Additional Data Mission

**Group name:** The Mighty Boosh

**Individual responsibility by:** Panayiotis Petsas

# Data Motivation

Currently, the dataset that is used in the Iterative Development mission holds 5572 emails/sms messages. Out of all these 5572 emails, 4826 of those are considered ham emails (meaning that they are not malicious and are not spam). This means that only 746 emails from that dataset are actual spam emails. This signifies that for *almost* every 6 normal emails; 1 spam email is contained in the dataset. This leads to us not having enough spam emails to work with, which is an indication that the error percentage can be high, since the more the data, the more accurate the results from the training and having only 746 spam emails to analyse could bring that error margin to be high. Moreover, having more data in the dataset, also means more testing, given that the training/testing split % remains the same.

# Data Preparation

By doing some extra research, I found another dataset which contains more spam messages/emails which could be added to the Iterative Development mission. The dataset is by Hakan Ozler, from the website of Kaggle. This additional dataset contains 500 more unique spam emails, along with 2500 more ham emails. This would bring the total number of emails to 8572. The ham emails would be 7326 while the spam emails would be 1246. This brings the ham/spam email ratio from almost 6 to 1, to 5.8 to 1.

Reading a dataset (excel .csv file) in Python using Panda is through the command:

import pandas as pd

dataframe = pd.read\_csv("Data/sms\_messages.csv")

But since now we will not be just reading out of only one dataset but two, our code is going to be a little bit different. Instead it would be like:

import pandas as pd

import glob # We import the glob module, which will later help us in selecting all the .csv files

path = 'Data/'

all\_files = glob.glob(path + "/\*.csv")

for filename in all\_files:

dataframe = pd.read\_csv(filename, index\_col=None, header=0)

li.append(dataframe)

frame = pd.concat(li, axis=0, ignore\_index=True)

**Explanation:**

When many file names have a similar pattern or if we want to select all the files from a specific folder, the **glob** module from the Python standard library is very useful. I started by importing the function glob from the built-in glob module. I then used “**\*.csv**” to match any strings that end with the suffix ‘*.csv’.* The asterisk**(\*)** is a wild-card, that matches zero or more standard characters. The function glob uses the wild-card pattern to create an interval object which I named ‘**all\_files**’ containing all files that end with ‘.csv’ in the ‘**Data**’ folder. Finally, the iterable ***all\_files*** is consumed in a list comprehension that makes a list called ***dataframe*** containing the relevant datasets.

Link to the additional dataset used: <https://www.kaggle.com/ozlerhakan/spam-or-not-spam-dataset>

The additional dataset used is also included in the .zip file.

# Impact Report

Through this change, by implementing an extra dataset into the software, I was able to have more spam and ham emails/messages in the system. This means that since there is more data for the machine to learn from, the results of the program identifying whether a message is spam or not is more accurate. This is because in ML(Machine Learning), the more the data that is provided, the more accurate the end results always are and because of this, the margin for error is less.