***Creating an Spam Email Filter with Machine Learning in R***

**ID: 10266499**

***Introduction***

Spam emails can be identified by variables such as the word frequency, character frequency and the amount of capital letters contained in the body of the email. This project will use R Studio to create three machine learning algorithms to correctly label the email as spam or ham (i.e. not spam). This is a binary classification problem. The data used in this study was taken from the UCI Machine Learning Repository Spambase data set (1999).

***Data Set Description***

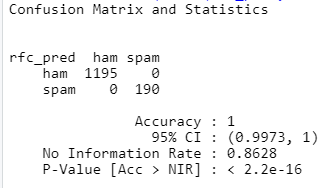
This dataset contains 5574 rows and two columns: the first variable is the content of the emails and the second variable the target variable, Class. Class is a categorical variable which is either “spam” or “ham”.

***Data Preparation***

During the data cleaning stage, all of the words were converted to lowercase, full stops were removed and non-English words were eliminated. The most frequently occurring words were also identified. Only the words which appeared more than 60 times were included in the final dataset. Finally, the dataset was split into a test and train subset.

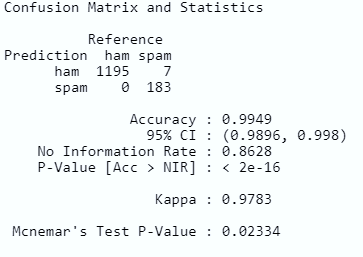
***Random Forest Classifier***

A Random Forest Model with 300 decision trees was created. Incredibly, the Random Forest model predicted the spam messages with an accuracy of 100% *(see Image 1).* This result suggests that the model may have been over-fitted on the dataset.

 ***Image 1: Random Forest Result***

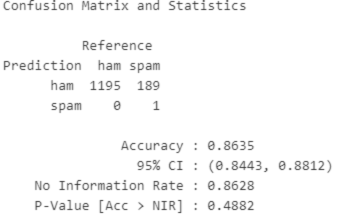
***Naive Bayes Classifier***

Next, aNaive Bayes Classifier Model was constructed. This model achieved 99.49% accuracy. Only 7 out of 1209 observations were misclassified (*see Image 2*). This result suggests that the Naive Bayes model was a strong model for predicting spam messages.

 ***Image 2: Naïve Bayes Result***

***Support Vector Machine***

Finally, aSupport Vector Machine (SVM) Model was created. This model was the least accurate out of the three models trialled in this project. The SVM model yielded an accuracy of 86.35%, but it had a specificity rate of 0% which means that the model did not provide a predictive value beyond random guessing (*see Image 3*).

** *Image 3: SVM Result***

***Conclusion***

The purpose of this study was to build a spam filter which could detect whether a message was spam or ham. The Random Forest was the most accurate model closely followed by the Naive Bayes model. However, the result of these models should be interpreted with caution due to suspected over-fitting. The SVM was found to be unreliable as a model for filtering spam messages due to its low specificity rate.

***References***

Hopkins, M., Reeber, E., Forman, G. and SuermondtSakar, J. (1999) *Spambase* [dataset], available at: <https://archive.ics.uci.edu/ml/datasets/Spambase>, date accessed: 21/05/2020