MITx: 15.071x The Analytics Edge

IMPORTANT NOTE: This problem is optional, and will not count towards your grade. We have created this problem to give you extra practice with the topics covered in this unit.

SEPARATING SPAM FROM HAM (PART 2 - OPTIONAL)

This optional 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 problem, if you choose to do so. 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

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.

EXPLANATION

A false negative means the model labels a spam email as ham. This results in a spam email being displayed in the main inbox.

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.

EXPLANATION

A false positive means the model labels a ham email as spam. This results in a ham email being sent to the Junk Email folder.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 5.2 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS

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

EXPLANATION

A false negative is largely a nuisance (the user will need to delete the unsolicited email). However a false positive can be very costly, since the user might completely miss an important email due to it being delivered to the spam folder. Therefore, the false positive is more costly.

HIDE ANSWER

You have used 1 of 1 submissions

PROBLEM 5.3 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS

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

EXPLANATION

A false negative results in spam reaching a user's main inbox, which is a nuisance. A user who is particularly annoyed by such spam would assign a particularly high cost to a false negative.

HIDE ANSWER

You have used 1 of 1 submissions

PROBLEM 5.4 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS

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

EXPLANATION

A false positive results in ham being sent to a user's Junk Email folder. While the user might catch the mistake upon checking the Junk Email folder, users who never check this folder will miss the email, incurring a particularly high cost.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 5.5 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS

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

EXPLANATION

While before many users would completely miss a ham email labeled as spam (false positive), now users will not miss an email after this sort of mistake. As a result, the cost of a false positive has been decreased.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 5.6 - ASSIGNING WEIGHTS TO DIFFERENT TYPES OF ERRORS

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

EXPLANATION

While using expert opinion is practical, it is not personalized (we would use the same cost for all users). Likewise, a random sample of user preferences doesn't enable personalized costs for each user.

While a survey of all users would enable personalization, it is impractical to obtain survey results from all or most of the users.

While it's impractical to survey all users, it is easy to automatically collect their usage patterns. This could enable us to select higher regression thresholds for users who rarely check their Junk Email folder but lower thresholds for users who regularly check the folder.

FINAL CHECK **SAVE** HIDE ANSWER *You have used 1 of 2 submissions*

PROBLEM 6.1 - INTEGRATING WORD COUNT INFORMATION

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 the document corresponding to the row. 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

EXPLANATION

spdtm has had sparse terms removed, which means we have removed some of the columns but none of the rows from dtm. This means rowSums will still return a sum for each row (one for each email), but it will not have counted the frequencies of any uncommon words in the dataset. As a result, wordCount will only count some of the words.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 6.2 - INTEGRATING WORD COUNT INFORMATION

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.

EXPLANATION

From hist(wordCount), nearly all the observations are in the very left of the graph, representing small values. Therefore, this distribution is skew right.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 6.3 - INTEGRATING WORD COUNT INFORMATION

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.

EXPLANATION

From hist(log(wordCount)), the frequencies are quite balanced, suggesting log(wordCount) is not skewed.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 6.4 - INTEGRATING WORD COUNT INFORMATION

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 ✗

EXPLANATION

We can add the variable and obtain the plot with:

```
emailsSparse$logWordCount = log(wordCount)

boxplot(emailsSparse$logWordCount~emailsSparse$spam)
```

We can see that the 1st quartile, median, and 3rd quartiles are all slightly lower for spam messages than for ham messages.

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 6.5 - INTEGRATING WORD COUNT INFORMATION

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.

EXPLANATION

These steps can be performed with:

```
train2 = subset(emailsSparse, spl == TRUE)

test2 = subset(emailsSparse, spl == FALSE)

spam2CART = rpart(spam~., data=train2, method="class")

set.seed(123)

spam2RF = randomForest(spam~., data=train2)
```

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

● Yes ✓

○ No

EXPLANATION

From prp(spam2CART), we see that the logWordCount was integrated into the tree (it might only display as "logWord", because prp shortens some of the variable names when it outputs them).

HIDE ANSWER *You have used 1 of 1 submissions*

PROBLEM 6.6 - INTEGRATING WORD COUNT INFORMATION

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

EXPLANATION

This can be accomplished with:

predTest2CART = predict(spam2CART, newdata=test2)[,2]

predTest2RF = predict(spam2RF, newdata=test2, type="prob")[,2]

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

0.9301513 ✓ **Answer:** 0.9301513

EXPLANATION

This can be obtained with:

table(test2$spam, predTest2CART > 0.5)

The accuracy is (1214+384)/nrow(test2)

CHECK **SAVE** **HIDE ANSWER** *You have used 1 of 3 submissions*

PROBLEM 6.7 - INTEGRATING WORD COUNT INFORMATION

What is the test-set AUC of spam2CART?

0.9582438 ✓ **Answer:** 0.9582438

EXPLANATION

This can be obtained with:

```
predictionTest2CART = prediction(predTest2CART, test2$spam)
```

```
as.numeric(performance(predictionTest2CART, "auc")@y.values)
```

CHECK **SAVE** **HIDE ANSWER** *You have used 1 of 3 submissions*

PROBLEM 6.8 - INTEGRATING WORD COUNT INFORMATION

What is the test-set accuracy of spam2RF, using a threshold of 0.5 for predicting if 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.)

0.9772992 ✓ **Answer:** 0.9772992

EXPLANATION

This can be obtained with:

```
table(test2$spam, predTest2RF > 0.5)
```

The accuracy is (1296+383)/nrow(test2)

CHECK **SAVE** **HIDE ANSWER** *You have used 1 of 3 submissions*

PROBLEM 6.9 - INTEGRATING WORD COUNT INFORMATION

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.)

0.9980905 ✓ **Answer:** 0.9980905

EXPLANATION

This can be obtained with:

```
predictionTest2RF = prediction(predTest2RF, test2$spam)

as.numeric(performance(predictionTest2RF, "auc")@y.values)
```

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

CHECK **SAVE** HIDE ANSWER *You have used 1 of 3 submissions*

USING N-GRAMS

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.

We do not have exercises in this class covering n-grams, but if you are interested in learning more, the "RTextTools", "tau", "RWeka", and "textcat" packages in R are all good resources.

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

Show Discussion

New Post