**Pre Processing**

The aim of the pre processing stage is to analyse the raw dataset and make the necessary modifications to put it in the most effective conditions to perform the actual machine learning task.

Pre processing steps can be grouped into data cleaning, data reduction and data transformation.

Data Cleaning

As shown by Little and Rubin[2], It’s important to identify the presence of missing data and implement an appropriate policy to solve the issue. In this case the training set is complete.

The presence of duplicate cases can create a bias in the dataset causing the algorithm to learn improperly the persistence of a pattern. McKinney [3] highlights two options to deal with duplicates: keep them or discard them assuming they are redundant. The training set has 547 duplicate observations. Our assumption is that intrusion attacks are independent therefore duplication in this case is considered as persistence of certain patterns. Therefore we decided to keep the duplicates.

Data Reduction

This step aims to simplify the dataset eliminating features that are irrelevant at best and constitute noise at worst. Effective data reduction results in greater learning speed and potentially better accuracy.

Inspecting the standard deviation of the features matrix highlighted a minimum of zero (table 1). Looking at the value counts for each level of standard deviation (table 2) revealed that 74 predictors assume the exact identical value across every training instance. They are redundant, therefore have been eliminated using the *Scikit Learn VarianceThreshold* selector. The resulting dataset has been named *Xselected.*

Data Transformation

Different algorithms make an assumption about the probability distribution of data and or work best when it comes to optimizing parameters if input features are within a certain range.

Given that there isn’t an algorithm that performs better than others a priori on a given task, the output of the data transformation phase consists of several dataset to experiments on.

A descriptive statistics analysis of the train set shows that the predictors are heterogeneous: some variables appear discrete, others binary, others continuous. As suggested in Kolias *et al.* [4] and in Aminanto *et al*. [5] a normalization step has been performed so that every predictor’s value range is between zero and one. *Scikit Learn* *MinMaxScaler* has been applied to the *Xselected* dataset.

Using skew as a measure of symmetry around the mean of each feature, highlights that the respective frequency distributions deviate substantially from the Gaussian (table1). Kolias et al. [4] showed that the Naive Bayes Algorithm was the most accurate (albeit not to a satisfactory level) in detecting impersonation attack. This algorithm, when dealing with continuous features, assumes a Gaussian distribution of the predictor given a class.

Yeo and Johnson [6] proposed a power transformation to reduce skew that does not require specific underlying assumptions on the variable to be transformed.This power transformation has been implemented on the *Xselected* dataset using *Scikit Learn PowerTransformer.* As a result the parameters of each feature distribution are closer to a Gaussian.

A final and alternative transformation has been performed on the *Xselected* dataset. Given the heterogeneity of features, it’s useful to scale each instance to have a norm of one in case at a later stage of the project an algorithm like Neural Network that weights input values, or that uses kernels (for example an SVM) to quantify the similarity of different examples will be deployed.

This transformation has been implemented using *Scikit Learn Normalizer*.

At the end of the pre processing phase, four datasets are available to experiment with different feature and model selections:

1. X.selected: features with variance greater than zero
2. NX3df: Xselected with features value in range [0,1]
3. PX2df: Xselected with a power Gaussian tranformation
4. NX2df: Xselected with each training instance to have unit norm.

The best results have been achieved just removing the features with zero variance.