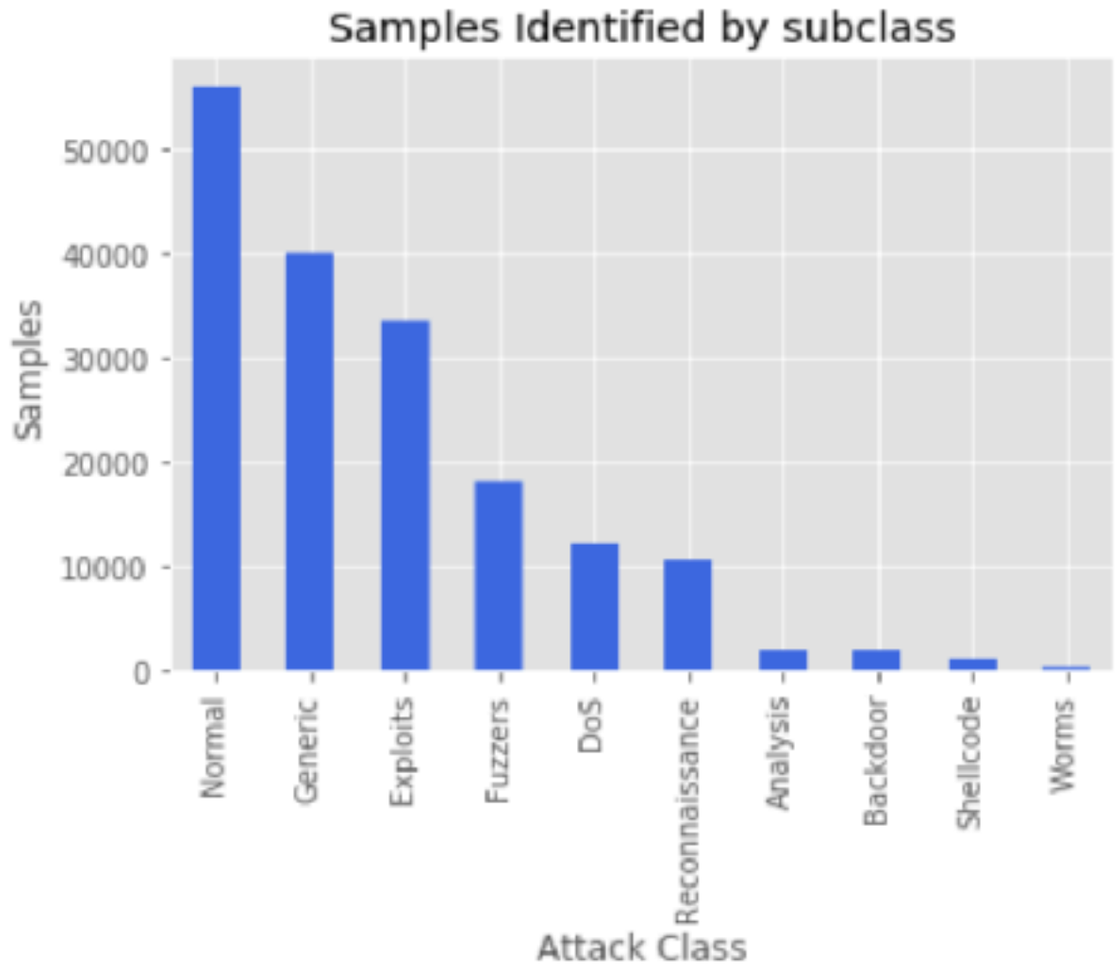
- Training set normal connections equal to 32% of the data and malicious connections equals to 68% of the data.

- Testing set normal connections equal to 45% of the data and malicious connections equal to 51% of the data.

Exploratory Data Analysis

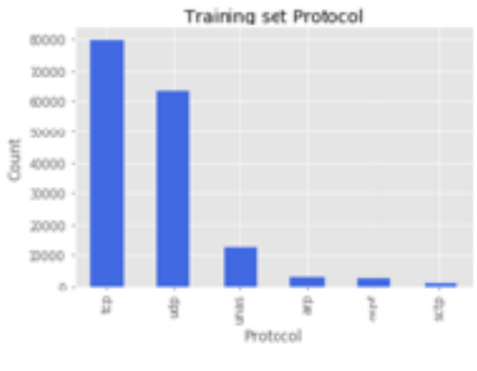
Sub-classes of attacks

- Fuzzers
- Analysis
- Backdoor
- DoS
- Explotis
- Generic
- Reconnaissance
- Shellcode
- Worms

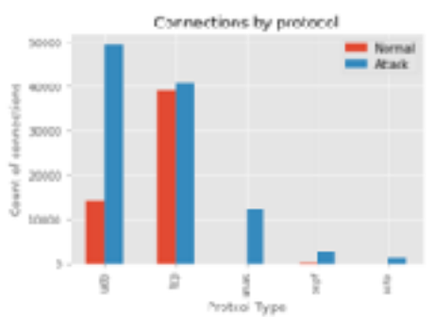


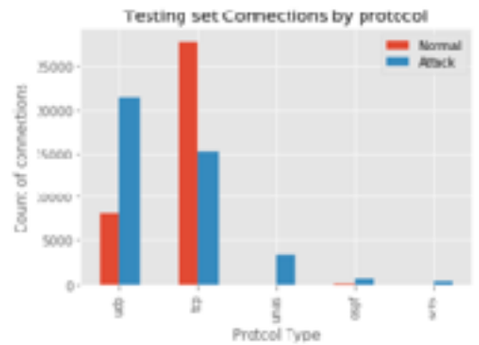
Exploratory Data Analysis

- Protocol and attacks
  - Network protocols are formal standards and policies comprised of rules, procedures and formats that define communication between two or more devices over a network.



- Most networks use a TCP or UDP protocol.



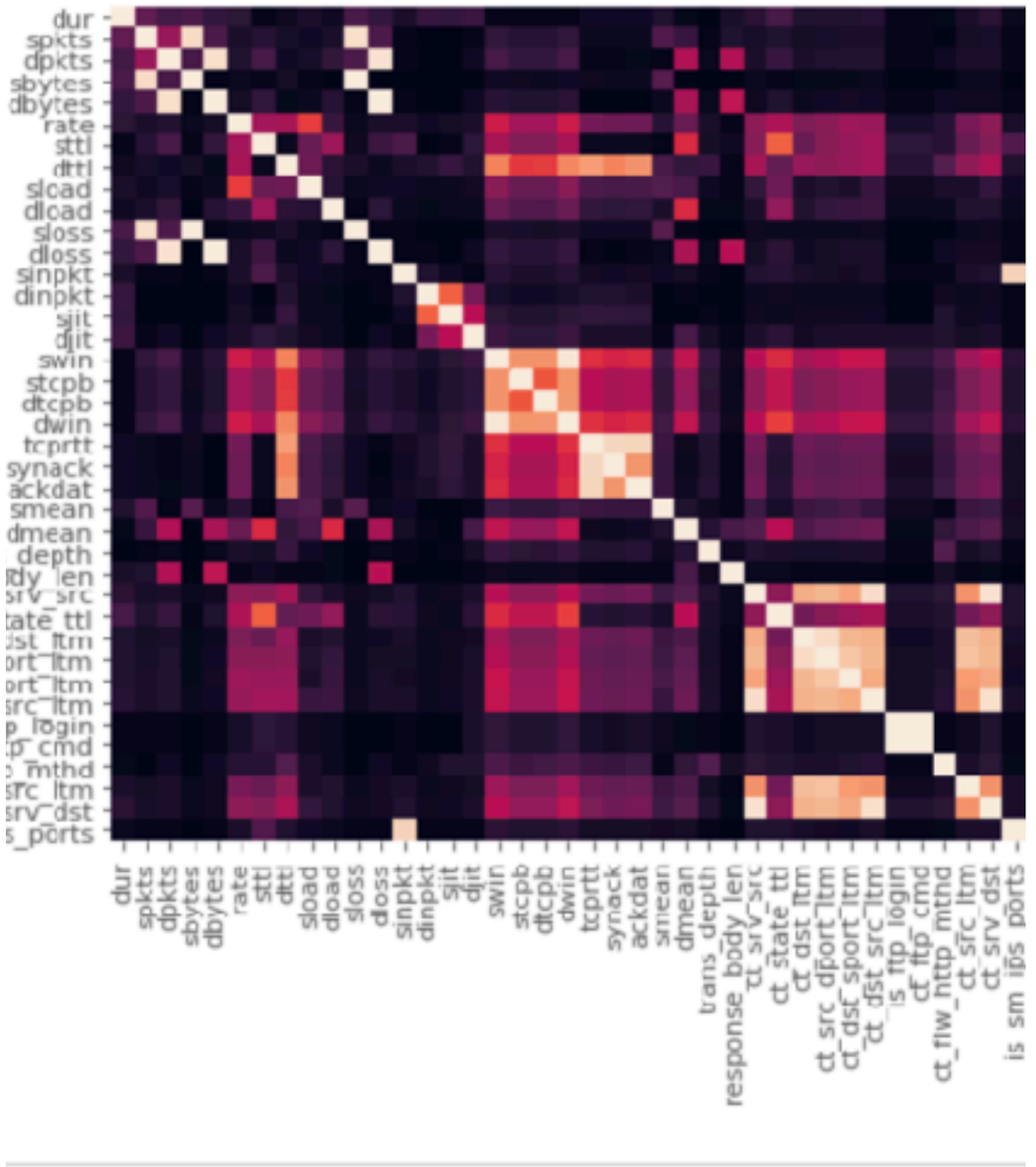


- The charts show the top 5 attacks by protocol
  - Percentages of attacks on UDP protocols is around 78%
  - Percentage of attacks on TCP protocols is around 51%

Exploratory Data Analysis

Feature Correlation

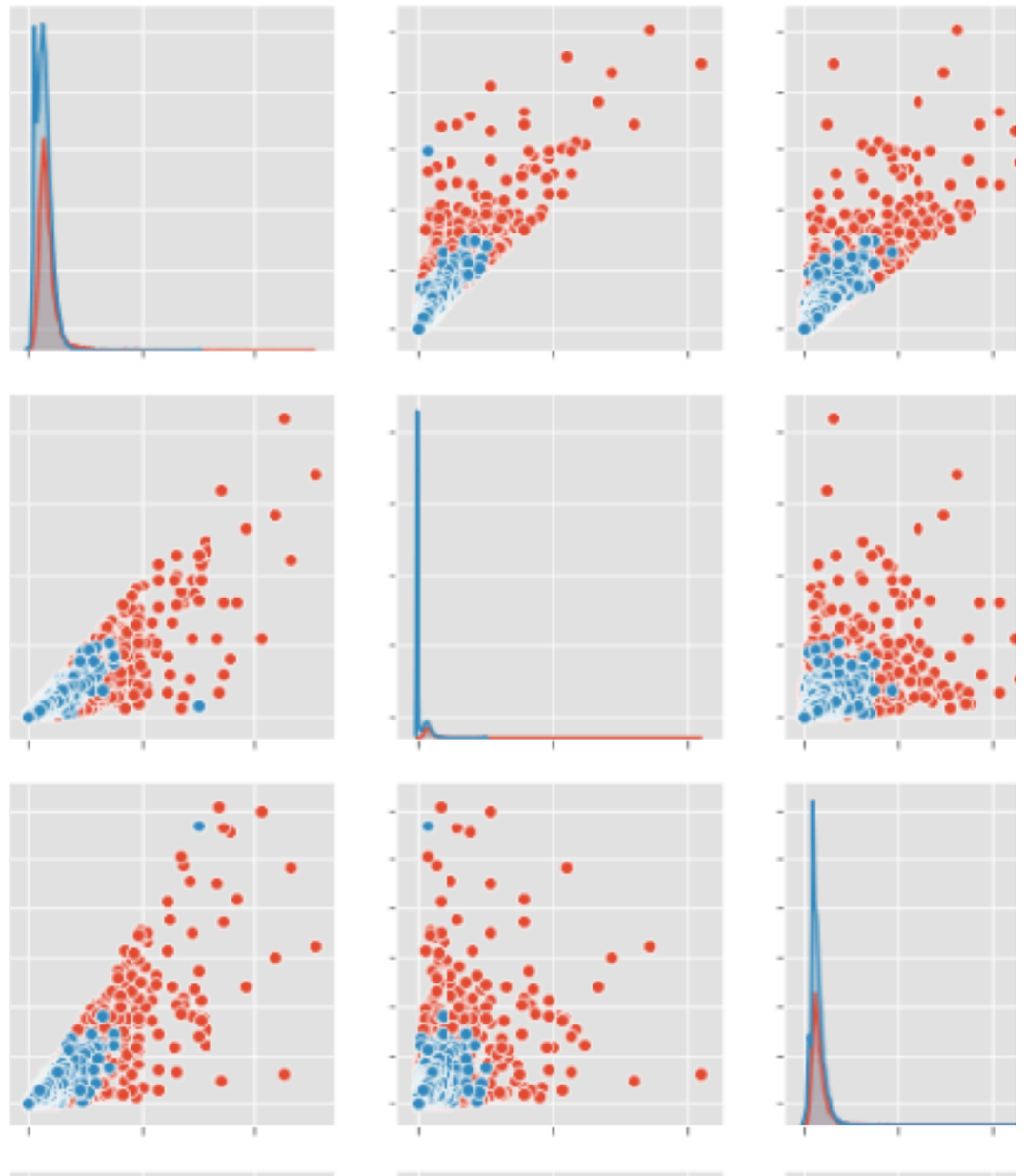
- The dataset consists of various network traffic features. Keeping highly correlated features ideal for our model. The most highly correlated features are:
  - Sloss and skytes
  - Dloss and bytes
  - is_ftp_login and ct_ftp_cmd



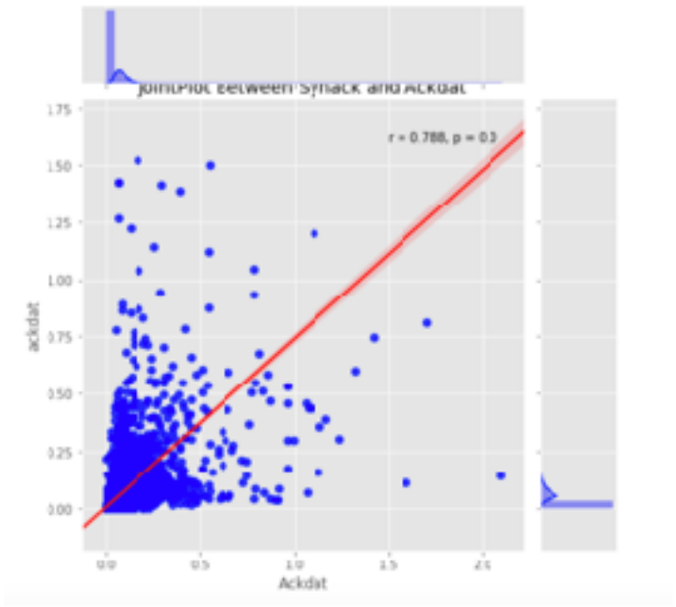
Exploratory Data Analysis

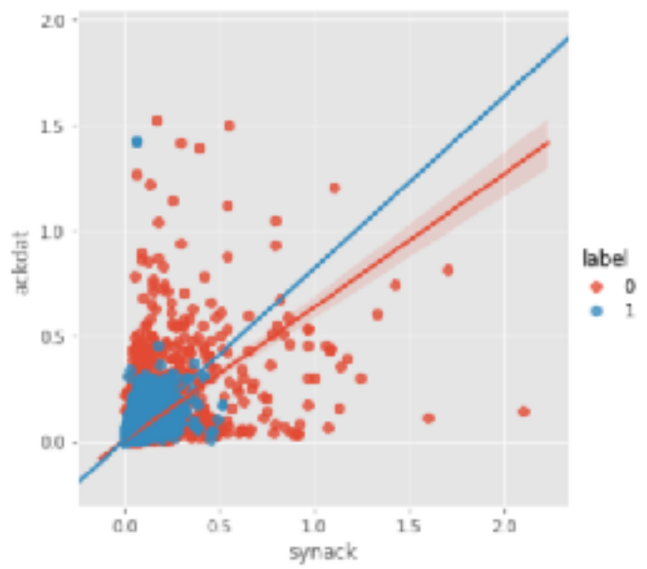
Pairplots

- Give a better visual understanding of the relationships between pairs of features.
- Two features in particular have a positive relationship, the features are sync and ackdat.



Exploratory Data Analysis





Pearson Correlation and p-value

- Synack Is the TCP connection setup time between the SYN and SYN_ACK packets and Ackdat is the TCP connection set up time between SYN_ACK and the ACK packets.
- A packet is a unit of data that is routed between an origin and a destination on the internet or any other packet-switched network.
- The correlation coefficients between the two connection times is positive with a score of 0.788. The p-value is close to 0.0.
- We can also see a clear indication between attacks and normal networks.

Performance Metrics

- **Classification Accuracy**: the number of correct predictions made as a ratio of all predictions made.

- **Classification Report:** Report displays the precision, recall, f1-score and support for each class
  - Precision: The ability of a classifier not to label an instance positive that is actually negative.
  - Recall: The ability of a classifier to find all positive instances.
  - F1 Score: The weighted harmonic mean of precision and recall.

Performance Metrics

- False Positives (FP) : False positives are the cases when the actual class of the data point was 0 (False) and the predicted is 1 (True).
- False Negatives(FN): False negatives are the cases when the actual class of the data point was 1 (True) and the predicted is 0 (False).

What to minimize?

- Minimize False Negatives

Preprocessing

- Convert categorical variables to dummy variables
- Standardize and normalize

Logistic Regression

- Baseline model - Logistic Regression
    - Goal is to predict attack or normal
    - Logistic Regression is an algorithm that is used for classification of binary data.
- **<u>Results for attack data</u>**
    - Accuracy: **80%**
    - Precision: **75%**
    - Recall: **97%**

Handling Imbalance Data

Resampling Techniques: Resampling is down to handle imbalance data.

- Weigted
- Upsampling
- Downsampling

Results:

- Weighted: Accuracy: 83%, Precision: 80%, Recall: 93%
- Upsampling: Accuracy: 82%, Precision: 80%, Recall: 93%
- Downsampling: Accuracy: 83%, Precision: 80%, Recall 93%

Random Forest

- Random Forest model takes advantage of the bagging process where it takes different training samples with replacement in order to get predictions for each observation and then averages all the predictions to obtain the estimation.

- **<u>Results</u>**
  - Accuracy: **86.6%**
  - Precision: **82%**
  - Recall: **96%**

Random Forest- with tuning

- **<u>Results</u>**
  - Accuracy: **81%**
  - Precision: **74%**
  - Recall: **100%**

XGBoost

- XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable.

- **Results**

  - Accuracy: **83%**

  - Precision: **77%**

  - Recall: **98%**

Deep Neural Network

- Deep Learning is an increasingly popular subset of machine learning. Deep learning models are build using neural networks. DNN takes in inputs, which are then processed in hidden layers using weights that are adjusted during the training. Then the model spits out a prediction.

- Pipeline for creating a DNN

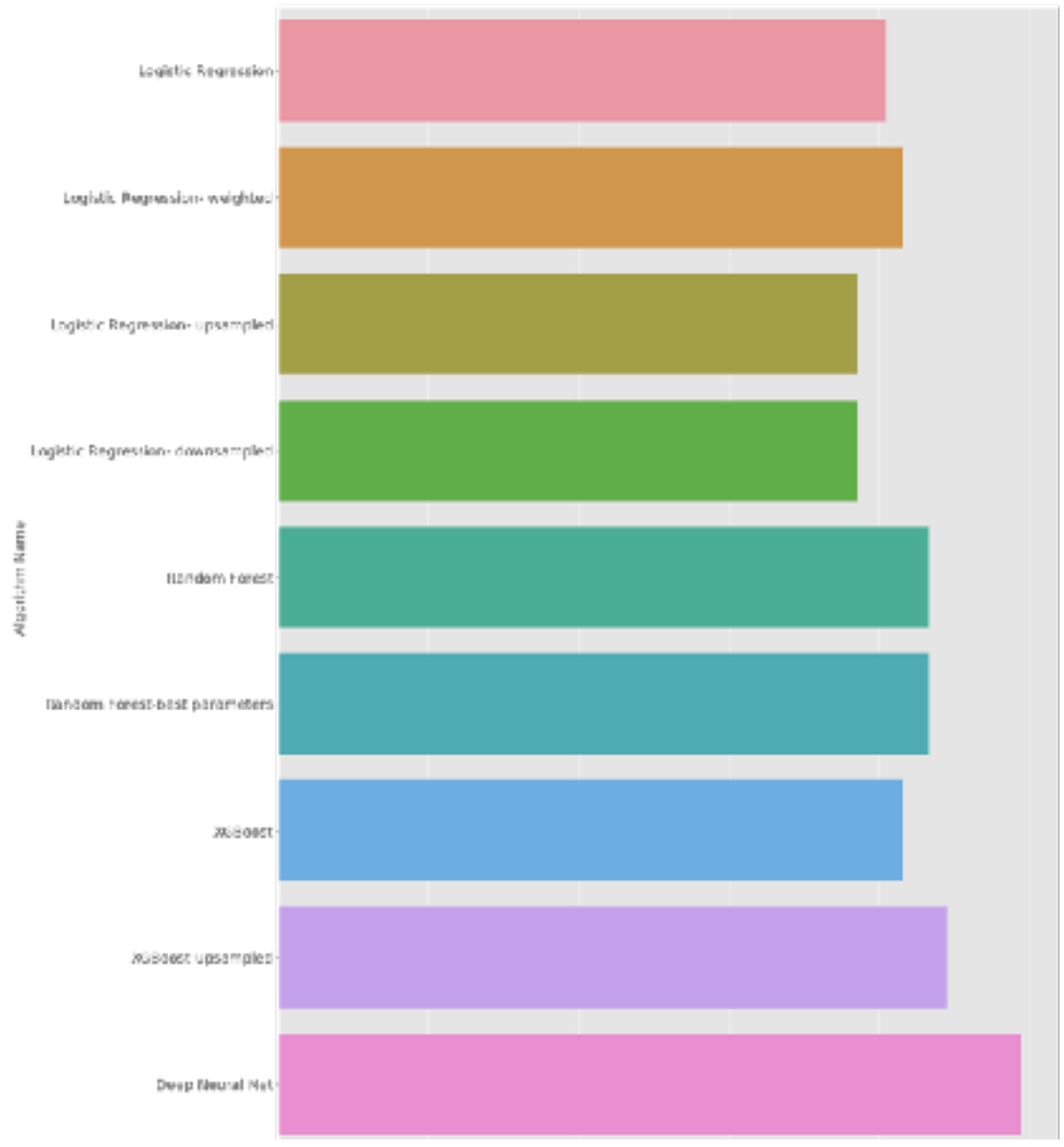
    - Building the model
    - Compiling the model
    - Training the model

- **Results**

    - Accuracy: **98.8%**
    - Precision: **81%**
    - Recall: **96%**

Results

- Deep Neural Nets computed the highest accuracy, followed by XGBoost on upsampled data and then random forest.



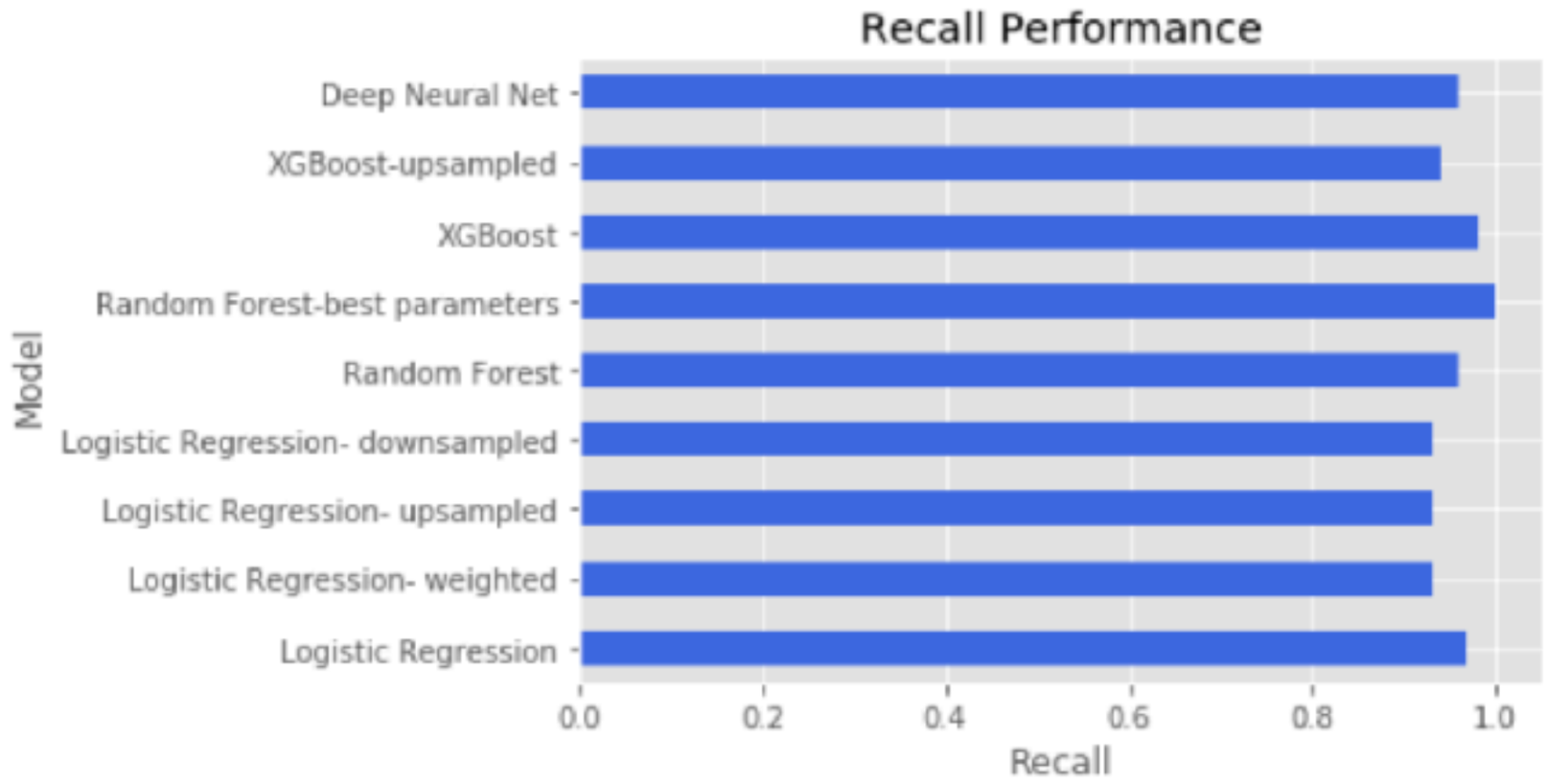
Results

Model	Accuracy	Precision	Recall	F1-Score	Roc/auc score
Logistic Regression	0.808118	0.75	0.97	0.85	0.953261
Logistic Regression- weighted	0.830649	0.80	0.93	0.86	0.953385
Logistic Regression- upsampled	0.771766	0.80	0.93	0.86	0.947094
Logistic Regression-downsampled	0.771984	0.80	0.93	0.86	0.948824
Random Forest	0.865969	0.82	0.96	0.89	0.958530
Random Forest-best parameters	0.865969	0.75	1.00	0.85	0.966968
XGBoost	0.830965	0.77	0.98	0.86	0.972704
XGBoost-upsampled	0.891112	0.88	0.94	0.90	0.972826
Deep Neural Net	0.988985	0.81	0.96	0.88	0.953169

- Intrusion detection systems are implemented to detect network attacks. Therefore we want to focus on attack labels.

- Our goal is the minimize false negatives. For our problem we want a recall near 1.0. This indicates the model was able to find all attacks that are actual attacks.

Results



- Random Forest with tuning gives us a recall of 1.00. This indicates the model correctly identified 100% of attacks that were actually attacked. However, the precision is at 75% indicating that 75% of positive cases were correctly identified.

Conclusion

Model’s Applicability

- Model is great at minimizing false negatives
- Model captures 100% of attack networks correctly
- Model gives important features
- The Random Forest Classifier model with hyper-parameter tuning is the best classifier. Holds a recall score of 100%.