Machine Learning – Designing a Classifier

Jordan Gribben

CMP417: Engineering Resilient Systems Part 1

BSc Ethical Hacking Year 4

2021/22

Contents

1 INTRODUCTION

1.1 BACKGROUND

ScottishGlen are looking to monitor their company network, this is to try and discover any suspicious traffic that may be present. The traffic should also be sorted into 10 categories, these categories represent a different attack type, these categories are: Normal, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. An efficient method of discovering the suspicious traffic as well as categorising it must be implemented within ScotishGlen.

A machine learning algorithm will be implemented within ScottishGlen to achieve this goal. The most appropriate type of algorithm for this project is a classification algorithm. A classification algorithm allows for items to be grouped together based on their common attributes (Mesevage, 2020). Once this process is complete, the algorithm provides a decision for the inputted data. Utilizing a classifier algorithm within ScottishGlen will help sort the suspicious network traffic.

This paper will investigate classifier algorithms, such as the random forest and decision tree algorithms. Once the algorithms have been discussed the more affective one will be chosen for implementation within ScottishGlen. Methods of evaluating the performance and accuracy of the chosen classifier will then be discussed, with an appropriate method being chosen to evaluate the classifier.

1.2 DATASET

A test dataset containing network traffic ScottishGlen collected has been provided and is stored in a Microsoft Excel file format. The file provides various details about the traffic such as, protocol, ID, state, and an attack category. The attack category lists the type of attack that is being used, listing one of the 10 types listed out in the previous section.

1.3 AIM

This project aims to design an appropriate classifier for the dataset provided. This classifier should be able to differentiate between malicious network and normal network traffic, the classifier should also be able to identify the various attack types.

2 ALGORITHMS

This section will investigate the various algorithms that could be implemented within ScottishGlen. Once the algorithms have been investigated, a summary will be written that will go over the pros and cons before a decision is made on what one will be used.

2.1 DECISION TREE

Decision trees are one of the most successful types of classification algorithm, with them even forming the basis for other classification algorithms (Su & Zhang, 2006). A visualisation of a decision tree looks like an actual tree, with the dataset being implemented into the initial 'root' decision, with further decisions branching off from the root, these branches form further branches until the process is complete. This allows for large decisions to be broken down into smaller ones, these smaller decisions also allow the process to be mapped more efficiently as well as being sorted accurately. Figure A shows a visualisation of a decision tree, showing off what they look like and how they work.

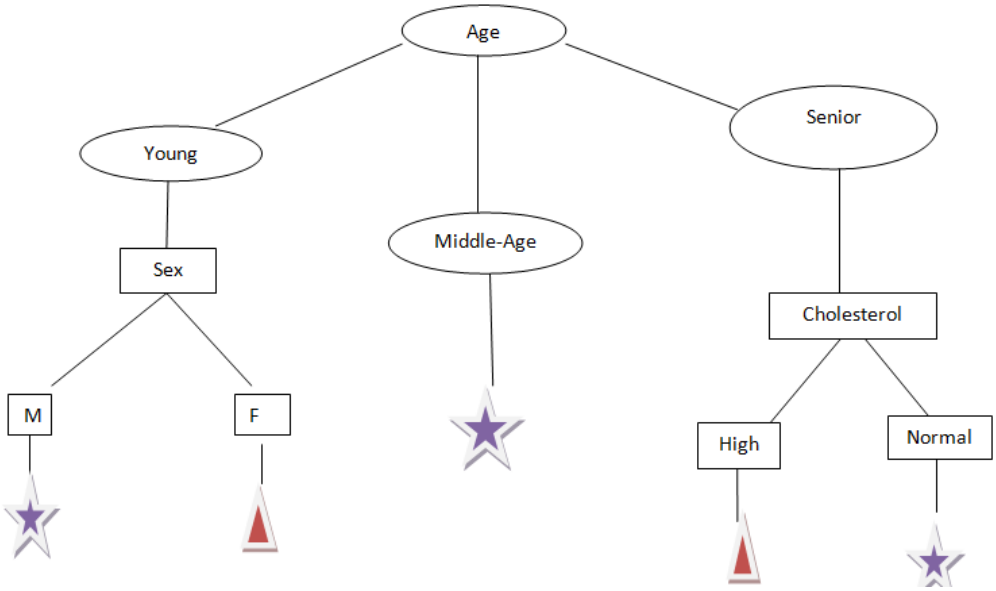

Figure A - Decision Tree Algorithm for Classification (Sharma, 2021)

Decisions trees are an extremely useful algorithm and could be used to investigate the suspicious network traffic. However, with the size of the data that is being investigated alongside the various categories it has to be sorted into, the decision tree would quickly become extremely large and complex. As a result, this could potentially make the results produced by this algorithm inaccurate. With the many complex branching paths the algorithm could take, efficiency could be decreased due to complexity, leading to confusion within the algorithm and inaccurate results.

2.2 RANDOM FOREST

The random forest algorithm can be seen as an expansion of the decision tree algorithm (Breiman, 2001). Random forest utilises multiple decision trees, creating a type of 'forest'. Using multiple decision trees allows the random forest algorithm to stay simple, while being fast learning compared to other algorithms. Each tree within the forest can produce a classifier prediction, due to this the algorithm is commonly used due to its different recognition systems.

With the forest containing multiple trees, multiple results are produced, this process is repeated, and all the results are combined. This allows for greater accuracy within the results as any false positives or other inaccurate results can be discredited due to the other results. This algorithm works best when dealing with larger datasets as it can use a "divide and conquer" approach to tackle the dataset (Dasgupta, 2018). To better show how the multiple trees within the forest work, figure B contains an example of how this algorithm would use multiple trees for an input.

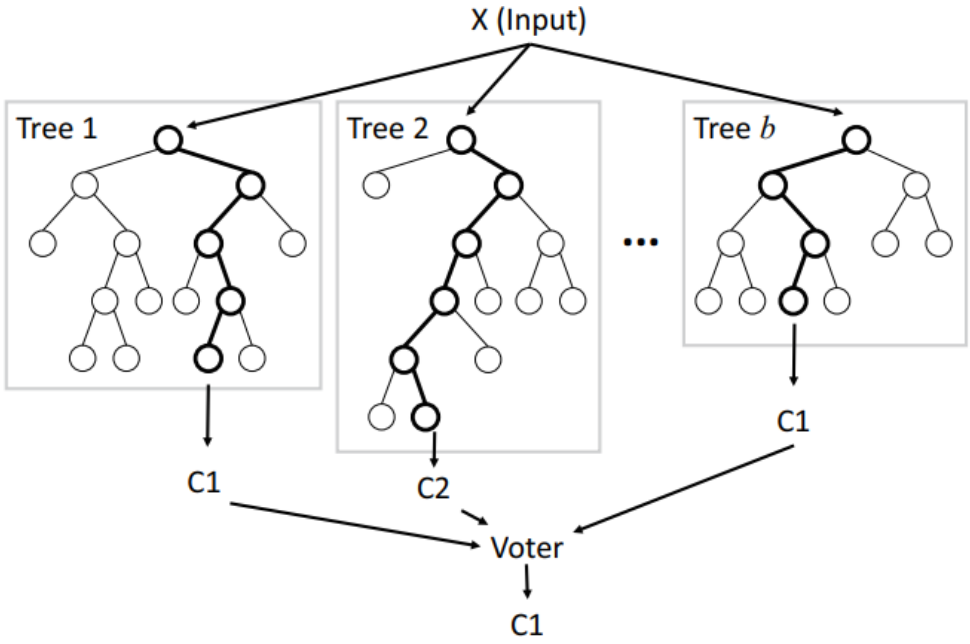

Figure B - Random Forest Diagram (Nakahara, et al., 2017)

In summary, due to random forest producing multiple trees, as well as having a diverse set of trees the results will have greater accuracy. With the multiple trees producing various results, these can be combined to discover what the most accurate results would be.

2.3 LOGISTIC REGRESSION

Logistic regression algorithms work well when dealing with binary identification such as emails. This algorithm can identify the legitimacy of an email with great accuracy (Wijaya & Bisri, 2016), by classifying a legitimate email as a 0 and an illegitimate email such as spam or phishing email as a 1. When displaying the results of a logistic regression algorithm, a graph is used. The graph draws an S-shaped curve between 0 and 1, the closer to 1 the curve gets the algorithm is predicting a likely probability of an event such as an illegitimate email occurring. Figure C shows an example of a logistic regression graph.

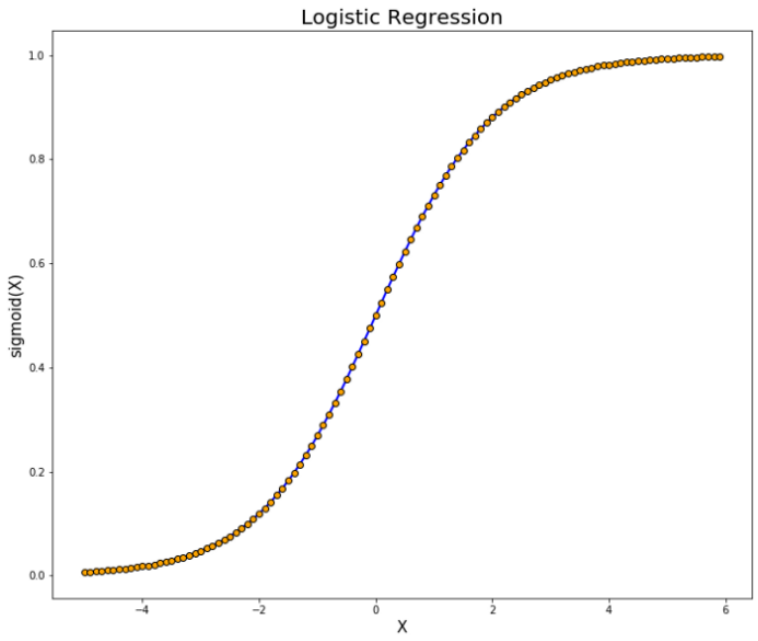

Figure C - Logistic Regression Graph (z_ai, 2020)

While logistic regression is a useful algorithm it would not be suitable for this project. This is due to the network traffic containing various attack categories, logistic regression works best when it comes to binary classification. While the algorithm can be amended to work with multiple variables, there are other algorithms which would be better suited at dealing with multiple variables such as random forest.

2.4 SUMMARY

Looking at the algorithms listed within this report, the random forest algorithm has been chosen to review the suspicious traffic within ScottishGlen. This algorithm was chosen due to the accuracy of the results it can produce, the algorithms simplicity, and that it best works with large datasets, like the one containing ScottishGlens network traffic.

Furthermore, while a logistic regression algorithm could be amended to work for this project, this would take up addition time for ScottishGlen. Even if the algorithm was to be amended the random forest results would most likely be more accurate.

While ScottishGlen will move ahead with using the random forest algorithm, it should be noted that it has its flaws. The final product produced by this algorithm tends to be large and complex, this does not fully align with ScottishGlens requirements.

3 CLASSIFIER

As laid out in section 2, ScottishGlen will implement a random forest algorithm to investigate and categorise the suspicious network traffic. When designing a classifier there are various factors that must be considered, these being; Data Ingestion/Pre-processing, Modelling, and the evaluation of Results.

Previous research using the UNSW-NB15 dataset has categorised the network data into 9 categories. Part of the UNSW-NB15 data set was divided into a training set consisting of 175,341 records, alongside a testing set containing 82,332 records (Moustafa & Slay, 2015). Each of the records were divided into the following 9 categories.

- Fuzzers – A fuzzer attack is when a system such as a network, operating system, or application is overloaded with a massive data input by using a security loophole found within the system.

- Analysis – Analysis is a variety of intrusions that target web application, these intrusions penetrate the web application via ports, emails, and web scripts.

- Backdoor – A backdoor is a stealth based technique, this attack allows the malicious user to bypass a systems normal authentication, providing the malicious user with direct access to the system.

- DoS – Denial of Service (DoS) is an attack that stops a system from performing its desired task. For example if a web application was to face this attack, the application would stop functioning or become inaccessible altogether.

- Exploit – An exploit is used by taking advantage of vulnerabilities within a system, utilising an exploit can cause the system to perform with unintended behaviour.

- Generic – Generic is an exploit used against block ciphers. By using hash functions, a generic attack will cause a collision as it has not been correctly configured to the block cipher.

- Reconnaissance – This is an attack designed to gather information about a network while evading security controls. This type of attack is usually carried out before another attack, as a malicious user could use the information discovered here to perform another attack.

- Shellcode – Shellcode is a type of malware, in which a malicious user will utilise a piece of code starting from a shell to gain access to a system.

- Worm – A worm is an attack that replicates itself. It utilises its self replication abilities to spread itself throughout a network and onto devices within that network.

Utilising this classification for network traffic there are only 9 categories listed, compared to the 10 listed in the introduction. Due to this any network traffic that is not categorised into one of these categories is most likely within the 10th category listed, being “normal”.

Using the test dataset provided, the random forest algorithm will be able to create trees for each of the listed attack categories. This would allow network traffic to be sorted by attack category, allowing ScottishGlen staff to see exactly which data is suspicious and what attacks they are facing.

4 EVALUATION

With the classifier chosen, effective methods to evaluate its performance must be in place. The following sections will discuss performance metrics, that could be effectively used to evaluate the performance of the random forest algorithm.

4.1 CONFUSION MATRIX

A confusion matrix gets its name from the way it displays its results, this is done using a matrix table. This table is used to describe the algorithms performance through the comparison of a predicted and true outcome. The confusion matrix can provide the following outputs:

- True Positive – Is positive and was predicted positive
- True Negative – Is negative and was predicted negative
- False Positive – Is negative but was predicted positive
- False Negative – Is positive but was predicted negative

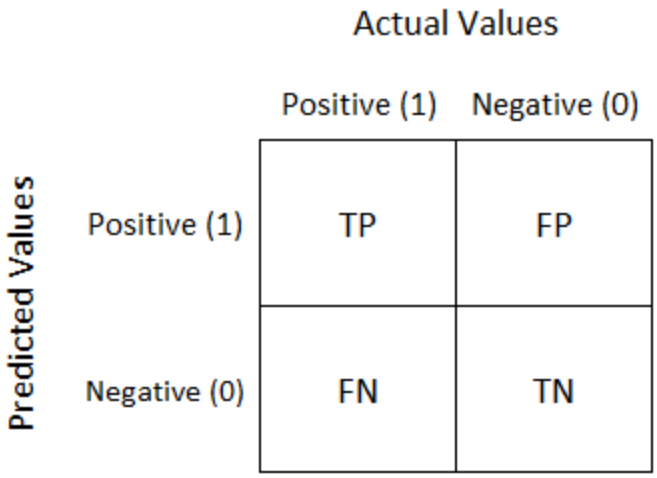

Figure D - Confusion Matrix Table (Narkhede, 2018)

This method of evaluating performance is extremely useful, the accuracy of this method can be calculated by reviewing both the true positive and false negative results (Narkhede, 2018).

4.2 Receiver Operating Characteristics Curve

Unlike the confusion matrix which utilises a table to display its results, a Receiver Operating Characteristics (ROC) curve uses a graph. This graph is used to display the true positive rate against the false positive rate. An example of this graph can be seen in figure E.

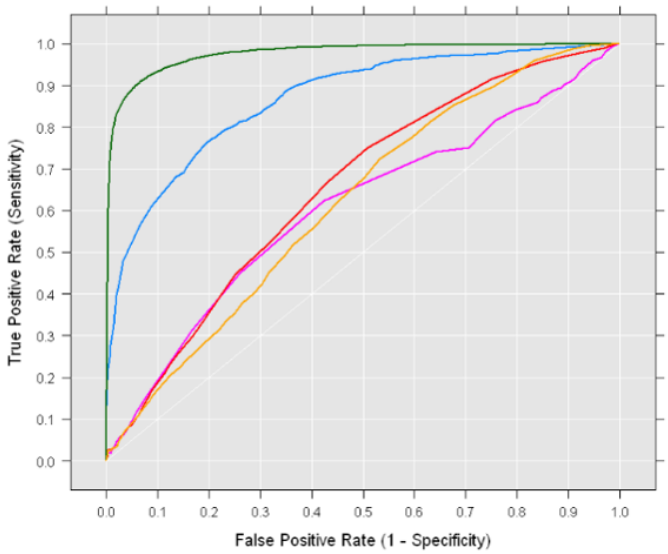

Figure E - ROC Curve Graph (Chan, 2022)

Within a ROC curve graph, the area under the curve is how the accuracy of the classifier is determined. If the graph is close to 0 it can be determined that the classifier is inaccurate, with the close to 1 it gets the more accurate the classifier gets (Narkhede, 2018).

4.3 Holdout Method

The holdout method is the most simplistic technique out of the ones listed within this report. This method of evaluating the classifier splits the dataset into two different sets, a training set and a testing set, when dividing this data the training set is larger than the testing set (Allibhai, 2018). The ratio of the two datasets tends to be 80:20 or 70:30 (Great Learning Team, 2020). The testing data provides the accuracy of the classifier.

While this method at first appears to be ideal for this project, due to having both a training and testing dataset, ultimately this method would not be an efficient method to evaluate the random forest classifier. This is due to how the method works, due to holdout method splitting 1 dataset into a training and testing set rather than inputting 2 datasets this method would not be affective. If this project was provided with just one dataset this evaluation method would have been accurate and efficient for this project.

4.4 SUMMARY

A method of evaluation will be used alongside the random forest algorithm to evaluate its efficiency. Out of the methods listed, the holdout method will be the least affective as without modification it would not be suitable as the dataset is in 2 parts. Out of both the ROC curve and the confusion matrix, ScottishGlen will implement the confusion matrix as their way to evaluate their random forest algorithm.

The confusion matrix has been chosen due to its accuracy alongside its visualisation with a matrix table. The confusion matrix table is simple to understand due to it having 4 outcomes, this allows it to be easier to understand at first glance compared to a ROC curve. This understanding will help communicate the network traffic affectively to the employees within ScottishGlen.

5 REFERENCES

Allibhai, E., 2018. *Hold-out vs. Cross-validation in Machine Learning.* [Online]
Available at: https://medium.com/@eijaz/holdout-vs-cross-validation-in-machine-learning-7637112d3f8f
[Accessed 12 April 2022].

Breiman, L., 2001. *Random Forests,* Berkeley: Kluwer Academic Publishers.

Chan, C., 2022. *What is a ROC Curve and How to Interpret It.* [Online]
Available at: https://www.displayr.com/what-is-a-roc-curve-how-to-interpret-it/
[Accessed 13 April 2022].

Dasgupta, S., 2018. *De(Coding) Random Forests.* [Online]
Available at: https://towardsdatascience.com/de-coding-random-forests-82d4dcbb91a1
[Accessed 12 April 2022].

Great Learning Team, 2020. *What is Cross Validation in Machine learning? Types of Cross Validation.* [Online]
Available at: https://www.mygreatlearning.com/blog/cross-validation/
[Accessed 13 April 2022].

Mesevage, T. G., 2020. *Machine Learning Classifiers - The Algorithms & How They Work.* [Online]
Available at: https://monkeylearn.com/blog/what-is-a-classifier/
[Accessed 2 April 2022].

Moustafa, N. & Slay, J., 2015. *The significant features of the UNSW-NB15 and the KDD99 data sets for Network Intrusion,* Sydney: University of New South Wales.

Nakahara, H., Jinguji, A., Sato, S. & Sasao, T., 2017. *A Random Forest Using a Multi-valued Decision Diagram on an FPGA,* Novi Sad: IEEE.

Narkhede, S., 2018. *Understanding AUC - ROC Curve.* [Online]
Available at: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5
[Accessed 12 April 2022].

Narkhede, S., 2018. *Understanding Confusion Matrix.* [Online]
Available at: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62
[Accessed 12 April 2022].

Narkhede, S., 2018. *Understanding Confusion Matrix.* [Online]
Available at: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62
[Accessed 12 April 2022].

Sharma, A., 2021. *Machine Learning 101: Decision Tree Algorithm for Classification.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2021/02/machine-learning-101-decision-tree-algorithm-for-

classification/#:~:text=The%20goal%20of%20this%20algorithm,internal%20node%20of%20the%20tree.
[Accessed 12 April 2022].

Su, J. & Zhang, H., 2006. *A Fast Decision Tree Learning Algorithm,* New Brunswick: AAAI.

Wijaya, A. & Bisri, A., 2016. *https://ieeexplore.ieee.org/abstract/document/7863267,* Yogyakarta: IEEE.

z_ai, 2020. *Logistic Regression Explained.* [Online]
Available at: https://towardsdatascience.com/logistic-regression-explained-9ee73cede081
[Accessed 12 April 2022].