AI Assignment 4 Report

Bayes classifier results :-

Binary

Train :-

Created a dictionary of all unique words and their counts (if the word occured in a document, the count was incremented by 1) in spam and non spam documents

Test :-

Calculated probability of each word in test dataset spam and non spam by checking if the word occurs in dictionary generated by train mode. If word occurs, calculating probabilty as number of times the word occured in spam/nonspam documents divided by the total number of spam/nonspam documents. Assigning the tag of spam or ham to max of the probabilites calculated.

```
Bayes Binary ------------------------------------------------------
Confusion Matrix - [TN,FP],[FN,TP]
[1349, 20]
[22, 1163]
Accuracy 0.983555207518
------------------------------------------------------------------
```

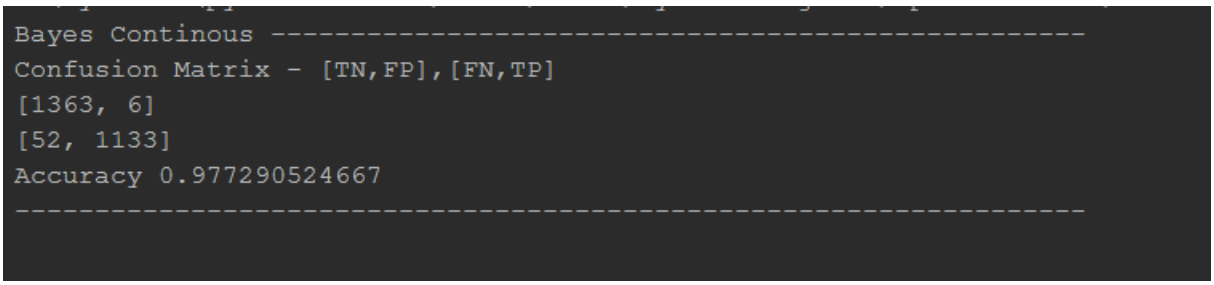
Continous

Train :-

Created a dictionary of all unique words and their counts(number of times they occured in the data, every instance of the word) in spam and non spam documents

Test :-

Calculated probability of each word in test dataset spam and non spam by checking if the word occurs in dictionary generated by train mode. If word occurs, calculating probabilty as number of times the word occured in spam documents divided by the total number of words in spam documents.Assigning the tag of spam or ham to max of the probabilites calculated.



Decision tree :-

Binary :-

Train :-

Creates a csv file of all the documents and all the unique words. If the word occured in the document, it's value is updated as one otherwise zero.

Continous :-

Creates a csv file of all the documents and all the unique words. Every instance of the word occurence in the document is tracked.

Binary/Continous

Test :-

Iterates through all the words which are features and finds best possible split by calculating information gain. Generates tree and classifies the test documents as spam or ham.

Decision tree results :-

Binary

```
Confusion Matrix :-
[1300, 32]
[69, 1153]
Accuracy : 0.960454189507
----------------------------------------
```

Continous

```
Confusion Matrix :-
[1170, 12]
[199, 1173]
Accuracy : 0.91738449491
--------------------------------
```

Example of a tree generated :-

```
Decision Tree binary ----------------------------------------------------
If Feature  hits and Value 0 :
Tree left->
If Feature  references and Value 0 :
  Tree left->
If Feature  example and Value 0 :
    Tree left->
If Feature  spambayes and Value 0 :
      Tree left->
If Feature  group and Value 0 :
        Tree left->
Result 1
        Tree right->
Result 1
      Tree right->
Result 0
    Tree right->
If Feature  tm and Value 0 :
      Tree left->
If Feature  subscribe and Value 0 :
        Tree left->
Result 1
        Tree right->
Result 0
      Tree right->
If Feature  jmason and Value 1 :
        Tree left->
Result 1
        Tree right->
Result 0
  Tree right->
If Feature  accounts and Value 0 :
    Tree left->
Result 0
    Tree right->
Result 1
Tree right->
Result 0
```

Based on the accuracy, we can say that naïve bayes classifier tends to perform better than the decision tree classifier on spam detection application.

Also decision tree binary mode performs better than continuous mode.

Top 10 Spam words :-

```
Words most associated with spam :-
x, mime, transfer, slashnull, dogma, single, drop, iso, click, labs

Words least associated with spam :-
mvfiaa, nasa, neale, newscientist, newsisfree, nmh, norealname, norte, nospaminc, zzzzteana
```

Pointers :-

Printing of tree in test mode for faster processing.

References:-

Few snippets of code for general preprocessing/dataframe slicing used from stackoverflow. Links posted as comments in code.

Used a stopword document to remove stopwords from the data. Links posted as comment in the code.