Assignment 4: Spam or Ham

2022 Fall EECS205002 Linear Algebra

Due: 2023/1/11

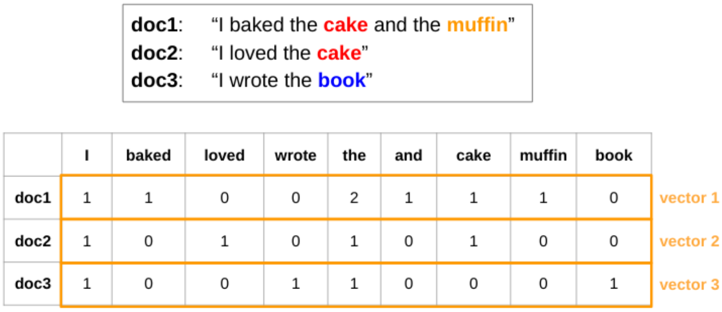
Junk emails are often called SPAM, whose name comes from a Monty Python sketch in which the name of the canned pork product Spam is ubiquitous, unavoidable, and repetitive.[1] On the other hand, HAM is referred to "E-mail that is generally desired and isn't considered spam," which was originally coined by SpamBayes sometime around 2001.

In this project, we want to use some basic techniques in Natural Language Process (NLP) to distinguish whether an email is a spam or a ham just from its contents. The idea is to collect a large dataset of emails, including spams and hams, and find their features that can distinguish them. One of the simple way is to form a document-term matrix, as shown in Figure 1[2], which has three documents. Each document is represented as a vector, which belongs to a *term* space. A term space is a very high dimensional space. Each term reprents a dimension. And a document is a point (vector) in that space. The numbers in the vector represents the *frequency*, which is how many times the term appears in the document.

Before the classification of spam and ham, we need to prepare document-term matrices for spam and ham. Suppose there are m_1 spam emails and m_2 ham emails, and the total number of terms is n. To make computation easy,

[1] https://www.merriam-webster.com/dictionary/spam

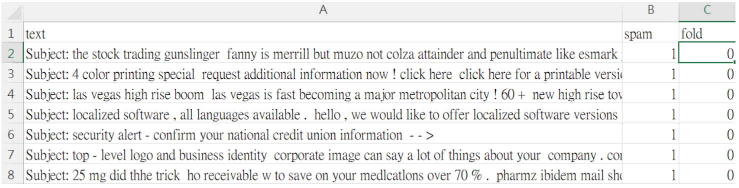
[2] https://rlads2021.github.io/LabBook/ch09



doc1: "I baked the **cake** and the **muffin**"
doc2: "I loved the **cake**"
doc3: "I wrote the **book**"

	I	**baked**	**loved**	**wrote**	**the**	**and**	**cake**	**muffin**	**book**	
doc1	1	1	0	0	2	1	1	1	0	vector 1
doc2	1	0	1	0	1	0	1	0	0	vector 2
doc3	1	0	0	1	1	0	0	0	1	vector 3

Figure 1: An example of document-term matrix.



	A	B	C
1	text	spam	fold
2	Subject: the stock trading gunslinger fanny is merrill but muzo not colza attainder and penultimate like esmark	1	0
3	Subject: 4 color printing special request additional information now ! click here click here for a printable versi	1	0
4	Subject: las vegas high rise boom las vegas is fast becoming a major metropolitan city ! 60 + new high rise to	1	0
5	Subject: localized software , all languages available . hello , we would like to offer localized software versions	1	0
6	Subject: security alert - confirm your national credit union information - - >	1	0
7	Subject: top - level logo and business identity corporate image can say a lot of things about your company . co	1	0
8	Subject: 25 mg did thhe trick ho receivable w to save on your medlcatlons over 70 % . pharmz ibidem mail sh	1	0

Figure 2: An example of the dataset.

we transpose the document-term matrix to a term-document matrix. We let S be an $n \times m_1$ term-document matrix for spam emails, and H be an $n \times m_2$ term-document matrix for ham emails.

To evaluate an email a spam or a ham, we first transform the email into a vector in the term space v. Then we can calculate the distance between v and the column space of S, called d_s, and the distance between v and the column space of H, called d_h. If $d_s > d_h$, it is a ham; otherwise it is a spam.

The distance from a vector b to a column space of a matrix A can be computed as follows. Let Ax be the projection of b onto A's column space, and the norm of their difference $r = Ax - b$, $\|r\|$, is the desired distance. Since Ax is a projection, so r is orthogonal to A's column vectors,

$$A^T r = A^T (Ax - b) = 0.$$

You can see that gives us the normal equation, $A^T Ax = A^T b$. So we can compute that using the least sqaure function.

We used the dataset from Kaggle [3], which is a website that collect various the codes and data, and holds competitions for data science. The dataset has three columns, text, spam, and fold, as shown in Figure 2. The text column contains the email context; the spam colum is either 1 (spam) or 0 (ham); the fold column has number from 0 to 4, which represents 5 folds. For each class (spam or ham), the data are evenly divided into 5 folds. We will use 4 folds of data for training and 1 fold of data for testing.

	Is a spam	Is a ham
Predicted as a spam	208 (TP)	1 (FP)
Predicted as a ham	73 (FN)	861 (TN)

The effectiveness of the spam filter can be analyzed through several methods. First, we can construct the confusion matrix as the following table. We use the fold 0-3 as training dataset and fold 4 as the test dataset. From it we can know

- The total number of test data is $n = 208 + 1 + 72 + 861 = 1142$.
- Each cell has a name:

[3] `https://www.kaggle.com/datasets/venky73/spam-mails-dataset`

- TP (true positive): It is a spam and the prediction is correct.
- FP (false positive): It is not a spam, and the prediction is wrong.
- TN (true negative): It is not a spam and the prediction is correct.
- FN (false negative): It is a spam, and the prediction is wrong.

- Accuracy: the ratio of correct predictions, which is

$$\text{Accuracy} = \frac{TP + TN}{N} = (208 + 861)/1142 = 93.6\%.$$

- Precision: from the predicted spams, how many of them are actial spams.

$$\text{Precision} = \frac{TP}{TP + FP} = 208/(208 + 1) = 99.5\%.$$

- Recall: from the actual spams, how many of them are predicted spams.

$$\text{Recall} = \frac{TP}{TP + FN} = 208/(208 + 72) = 74.3\%.$$

1 Assignments

1. (10%) Down load the data set, and find 5 spam emails you have received and put them into the data, one for each fold. And upload your spam emails to the website `https://forms.gle/6g15t9ELTC7T2cpC9`.

2. (10%) We use a python package `sklearn.feature_extraction.text` to construct the document-term matrix. Please study the following terms and explain their usage.

   (a) TFIDF

   (b) stop words

3. (20%) Use different folds for training/testing, and compute their confusion matrix, accuracy, precision, and recall.

4. (20%) Use SVD `numpy.linalg.svd` to construct the orthogonal basis for the column space of S and the column space of H, and use them to find the distances from the test data to the column spaces. You may check out the concepts in Chap 5.5 and Chap 6.5 of the textbook. Are the answers different from (3)? If so, explain why, and explain which one is more accurate.

5. (30%) In the example code, you see the recall is poor because FN is high. One way to reduce it is using the Latent Sematic Analysis (LSA). The idea of LSA is to explore the latent sematic from the document-term relation using the low-rank approximation. In this assignment, you need to do the following.

(a) Try different low rank approximation using SVD for S and H, such as (rank(S)=200 and rank(H)=600), for the train set = fold 0-3 and test set = fold 4. And compare the results (accuracy, precision, recall) with the results above.

(b) Google what "LSA" is and why it can improve the accuracy.

(c) How to find the best rank for S and for H such that the accuracy, precision, and recall can be optimized?

6. (10%) One problem for SVD is the basis U and V may contain negative numbers, which are hard to explain their meanings. One way to solve that is the non-negative matrix factorization. Google how to compute it and use it in the spam filter.

2 Submission

1. Write a report in PDF file that includes the answers of question (1)-(6).
2. The code of (3), (4), (5).
3. Zip them and submit the zip file to the eeclass.