

**CSIS 3290-001**

**Fundamentals of Machine Learning**

**Instructor: Dr. Ivan Wong**

**Project 03**

By

**Bryan Rodriguez**

**Hazel Ibasco**

**Nathaniel Pearson**

**Shedrach Ezenwali**

A Final Project Report Submitted to the Department of Computing Studies & Information Systems in Partial Fulfillment of the Requirements for the Degree

**December – 2023**

Table of Contents

[I. Introduction and discovery 3](#_Toc152695066)

[i. Introduction 3](#_Toc152695067)

[ii. Framing the problem 3](#_Toc152695068)

[iii. Initial hypotheses 3](#_Toc152695069)

[II. Data Preparation 4](#_Toc152695070)

[i. Data inventory 4](#_Toc152695071)

[ii. Data processing 4](#_Toc152695072)

[III. Model Planning and Implementation 6](#_Toc152695073)

[i. Proposed model(s) and justification 6](#_Toc152695074)

[ii. Determination of Model Workflow 6](#_Toc152695075)

[iii. Efficiency of Workflows 6](#_Toc152695076)

[iv. Testing Hypotheses and Modeling Objectives: 6](#_Toc152695077)

[IV. Results Interpretation and Implications 7](#_Toc152695078)

[i. Results 7](#_Toc152695079)

[ii. Assess and Key findings 7](#_Toc152695080)

[iii. Major insights 7](#_Toc152695081)

[V. Out-of-sample Predictions 7](#_Toc152695082)

[iv. Using new data 7](#_Toc152695083)

VI. Conclusion Remarks……………………………………………………………………………………………………………………………8

# Introduction and discovery

## Introduction

The business domain revolves around traffic data analysis with a focus on encrypted malicious traffic detection. The dataset is a compilation of balanced sizes of encrypted malicious and legitimate traffic from five public datasets. This dataset is intended for machine learning model training in the domain of cybersecurity. The data set contains 10,476 rows and 113 columns and data from both conventional devices as well as “IoT” devices from a plethora of different environments including homes, offices, etc. It is not a highly specialized data set concerning a very particular circumstance and type of traffic, but rather a large compilation of data from many different sources collected and then randomly sampled.

The target organization, in this case, is not explicitly mentioned, but the focus is on creating a dataset that aids in developing models for the detection of encrypted malicious traffic. This is crucial in the context of increasing cyber threats, where the distinction between legitimate and malicious traffic becomes challenging due to encryption.

## Framing the problem

Framing the problem involves addressing questions related to the effective identification of encrypted malicious traffic within network data. The importance of this analysis lies in enhancing cybersecurity measures by developing accurate and efficient models that can distinguish between normal and potentially harmful network activities.

An issue with detecting malicious software is that often datasets are skewed heavily in favor of non-malicious traffic, as most traffic that occurs on is completely safe. The amount of available non-sensitive, well labeled, and well-maintained data are all limiting factors. It is also hard to tell whether the model you create for filtering out malicious traffic will be able to keep up in the future. The models and variables you pick today may or may not be as successful later on. Polymorphic and metamorphic malware create modified versions of themselves in order to evade detection. When those creating malware and the malware itself are continuously adapting, it is hard to know how effective any model you create will be once implemented.

Any model that is created must be able to detect threats in real time, meaning that it must be resource efficient and not very computationally expensive. Our goal in this project is to create a model that can operate quickly and efficiently given the nature of the problem.

## Initial hypotheses

* The proportion of encrypted malicious traffic within the dataset is expected to be relatively small compared to legitimate traffic due to the nature of cybersecurity threats.
* The balance between malicious and legitimate traffic, along with the inclusion of both conventional and IoT devices, aims to create a dataset that represents real-world scenarios, leading to robust model training.
* The number of variables chosen in variable selection will be much smaller than the total number of columns. A few key columns will play a disproportionate role in determining whether traffic is malicious.
* Malicious traffic will be hard to distinguish between non-malicious traffic due to things like polymorphic malware.

# Data Preparation

## Data inventory

The dataset is composed of five public traffic datasets, including the Malwares Capture Facility Project dataset and the CICIDS-2017 dataset. The data has been pre-processed to ensure balance, both in terms of the malicious and legitimate traffic and the contributions from individual datasets. Details of each selected public dataset and the final composed dataset are provided in the "Dataset Statistic Analysis Document."

The dataset was obtained from <https://data.mendeley.com/datasets/ztyk4h3v6s/2> searching with <https://datasetsearch.research.google.com/>

## Data processing

1. Most machine learning algorithms can only work with numerical data. But, in many cases, what we have is categorical data. Because algorithms cannot perform mathematical operations on them, it is necessary to transform them before they can be used to train any machine learning model. Something that can be achieved by creating dummy variables or binary variables. The most common method for creating dummy variables is to convert each of the categories of the original variable into a new binary column. If the row corresponds to the category, it will have a value of 1, in any other case it will have 0.
2. As we have a type of high-dimensional data sets and they are all used for the creation of Machine Learning models, it can cause:

* The additional features act as noise, which the Machine Learning model can perform extremely poorly.
* The model takes longer to train.
* Allocation of unnecessary resources for these features.

For all this, feature selection must be implemented in Machine Learning projects.

Feature Selection is the process of selecting the most important and/or relevant features of a data set, intending to improve the prediction performance of predictors, providing faster and more cost-effective predictors, and providing a better understanding of the process.

1. Check to avoid null data in the data set.
2. As the combination of RandomForest\_Feature\_selection + RobustScaler (scaling method) + Random Forest (classifier) is the best feature selection method we will work with it. Avoiding executing the test again in Jupiter Notebook because of machine resources.

A white background with black and purple text

Description automatically generated

1. In the dataset, it appears that the target variable is the "label" column. This column contains binary values, with 1 indicating a certain condition or class and 0 indicating another. The specific meaning of these classes would depend on the context of your dataset and the problem you are trying to solve.

For instance, in some network security or intrusion detection scenarios, the label could represent whether a network activity is classified as normal (0) or malicious (1). Alternatively, it could represent different types of network activities or any binary classification task relevant to your domain.

# Model Planning and Implementation

## Proposed model(s) and justification

Given the constraint that a deep learning approach is not feasible, a more traditional machine learning model will be employed. A Random Forest classifier is a suitable choice. Random Forest is known for its efficiency, interpretability, and ability to handle diverse datasets without overfitting. Its ensemble nature makes it robust against noise and suitable for capturing complex relationships within the data.

## Determination of Model Workflow

The workflow will involve initial data preprocessing, feature selection, and training a Random Forest classifier. Instead of clustering, which might add complexity and computational overhead, we will focus on a single model. The model will be trained on the balanced and preprocessed dataset.

## Efficiency of Workflows

To meet the time constraint, the workflow will be streamlined using pipelines for data preprocessing, feature selection, and model training. This ensures a sequential and efficient process.

## Testing Hypotheses and Modeling Objectives:

The Random Forest model will be evaluated based on its ability to accurately detect encrypted malicious traffic within the given time constraint. Insights gained from the model will contribute to understanding the features crucial for distinguishing between malicious and legitimate traffic.

Random Forest is known for its efficiency, and the use of pipelines will contribute to meeting the 2-minute runtime requirement. Feature selection techniques will be applied judiciously to reduce dimensionality and speed up the training process.

# Results Interpretation and Implications

## Results

**Precision**: Precision is the number of true positives divided by the sum of true positives and false positives. It measures the accuracy of the positive predictions. In our case, the precision for class 0 is 0.98, and for class 1 is 1.00. High precision indicates that the model has a low rate of false positives.

**Recall (Sensitivity)**: Recall is the number of true positives divided by the sum of true positives and false negatives. It measures the ability of the model to capture all the relevant examples. In our case, the recall for class 0 is 1.00, and for class 1 is also 0.98. High recall indicates that the model can identify most of the positive examples.

**F1-score**: The F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall. In our case, the F1-score for class 0 is 0.99, and for class 1 is also 0.99. The weighted average of these scores is also 0.99. High F1-score indicates a good balance between precision and recall.

**Support**: Support is the number of actual occurrences of the class in the specified dataset. In your case, the support for class 0 is 1065, and for class 1 is 1031.

**Accuracy**: Accuracy is the overall correctness of the model and is calculated as the sum of true positives and true negatives divided by the total number of samples. In your case, the overall accuracy is 0.98.

## Assess and key findings

Based on the results explained in the previous section, the accuracy of our prediction model is acceptable; however, we aim to improve it further by merging the selected model with other performers from the search. Our objective is to create an Ensemble model (soft vote) which will be evaluated to determine if it can reduce the count of 26 False Negative instances, while maintaining a low likelihood of a malicious attack.

Creating an Ensemble Model. Since our objective is to create an Ensemble model, we proceeded to do so. The result had an accuracy score of 0.97, which is slightly lower than our initial model. Although it is lower, we propose this model instead of our initial model since we have produced 8 as the False Negative and 1023 as the True Positive. This is better considering the business logic and the need to protect the server from malicious traffic, and we will put first the ability to detect as much of such traffic as possible.

In the context of cyber security, a false negative is more consequential than a false positive. It may be that we would expect a properly working malware detection system to flag many more false positives than false negatives.

If you use malware detection software yourself, you probably encounter many false positives and hopefully very few false negatives. If we were to put this model into practice, we would definitely want a result where the number of false positives was much higher, but the number of false negatives was close to 0, or at the very least, lower than our false positives. For a machine learning model, these results are not terrible but in a real-world application they do not appear to be up to par. It may partially jeopardize the user experience to alter the model to skew towards false positives, but overall, it would make more sense.

# Out-of-sample Predictions

## Using new data

Using a new dataset:

A white box with red text

Description automatically generated

# Concluding Remarks

In conclusion, our analysis focused on the critical task of detecting encrypted malicious traffic within network data, addressing the challenges posed by the increasing use of encryption in cyber threats.

The Random Forest classifier, which was our machine learning model of choice, was justified due to its effectiveness, readability, and capacity to handle a variety of datasets without overfitting. Pipelines were used to streamline the workflow, which included feature selection, model training, and data preprocessing, to meet time constraints.

Finally, our study's approach to solving the problems of encrypted malicious traffic detection resulted in the creation of a machine learning model that is both successful and insightful, adding to the ongoing efforts to fortify cybersecurity defenses against ever-evolving threats.