HW5 REPORT

Contribution:

Risabh Baheti, Ankit Dewan - Worked on Naive Bayes equally.
Kushal Dhar, Ankur Rastogi- Worked on ID3 equally.

Naive Bayes Classifier:

In our implementation we have made the following functions:

- **train_data(wordSplit)**: This function takes in the split word list(for each line), checks whether it is ham or spam, add it the global word list containing all words and its corresponding ham or spam count..

- **calculateProbability(alpha)**: This method is applying laplacian smoothing with value alpha and calculating the probability of each word in spam or ham. It is calculated using the formula:

  - $Pr(word/spam) = \frac{(frequency\ of\ word\ in\ spam+alpha)}{(total\ words\ in\ spam+\ alpha*no.of\ unique\ words\ )}$

  Similarly it is done for ham

- **test_data(test_split)**: Here we send in the split list of test data. For each word we are calculating the spam and ham probability using the formula:

  - $Pr(email\ being\ spam) = \prod_{i} (Pr(word\_i\ /spam) * freq\ of\ word\_i) \ * \ Pr(spam)$

  Similarly for ham. We classify the email as spam or ham depending on which probability comes higher. Below is our table showing accuracy values for various alphas:

Value of Alpha(Laplacian Constant)	Accuracy(in %)
1	87.4
5	87.6
10	87.5
15	87.4
50	87.2
100	86.8

We obtained max accuracy for alpha=5 and our maximum accuracy is 87.6%