Assignment - 03
CS5691 Pattern Recognition and Machine Learning

Ramasamy Kandasamy
CS22M068

November 17, 2022

Data Information

To my model I used the **Apache spamassassin dataset**. This dataset is accessible from the following site: https://spamassassin.apache.org/old/publiccorpus/
For the train and testing the following files were used:

- `20030228_easy_ham.tar.bz2`
- `20030228_easy_ham_2.tar.bz2`
- `20030228_hard_ham.tar.bz2`
- `20030228_spam.tar.bz2`
- `20030228_spam_2.tar.bz2`

The above data were in the later steps split in 90:10 ratio for train and test. In addition to this I also used the following dataset to further test the accuracy of the classifier.

- `20021010_easy_ham.tar.bz2`
- `20021010_hard_ham.tar.bz2`
- `20050311_spam_2.tar.bz2`

The raw obtained was parsed using the `email` library and `text2html` library to extract the subject and body of the email in text format. This preprocessing is done using the script `preprocess_data.py` and `parse.py`. I then created the `bag_of_words.txt` which contains the words in sorted order occurring in the training/test datasets. I use this to extract feature vector from the input email text using the script `get_feature_vector.py`. The feature vector is an array of ones and zeroes indicating the presence or absence of a words.

Training procedure

For classification, I trained SVM without a kernel. I trained SVM with different values for C ranging from 0.1 to 10^8.
The accuracy of the model on two test sets is as follows:

```
First test set:
C = 0.1
Spam: 114/114
Ham: 276/278
C = 0.2
Spam: 114/114
Ham: 276/278
C = 0.3
Spam: 114/114
Ham: 276/278
C = 0.4
Spam: 114/114
Ham: 276/278
C = 0.5
Spam: 114/114
Ham: 276/278
C = 0.6
Spam: 114/114
Ham: 276/278
C = 0.7
Spam: 114/114
Ham: 276/278
C = 0.8
Spam: 114/114
Ham: 276/278
C = 0.9
Spam: 114/114
Ham: 276/278
C = 1.0
Spam: 114/114
Ham: 276/278
C = 100
Spam: 114/114
Ham: 276/278
C = 1000
Spam: 114/114
Ham: 276/278
C = 10000
Spam: 114/114
Ham: 276/278
C = 100000
Spam: 114/114
Ham: 276/278
C = 1000000
Spam: 114/114
Ham: 276/278
C = 10000000
Spam: 114/114
Ham: 276/278
C = 100000000
Spam: 114/114
Ham: 276/278
```

```
Second Test Set:
C = 1.0
Spam: 832/832
Ham: 1877/1879
C = 2.0
Spam: 832/832
Ham: 1877/1879
C = 3.0
Spam: 832/832
Ham: 1877/1879
C = 4.0
Spam: 832/832
Ham: 1877/1879
C = 5.0
Spam: 832/832
Ham: 1877/1879
C = 6.0
Spam: 832/832
Ham: 1877/1879
C = 7.0
Spam: 832/832
Ham: 1877/1879
C = 8.0
Spam: 832/832
Ham: 1877/1879
C = 9.0
Spam: 832/832
Ham: 1877/1879
C = 1.0
Spam: 832/832
Ham: 1877/1879
C = 10
Spam: 832/832
Ham: 1877/1879
C = 100
Spam: 832/832
Ham: 1877/1879
C = 1000
Spam: 832/832
Ham: 1877/1879
C = 10000
Spam: 832/832
Ham: 1877/1879
C = 100000
Spam: 832/832
Ham: 1877/1879
C = 1000000
Spam: 832/832
Ham: 1877/1879
C = 10000000
Spam: 832/832
Ham: 1877/1879
```

Prediction

The model is stored as `linear_0.1.svm`. I chose the model trained with `C = 0.1` because the accuracy nearly 100% for both spam and ham inputs. So, I chose one with lower C value because it would give higher weight to the $| w |^2$ term and therefore might generalize better.

To obtain prediction for some input test, please copy the test files to the `src/test` folder and run `python3 main.py`