Logistic Regression

Ryan Hastings

March 2020

Homework for STA 210: Regression Analysis

Load packages

```r
library(tidyverse)
library(broom)
library(knitr)
library(pROC)
library(plotROC)
library(patchwork)
```

I use the email dataset to create a simple spam filter that uses characteristics of an email to determine if an email is considered spam.

We will use the following variables in the analysis:

- `spam`: Indicator for whether the email was spam.
- `to_multiple`: Indicator for whether the email was addressed to more than one recipient.
- `num_char`: The number of characters in the email, in thousands.
- `number`: Factor variable saying whether there was no number, a small number (under 1 million), or a big number.

First, I load the data, preview it, and change the type of some of the variables.

```r
email <- read_csv("data/email.csv")
glimpse(email)
```

```
## Rows: 3,921
## Columns: 21
## $ spam         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ to_multiple  <dbl> 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ from         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ cc           <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 2, 1, 0, 2, ...
## $ sent_email   <dbl> 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, ...
## $ time         <dttm> 2011-12-31 22:16:41, 2011-12-31 23:03:59, 2012-01-01 ...
## $ image        <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ attach       <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ dollar       <dbl> 0, 0, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 5, 0, ...
## $ winner       <chr> "no", "no", "no", "no", "no", "no", "no", "no", "no", ...
## $ inherit      <dbl> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ viagra       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ password     <dbl> 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...
## $ num_char     <dbl> 11.370, 10.504, 7.773, 13.256, 1.231, 1.091, 4.837, 7....
## $ line_breaks  <dbl> 202, 202, 192, 255, 29, 25, 193, 237, 69, 68, 25, 79, ...
## $ format       <dbl> 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, ...
## $ re_subj      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, ...
## $ exclaim_subj <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...
## $ urgent_subj  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ exclaim_mess <dbl> 0, 1, 6, 48, 1, 1, 1, 18, 1, 0, 2, 1, 0, 10, 4, 10, 20...
## $ number       <chr> "big", "small", "small", "small", "none", "none", "big...
```

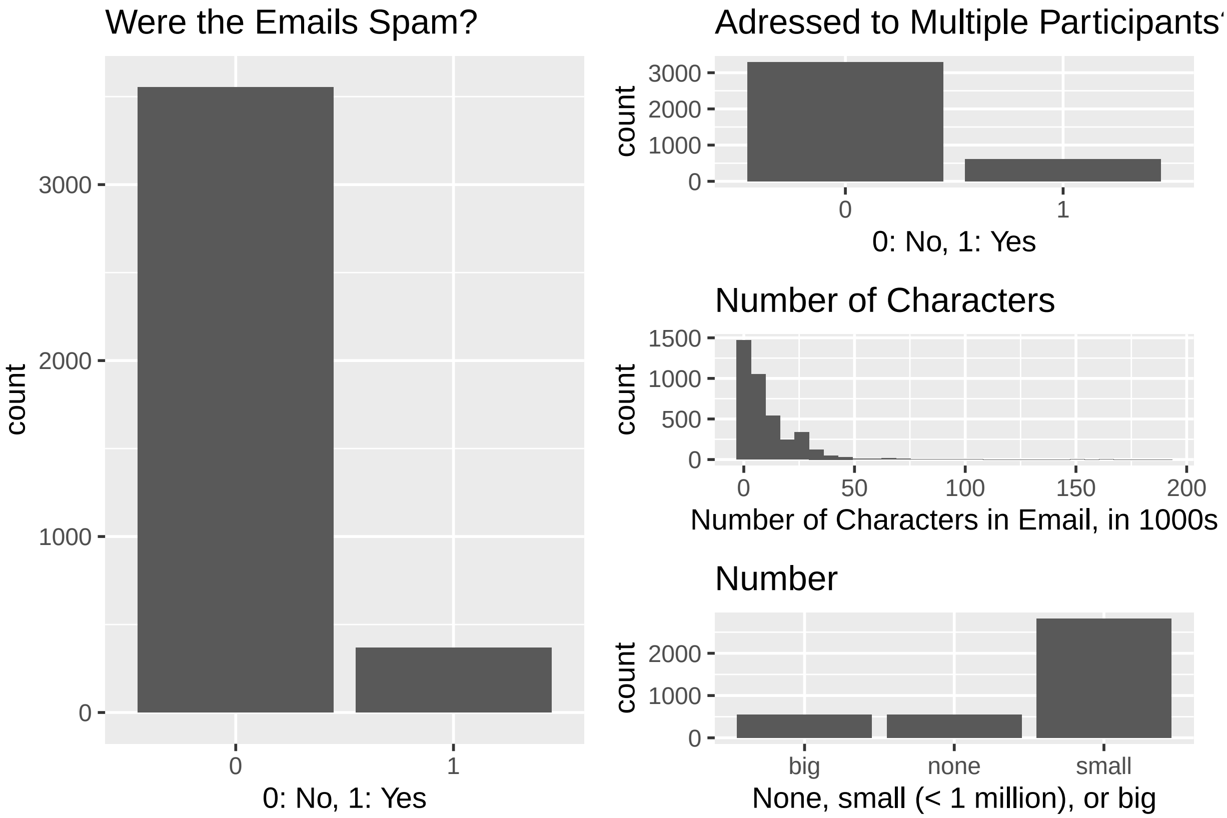
```
email = email %>%
  mutate(spam = as.factor(spam), to_multiple = as.factor(to_multiple))
```

Exploratory Data Analysis

Next, I conduct univariate and bivariate exploratory data analysis.

```
p1 <- email %>%
  ggplot()+
  geom_bar(mapping=aes(x=spam))+
  labs(title="Were the Emails Spam?", x="0: No, 1: Yes")
p2<- email %>%
  ggplot()+
  geom_bar(mapping=aes(x=to_multiple))+
  labs(title="Adressed to Multiple Participants?", x="0: No, 1: Yes")
p3 <- email %>%
  ggplot()+
  geom_histogram(mapping=aes(x=num_char))+
  labs(title="Number of Characters", x="Number of Characters in Email, in 1000s")
p4 <- email %>%
  ggplot()+
  geom_bar(mapping=aes(x=number))+
  labs(title="Number", x="None, small (< 1 million), or big")
p1 + (p2 / p3 / p4)
```

Univariate EDA



```
email %>%
  group_by(spam)%>%
  summarize(count=n())%>%
  kable(format="markdown")
```

spam	count
0	3554
1	367

```
email %>%
  group_by(to_multiple)%>%
  summarize(count=n())%>%
  kable(format="markdown")
```

to_multiple	count
0	3301
1	620

```
email %>%
  summarize(count=n(),
            median=median(num_char),
            iqr=IQR(num_char))%>%
  kable(format="markdown")
```

count	median	iqr
3921	5.856	12.625

```
email %>%
  group_by(number)%>%
  summarize(count=n())%>%
  kable(format="markdown")
```

number	count
big	545
none	549
small	2827

The distribution of spam is depicted, with 367 out of 3921 emails being marked as spam. There appear to be significantly more emails that are not spam.

The distribution of to_multiple is depicted, with 3301 being sent to only one recipient and 620 being sent to multiple recipients. There appear to be significantly more emails that are sent to only one recipient.

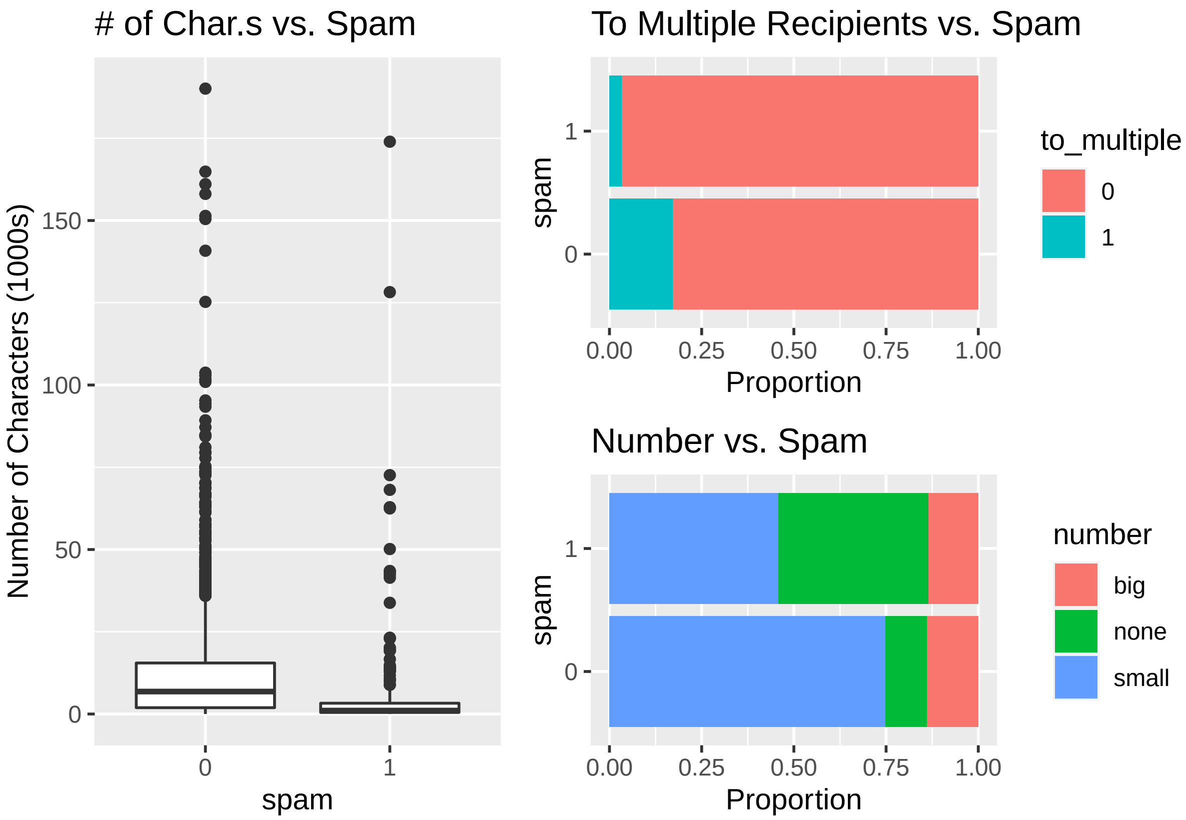
The distribution of num_char is depicted. The data appear to be skewed right, with a median of 5,856 characters and an IQR of 12,625 characters.

The distribution of number is depicted. There are 545 with a big number, 549 with no number, and 2827 with a small number. There appear to be significantly more with a small number than in the other two categories.

```


p6 <- ggplot(data = email, aes(x = spam, fill = to_multiple)) +
  geom_bar(position = "fill") +
  labs(y = "Proportion",
       title = "To Multiple Recipients vs. Spam") +
  coord_flip()
p7 <- ggplot(data = email, aes(x = spam, y = num_char)) +
  geom_boxplot() +
  labs(title="# of Char.s vs. Spam", x="spam", y="Number of Characters (1000s)")
p8 <- ggplot(data = email, aes(x = spam, fill = number)) +
  geom_bar(position = "fill") +
  labs(y = "Proportion",
       title = "Number vs. Spam") +
  coord_flip()
p7 + (p6 / p8)
```

Bivariate EDA



```
email %>%
  group_by(spam) %>%
  summarize(mean=mean(num_char),
            IQR= IQR(num_char)) %>%
  kable(format="markdown")
```

spam	mean	IQR
0	11.250517	13.58225
1	5.439204	2.81800

```
email %>%
  group_by(spam, to_multiple) %>%
  summarize(count=n()) %>%
  kable(format="markdown")
```

spam	to_multiple	count
0	0	2946
0	1	608
1	0	355
1	1	12

```
email %>%
  group_by(spam, number) %>%
  summarize(count=n()) %>%
  kable(format="markdown")
```

spam	number	count
0	big	495
0	none	400
0	small	2659
1	big	50
1	none	149
1	small	168

While the distribution of number of characters is skewed right for both spam and non-spam emails, the spam emails have a lower median number (5,439 vs. 11,250) and a lower IQR (2,818 vs. 13,582).

There appear to be a lower proportion of spam emails that are sent to multiple recipients. 12 out 367 spam emails were sent to multiple recipients, while 608 out of 3554 non-spam emails were sent to multiple recipients.

Comparing number, there appears to be a significantly higher proportion of spam emails with no number than the proportion of non-spam emails with no number. There also appears to be a significantly lower proportion of spam emails with a small number than non-spam emails with a small number. There appears to be a similar proportion of emails with a big number whether the email is spam or not.

Model & Drop-in-Deviance Test

Now, I fit a model with to_mutiple and num_char as the predictor variables. I use a drop-in-deviance test to determine if number should be included in the model.

```
m1 <- glm(spam ~ num_char + to_multiple, data = email, family = binomial)
m2 <- glm(spam ~ num_char + to_multiple + number, data = email, family = binomial)
tidy(m1, conf.int = TRUE, exponentiate = FALSE) %>%
  kable(format = 'markdown', digits = 3)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	-1.619	0.073	-22.167	0	-1.764	-1.477
num_char	-0.064	0.008	-8.018	0	-0.080	-0.049
to_multiple1	-1.888	0.298	-6.347	0	-2.527	-1.351

```
tidy(m2, conf.int = TRUE, exponentiate = FALSE) %>%
  kable(format = 'markdown', digits = 3)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	-1.558	0.170	-9.153	0	-1.900	-1.232
num_char	-0.038	0.008	-4.944	0	-0.054	-0.024
to_multiple1	-1.928	0.300	-6.428	0	-2.571	-1.385
numbernone	0.813	0.191	4.257	0	0.444	1.194

term	estimate	std.error	statistic	p.value	conf.low	conf.high
numbersmall	-0.707	0.172	-4.107	0	-1.038	-0.361

```
anova(m1, m2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: spam ~ num_char + to_multiple
## Model 2: spam ~ num_char + to_multiple + number
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1      3918     2274.6
## 2      3916     2148.0  2   126.62 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As the p value of the deviance statistic is less than 0.01, I choose to use the model including number. The data suggest that at least one of the coefficients on the levels in the 'number' variable is significantly different from zero.

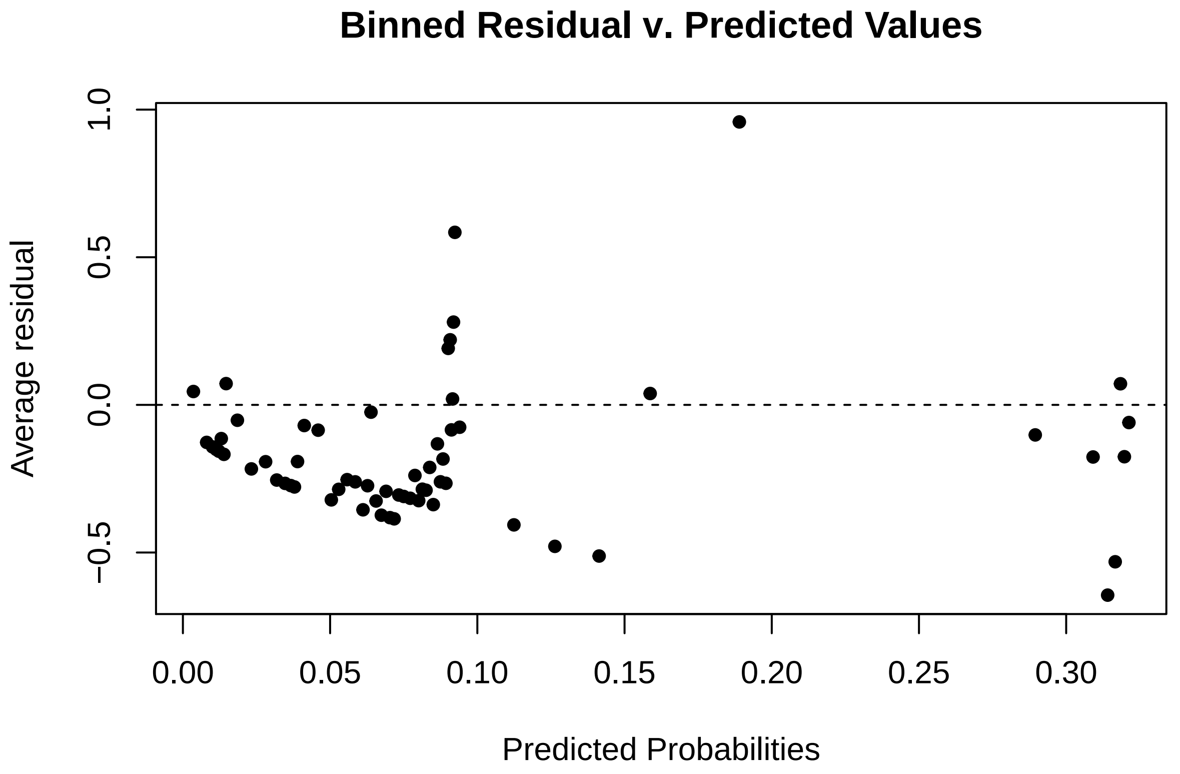
```
spam_m <- augment(m2, type.predict = "response", type.residuals = "deviance")
```

Assumptions

Now, I check the assumptions for this model.

```
arm::binnedplot(x=spam_m$.fitted,
                y=spam_m$.resid,
                xlab="Predicted Probabilities",
                main = "Binned Residual v. Predicted Values",
                col.int= FALSE)
```

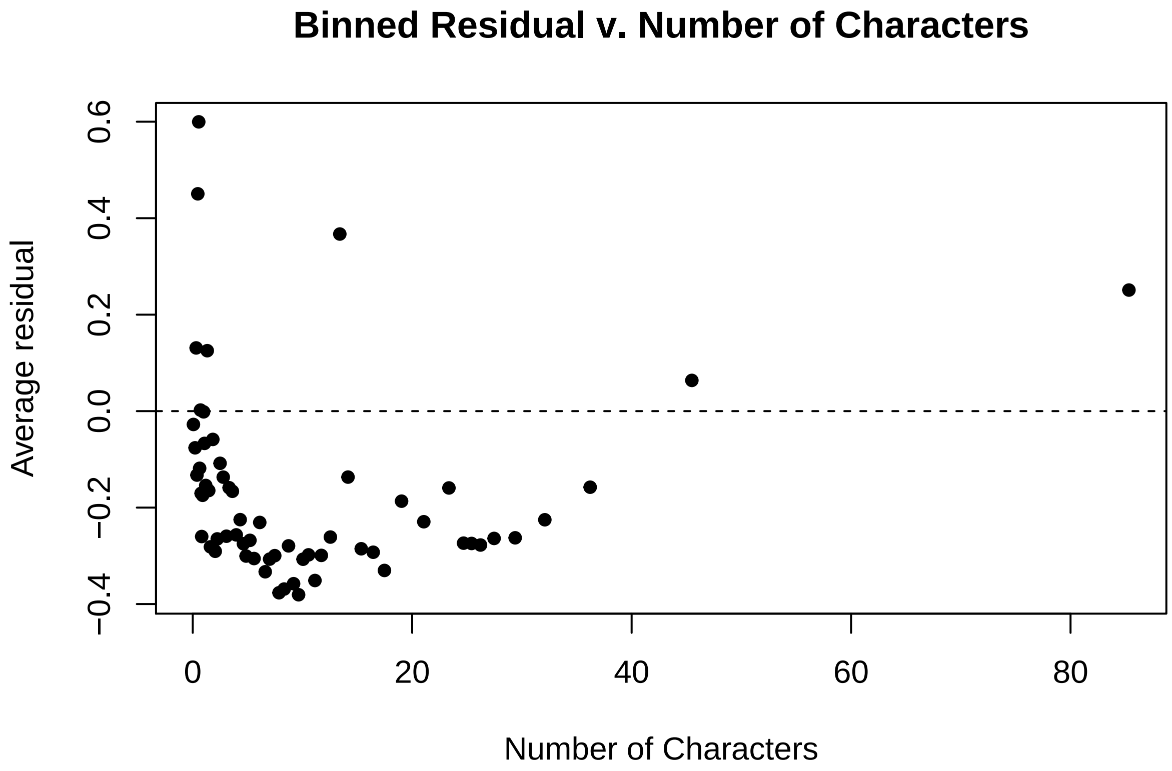
Linearity



```
spam_m %>%
  group_by(to_multiple) %>%
  summarise(mean_resid = mean(.resid))
```

```
## # A tibble: 2 x 2
##   to_multiple mean_resid
##   <fct>            <dbl>
## 1 0               -0.177
## 2 1               -0.131
```

```
arm::binnedplot(x=spam_m$num_char,
                y=spam_m$.resid,
                xlab="Number of Characters",
                main = "Binned Residual v. Number of Characters",
                col.int= FALSE)
```



```
spam_m %>%
  group_by(number) %>%
  summarise(mean_resid = mean(.resid))
```

```
## # A tibble: 3 x 2
##   number mean_resid
##   <chr>       <dbl>
## 1 big        -0.169
## 2 none       -0.127
## 3 small      -0.178
```

While the mean residuals across levels of number and to_multiple are similar and all approximately zero, the graphs of the residuals of the binned residuals vs. num_char and the binned residuals vs. the predicted probabilities have clear patterns (they do not appear randomly scattered), the linearity assumption is violated. However, I will continue with my analysis for now and might consider nonlinear transformations later.

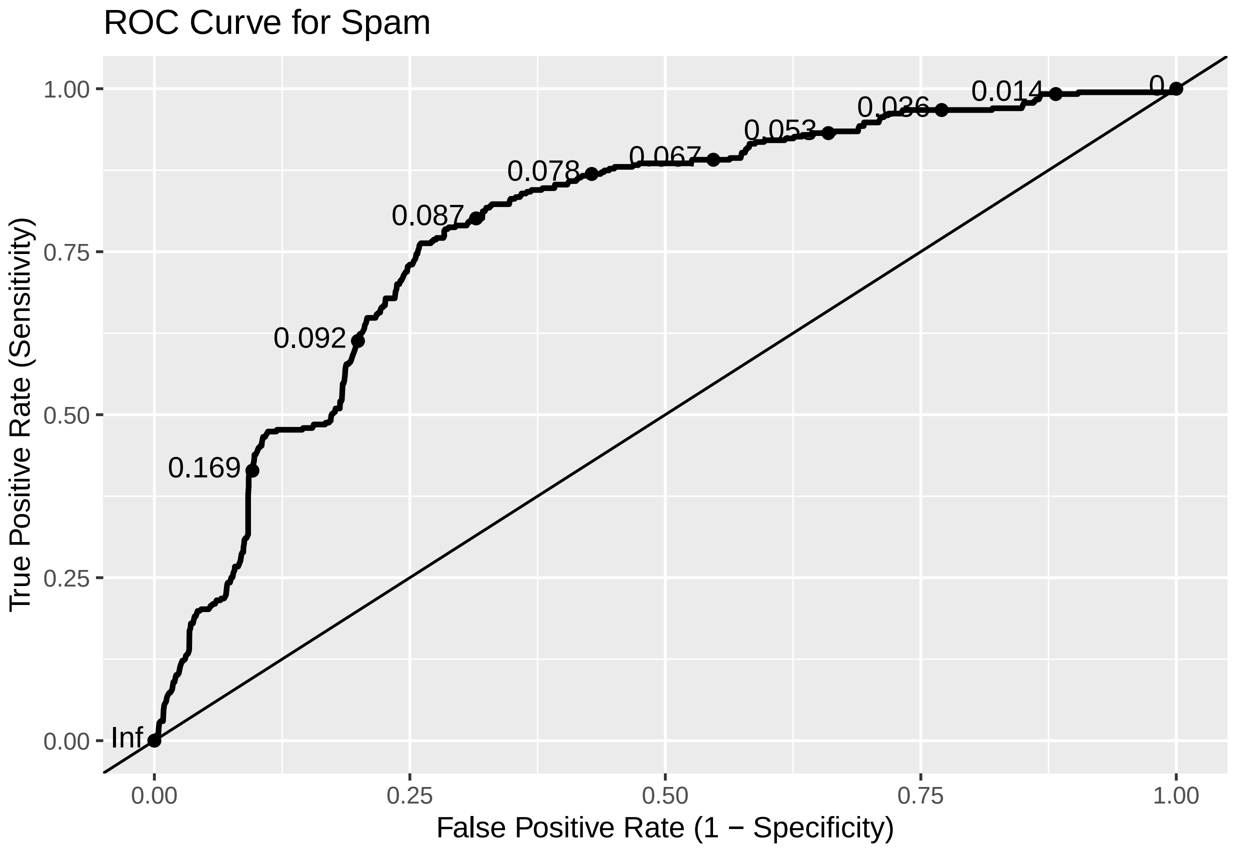
Independence These data are collected over three months in 2012, so I would have to graph the variables over time to be sure that the values of the analyzed variables for emails received close in time to each other are independent.

Randomization I have no reason to believe that this sample is not representative of the broader population of emails this individual receives, so the randomization assumption is reasonably met.

Using the Model to Filter Spam

Now, I make an ROC curve and calculate the AUC for the model.

```
roc_curve <- spam_m %>%
  ggplot(aes(d = as.numeric(spam) - 1, m = .fitted)) +
  geom_roc(n.cuts = 10, labelround = 3) +
  geom_abline(intercept = 0) +
  labs(x = "False Positive Rate (1 - Specificity)",
       y = "True Positive Rate (Sensitivity)",
       title = "ROC Curve for Spam")
roc_curve
```



```
calc_auc(roc_curve)$AUC
```

```
## [1] 0.7875315
```

The AUC is 0.787.

If I were a data scientist developing a spam filter, I might choose a threshold of 0.092. I would want to maximize the amount of spam emails that are counted as spam. However, I would not want many non-spam emails to be marked as spam, as the recipient may not see these emails or check this folder. As such, I choose a threshold that has a high sensitivity and a low false positive rate. The confusion matrix for such a threshold is displayed below.

```
threshold <- 0.092
spam_m %>%
  mutate(spam_predict = if_else(.fitted > threshold, "1: Yes", "0: No")) %>%
  group_by(spam, spam_predict) %>%
  summarise(n = n()) %>%
  kable(format="markdown")
```

spam	spam_predict	n
0	0: No	2867
0	1: Yes	687
1	0: No	151
1	1: Yes	216