math456 hw7

lgeel

April 2022

1 Introduction

spam $\leftarrow read.table(".. /input/spam-vs-nonspam-emails/spam.txt", header = T)$

Now I'm going to split the data into a training set and a test set but since the data isn't organized by spam and not spam I will randomize the rows in the dataset with this code:

set.seed(42) (This is to ensure that we can reproduce our data.)

rows $\leftarrow sample(nrow(spam))(Shufflestherowsorourdataset.)$

$spam\_ran \leftarrow spam[rows,](Callsthisnewdataframespam\_ran.)$

$spam\_ran$

Now I will split the data into a training set and a test set. 80% will go to training and the other 20% will go to the test set.

library(rsample)
split $\leftarrow initial_split(spam_ran,$prop = .80)
$spam\_train \leftarrow training(split)$
$spam\_test \leftarrow testing(split)$

Now I'll build the discriminant rule
library(MASS)
spam_full_lda $\leftarrow lda(yesno\ ., data = spam\_train)$
$spam\_full\_lda$

OUTPUT:
Call:
lda(yesno   ., data = spam_train)

Prior probabilities of groups:
n y

0.6 0.4

Group means:
crl.tot dollar bang money n000 make
n 166.7328 0.01097373 0.1144307 0.0159058 0.007762681 0.07166667
y 455.5516 0.17542867 0.5084368 0.2100815 0.256467391 0.14582880

Coefficients of linear discriminants:
LD1
crl.tot 0.0006492482
dollar 1.5313958461
bang 0.4604992618
money 0.7875070193
n000 1.5160260146
make 0.0799526977

This output shows us that 60% of the days in our training data correspond to non-spam e-mails whereas the remaining 40% correspond to spam emails.

Now I'm going to make a pairwise plot to see which variables are most important when detecting spam.

pairs(spam[1:6], main = "Spam Data", pch = 21, bg = c("Red","Blue")[unclass(spam$yesno)])

OUTPUT:
See OUTPUT1 attached file

library(MASS)
lda_fit ← $lda(yesno\ n000 + make, data = spam\_train)$
lda_fit

OUTPUT:
Call:
lda(yesno n000 + make, data = spam_train)

Prior probabilities of groups:
n y
0.6 0.4

Group means:
n000 make
n 0.007762681 0.07166667
y 0.256467391 0.14582880

Coefficients of linear discriminants:
LD1

n000 2.744939
make 0.809776

This output shows us that 60% of the days in our training data correspond to non-spam e-mails whereas the remaining 40% correspond to spam emails. We will now plot out linear discriminant function to see how effective it is by using the code.

library(klaR)
spam_yesno $\leftarrow as.factor(spam\$yesno)$
$partimat(x = spam[c("n000", "make")], grouping = spam\_yesno, method = "lda",$
$col.mean = 1, image.colors = c("grey", "white"), prec = 400)$

OUTPUT:
See OUTPUT2 attached file

I will predict if the observations in our test data is spam or not spam and compute the confusion matrix. lda_pred $\leftarrow predict(lda\_fit, spam\_test)table(spam\_test\$yesno, lda\_pred\$cla$

Output:

	n	y
n	567	13
y	252	89

This matrix shows us that we have a misclassification rate of (250+12)/(555+12+250+103)=0.28 (28%), so it correctly classifies 72% of the test observations. Going by the plot, it seems that QDA would not be useful in this case. Therefore we will use LDA. The data is clustered closely together so it's difficult to apply a discriminant rule. I don't think that there is a good discriminant rule to be used. The conclusion is that spam emails are getting better at looking like normal emails.