SEPARATING SPAM FROM HAM (PART 2)

This homework assignment is the second part of the assignment from the previous page. Please complete Problems 1-4 on the previous page before starting this assignment. A description of the problem and the dataset can be found on the previous page.

PROBLEM 5.1 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (2 points possible)

Thus far, we have used a threshold of 0.5 as the cutoff for predicting that an email message is spam, and we have used accuracy as one of our measures of model quality. As we have previously learned, these are good choices when we have no preference for different types of errors (false positives vs. false negatives), but other choices might be better if we assign a higher cost to one type of error.

Consider the case of an email provider using the spam filter we have developed. The email provider moves all of the emails flagged as spam to a separate "Junk Email" folder, meaning those emails are not displayed in the main inbox. The emails not flagged as spam by the algorithm are displayed in the inbox. Many of this provider's email users never check the spam folder, so they will never see emails delivered there.

In this scenario, what is the cost associated with the model making a false negative error?

- A ham email will be sent to the Junk Email folder, potentially resulting in the email user never seeing that message.
- A spam email will be displayed in the main inbox, a nuisance for the email user.
- There is no cost associated with this sort of mistake.

In this scenario, what is the cost associated with our model making a false positive error?

- A ham email will be sent to the Junk Email folder, potentially resulting in the email user never seeing that message.
- A spam email will be displayed in the main inbox, a nuisance for the email user.
- There is no cost associated with this sort of mistake.

Show Answer *You have used 0 of 1 submissions*

PROBLEM 5.2 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (1 point possible)

Which sort of mistake is more costly (less desirable), assuming that the user will never check the Junk Email folder?

- False negative
- False positive
- They are equally costly

Show Answer You have used 0 of 1 submissions

PROBLEM 5.3 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (1 point possible)

What sort of user might assign a particularly high cost to a false negative result?

- A user who does not mind spam emails reaching their main inbox
- A user who is particularly annoyed by spam email reaching their main inbox
- A user who never checks their Junk Email folder
- A user who always checks their Junk Email folder

Show Answer *You have used 0 of 1 submissions*

PROBLEM 5.4 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (1 point possible)

What sort of user might assign a particularly high cost to a false positive result?

- A user who does not mind spam emails reaching his/her main inbox
- A user who is particularly annoyed by spam email reaching his/her main inbox
- A user who never checks his/her Junk Email folder
- A user who routinely checks his/her Junk Email folder

Show Answer *You have used 0 of 1 submissions*

PROBLEM 5.5 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (1 point possible)

Consider another use case for the spam filter, in which messages labeled as spam are still delivered to the main inbox but are flagged as "potential spam." Therefore, there is no risk of the email user missing an email regardless of whether it is flagged as spam. What is the largest way in which this change in spam filter design affects the costs of false negative and false positive results?

- The cost of false negative results is decreased
- The cost of false negative results is increased
- The cost of false positive results is decreased
- The cost of false positive results is increased

Show Answer *You have used 0 of 1 submissions*

PROBLEM 5.6 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS (1 point possible)

Consider a large-scale email provider with more than 100,000 customers. Which of the following represents an approach for approximating each customer's preferences between a false positive and false negative that is both practical and personalized?

- Use the expert opinion of a project manager to select the relative cost for all users
- Automatically collect information about how often each user accesses his/her Junk Email folder to infer preferences
- Survey a random sample of users to measure their preferences
- Survey all users to measure their preferences

Show Answer *You have used 0 of 2 submissions*

PROBLEM 6.1 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

While we have thus far mostly dealt with frequencies of specific words in our analysis, we can extract other information from text. The last two sections of this problem will deal with two other types of information we can extract.

First, we will use the number of words in the each email as an independent variable. We can use the original document term matrix called dtm for this task. The document term matrix has documents (in this case, emails) as its rows, terms (in this case word stems) as its columns, and frequencies as its values. As a result, the sum of all the elements in a row of the document term matrix is equal to the number of terms present in this document. Obtain the word counts for each email with the command:

wordCount = rowSums(as.matrix(dtm))

IMPORTANT NOTE: If you received an error message when running the command above, it might be because your computer ran out of memory when trying to convert dtm to a matrix. If this happened to you, try running the following lines of code instead to create wordCount (if you didn't get an error, you don't need to run these lines). This code is a little more cryptic, but is more memory efficient.

library(slam)

wordCount = rollup(dtm, 2, FUN=sum)$v

When you have successfully created wordCount, answer the following question.

What would have occurred if we had instead created wordCount using spdtm instead of dtm?

- ○ wordCount would have only counted some of the words and it would have only returned a result for some of the emails
- ○ wordCount would have counted all of the words, but would have only returned a result for some the emails
- ○ wordCount would have only counted some of the words, but would have returned a result for all the emails
- ○ wordCount would have counted all the words and it would have returned a result for all the emails

Show Answer *You have used 0 of 1 submissions*

PROBLEM 6.2 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

Use the hist() function to plot the distribution of wordCount in the dataset. What best describes the distribution of the data?

- ○ The data is skew right -- there are a large number of small wordCount values and a small number of large values.
- ○ The data is not skewed -- there are roughly the same number of unusually large and unusually small wordCount values.
- ○ The data is skew left -- there are a large number of large wordCount values and a small number of small values.

Show Answer *You have used 0 of 1 submissions*

PROBLEM 6.3 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

Now, use the hist() function to plot the distribution of log(wordCount) in the dataset. What best describes the distribution of the data?

- ○ The data is skew right -- there are a large number of small log(wordCount) values and a small number of large values.
- ○ The data is not skewed -- there are roughly the same number of unusually large and unusually small log(wordCount) values.

○ The data is skew left -- there are a large number of large log(wordCount) values and a small number of small values.

Show Answer *You have used 0 of 1 submissions*

PROBLEM 6.4 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

Create a variable called logWordCount in emailsSparse that is equal to log(wordCount). Use the boxplot() command to plot logWordCount against whether a message is spam. Which of the following best describes the box plot?

○ logWordCount is much smaller in spam messages than in ham messages
○ logWordCount is slightly smaller in spam messages than in ham messages
○ logWordCount is slightly larger in spam messages than in ham messages
○ logWordCount is much higher in spam messages than in ham messages

Show Answer *You have used 0 of 1 submissions*

PROBLEM 6.5 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

Because logWordCount differs between spam and ham messages, we hypothesize that it might be useful in predicting whether an email is spam. Take the following steps:

1) Use the same sample.split output you obtained earlier (do not re-run sample.split) to split emailsSparse into a training and testing set, which you should call train2 and test2.

2) Use train2 to train a CART tree with the default parameters, saving the model to the variable spam2CART.

3) Use train2 to train a random forest with the default parameters, saving the model to the variable spam2RF. Again, set the random seed to 123 directly before training spam2RF.

Was the new variable used in the new CART tree spam2CART?

○ Yes
○ No

Show Answer *You have used 0 of 1 submissions*

PROBLEM 6.6 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

Perform test-set predictions using the new CART and random forest models.

What is the test-set accuracy of spam2CART, using threshold 0.5 for predicting an email is spam?

Show Answer *You have used 0 of 3 submissions*

PROBLEM 6.7 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

What is the test-set AUC of spam2CART?

Show Answer *You have used 0 of 3 submissions*

PROBLEM 6.8 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

What is the test-set accuracy of spam2RF, using threshold 0.5 for predicting an email is spam? (Remember that you might get a different accuracy than us even if you set the seed, due to the random behavior of randomForest on some operating systems.)

Show Answer *You have used 0 of 3 submissions*

PROBLEM 6.9 - INTEGRATING WORD COUNT INFORMATION (1 point possible)

What is the test-set AUC of spam2RF? (Remember that you might get a different AUC than us even if you set the seed when building your model, due to the random behavior of randomForest on some operating systems.)

In this case, adding the logWordCounts variable did not result in improved results on the test set for the CART or random forest model.

Show Answer *You have used 0 of 3 submissions*

PROBLEM 7.1 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

Another source of information that might be extracted from text is the frequency of various n-grams. An n-gram is a sequence of n consecutive words in the document. For instance, for the document "Text analytics rocks!", which we would preprocess to "text analyt rock", the 1-grams are "text", "analyt", and "rock", the 2-grams are "text analyt" and "analyt rock", and the only 3-gram is "text analyt rock". n-grams are order-specific, meaning the 2-grams "text analyt" and "analyt text" are considered two separate n-grams. We can see that so far our analysis has been extracting only 1-grams.

In this last subproblem, we will add 2-grams to our predictive model. Begin by installing and loading the RTextTools package. We can create a document term matrix containing all 2-grams in our dataset using (be patient, as this may take a few minutes):

```
dtm2gram = create_matrix(as.character(corpus), ngramLength=2)
```

How many terms are in dtm2gram?

Show Answer You have used 0 of 3 submissions

PROBLEM 7.2 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

It's clearly more important than ever to remove terms that appear infrequently. Use removeSparseTerms to build a document term matrix spdtm2gram that contains only 2-grams appearing in at least 5% of the emails. How many terms are in spdtm2gram?

Show Answer You have used 0 of 3 submissions

PROBLEM 7.3 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

spdtm and spdtm2gram contain all 1-grams and 2-grams, respectively, that appear in at least 5% of the documents in our corpus. In this case, our corpus spdtm contains many more terms than spdtm2gram. Which of the following is true?

- For any corpus, spdtm constructed in this way will have as many or more terms than spdtm2gram.
- For some corpus, spdtm2gram constructed in this way will contain more terms than spdtm.

Show Answer You have used 0 of 1 submissions

PROBLEM 7.4 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

Now, let's include all the 2-grams in our spam/ham prediction models. Complete the following steps:

1) Build data frame emailsSparse2gram from spdtm2gram, using as.data.frame() and as.matrix().

2) Convert the column names of emailsSparse2gram to valid names using make.names().

3) Combine the original emailsSparse with emailsSparse2gram into a final data frame with the command "emailsCombined = cbind(emailsSparse, emailsSparse2gram)".

4) Use the same sample.split output you obtained earlier (do not re-run sample.split) to split emailsCombined into a training and testing set, which you should call trainCombined and testCombined.

5) Use trainCombined to train a CART tree with the default parameters, saving the model to the variable spamCARTcombined.

6) Use trainCombined to train a random forest with the default parameters, saving the model to the variable spamRFcombined. Again, set the random seed to 123 directly before training the random forest model.

How many 2-grams were used as splits in spamCARTcombined? A 2-gram is denoted by two words separated by a period or dot. You can pass the "varlen=0" option to the prp() function to display full variable names instead of truncated names.

Show Answer *You have used 0 of 5 submissions*

PROBLEM 7.5 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

Perform test-set predictions using the new CART and random forest models.

What is the test-set accuracy of spamCARTcombined, using a threshold of 0.5 for predictions?

Show Answer *You have used 0 of 5 submissions*

PROBLEM 7.6 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

What is the test-set AUC of spamCARTcombined?

Show Answer *You have used 0 of 3 submissions*

PROBLEM 7.7 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

What is the test-set accuracy of spamRFcombined, using a threshold of 0.5 for predictions? (Remember that you might get a different accuracy than us even if you set the seed, due to the random behavior of randomForest on some operating systems.)

Show Answer *You have used 0 of 5 submissions*

PROBLEM 7.8 - USING 2-GRAMS TO PREDICT SPAM (1 point possible)

What is the test-set AUC of spamRFcombined? (Remember that you might get a different AUC than us even if you set the seed before building the model, due to the random behavior of randomForest on some operating systems.)

For this problem, adding 2-grams did not dramatically improve our test-set performance. Adding n-grams is most effective in large datasets. Given the billions of emails sent each day, it's reasonable to expect that email providers would be able to construct datasets large enough for n-grams to provide useful predictive power.

Show Answer *You have used 0 of 3 submissions*

Please remember not to ask for or post complete answers to homework questions in this discussion forum.

Show Discussion New Post