PRML Assignment-3 **Name:Aditya Kumar(CS20M009),Aman Kumar(CS20M010)**

# Spam Classifier using Naive Bayes

## Dataset reference:

We have used the below metioned dataset to train our spam classifier model:

`http://openclassroom.stanford.edu/MainFolder/DocumentPage.php?course=MachineLearning&doc=exercises/ex6/ex6.html`

## Algorithm:

We are using naive bayes algorithm for training our spam classifier.The steps involved in this algorithm are as follows:

### i) Preprocessing:

In this step, we are pre-processing each mail present in the "train" folder. In the "train" folder, there are two sub folders "spam" and "non-spam".
We are then taking the raw emails provided to us and cleaning or pre-processing them such that all the special characters or any punctuation are removed and all the alphabets are changed into small. In other words, after the pre-processing stage, we'll only have words separated by spaces.
This pre-processing is also done to the test dataset.

### ii) Training the model:

While training our spam classifier model, we are creating CSV files, in order to store the all the vocabulary present in those emails, and we are also storing the frequency of their occurrences.
There are three CSV files that are being created during training the model:
a)"vocabulary.csv" : This file contains all the vocabularies and their corresponding frequencies present in both spam emails as well as non-spam emails.
b)"spam.csv" : This file contains as much as 5000 vocabularies and their corresponding frequencies that are present in the spam emails.
c)"non-spam.csv" : This file contains as much as 5000 vocabularies and their corresponding frequencies that are present in the non-spam emails.

### iii) Prediction of emails:

We are assuming the prior probabilities of P(spam)=0.5, and P(non-spam)=0.5.
Suppose an email contain n words($w_1, ....., w_n$).

Then, given those words, we are calculating the probabilities in the following ways:

$$P(spam/w_1 \cap w_2 \cap ..... \cap w_n) = \Pi_{i=1}^{n}(\frac{P(w_i/spam).P(spam)}{P(w_i)})$$
$$P(non-spam/w_1 \cap w_2 \cap ..... \cap w_n) = \Pi_{i=1}^{n}(\frac{P(w_i/non-spam).P(non-spam)}{P(w_i)})$$

We are calculating the probability of a given word in the following way:

$$P(word) = P(word/spam) \times P(spam) + P(word/non-spam) \times P(non-spam)$$

Suppose a new word is coming for the first time, then it will make P(word)=0, so we are using additive smoothing to avoid this problem.

At last, after comparing $P(spam/w_1 \cap w_2 \cap ..... \cap w_n)$ and $P(non-spam/w_1 \cap w_2 \cap ..... \cap w_n)$, we are predicting whether the given mail is spam or non-spam, i.e.,
if the $P(spam/w_1 \cap w_2 \cap ..... \cap w_n) > P(non-spam/w_1 \cap w_2 \cap ..... \cap w_n)$, then the email will be classified as spam, else non-spam.