Project 4 - Document Classification

CUNY MSDA DATA 607

Duubar Villalobos Jimenez mydvtech@gmail.com

April 16, 2017

Figure 1:

PROJECT 4: Document Classification

It can be useful to be able to classify new "test" documents using already classified "training" documents. A common example is using a corpus of labeled spam and ham (non-spam) e-mails to predict whether or not a new document is spam.

For this project, you can start with a spam/ham dataset, then predict the class of new documents (either withheld from the training dataset or from another source such as your own spam folder). One example corpus: https://spamassassin.apache.org/publiccorpus/

Workspace preparation

Create vector with all needed libraries.

```
load_packages <- c(
                    "knitr",
                    "R.utils",
                    "tm",
                    "wordcloud",
```

```
                    "topicmodels",
                    "SnowballC",
                    "e1071",
                    "data.table",
                    "RMySQL",
                    "tidyverse",
                    "tidyr",
                    "dplyr",
                    "stringr",
                    "stats"
                  )
```

Selected datasets

The selected datasets selected are as follows:

```
url.spam <- "http://spamassassin.apache.org/old/publiccorpus/"
file.spam <- "20050311_spam_2.tar.bz2"

url.ham <- "http://spamassassin.apache.org/old/publiccorpus/"
file.ham <- "20030228_easy_ham.tar.bz2"
```

Preparing datasets

Download

Function to download the desired files

```
downloadTAR <- function(filetype=NULL, myurl=NULL, myrootfile=NULL){

  destfile <- paste(filetype,".tar", sep="")

  if(!file.exists(destfile)){
      myfile <- paste(myurl,myrootfile,sep="")
      destfile <- paste(filetype,".tar.bz2", sep="")

      download.file(myfile, destfile= destfile)

      bunzip2(destfile)
      # untar(destfile)
  }

  mycompresedfilenames <- untar(destfile, list = TRUE)
  return(mycompresedfilenames)
}

spamFileNames <- downloadTAR("Spam", url.spam, file.spam)
hamFileNames <- downloadTAR("Ham", url.ham, file.ham)
```

Obtaining file names

```
spamfiles <- str_trim(str_replace_all(spamFileNames, "spam_2/", ""))
hamFiles <- str_trim(str_replace_all(hamFileNames, "easy_ham/", ""))

spamfiles <- subset(spamfiles, nchar(spamfiles) == 38)
hamfiles <- subset(hamFiles , nchar(hamFiles) == 38)
```

Read contents

```
readFileContents <- function(importtype=NULL, filenames=NULL){

  if (importtype == "Spam") {
    globalcon <- paste("C:/Users/mydvtech/Documents/GitHub/MSDA/Spring-2017/607/Projects/Project4/spam_
  }
  if (importtype == "Ham") {
    globalcon <- paste("C:/Users/mydvtech/Documents/GitHub/MSDA/Spring-2017/607/Projects/Project4/easy_
  }
  temp <- data.frame(stringsAsFactors = FALSE)

  mydata <- matrix()

  for(i in 1:length(filenames)){
    con <- file(globalcon[i], "r", blocking = FALSE)
     temp <- readLines(con)
    close(con)
    temp <- str_c(temp, collapse = "")
    temp <- as.data.frame(temp, stringsAsFactors = FALSE)
    names(temp) <- "Content"
    mydata[[i]] <- temp
  }

  return(mydata)
}

spams <- readFileContents("Spam", spamfiles)
hams <- readFileContents("Ham", hamfiles)
```

Some results

The total number of known spams are: 1396.

The total number of known hams are: 2500.

Grand total of Emails: 3896.

Sample emails

Spam

Ham

```
From aifrik@corpusmail.com  Fri Jun 29 02:51:20 2001
Return-Path: <aifrik@corpusmail.com>
Delivered-To: yyyy@netnoteinc.com
Received: from smtp.easydns.com (ns1.easydns.com [216.220.40.243]) by
    mail.netnoteinc.com (Postfix) with ESMTP id 70599130028; Fri,
    29 Jun 2001 02:51:18 +0100 (IST)
Received: from egon.instakom.ch (client197-202.hispeed.ch [62.2.197.202])
    by smtp.easydns.com (8.11.3/8.11.0) with ESMTP id f5T1pEa11156;
    Thu, 28 Jun 2001 21:51:14 -0400
Received: from Artic.net (ip-129-9.newgen.net.ph [202.171.129.9]) by
    egon.instakom.ch with SMTP (Microsoft Exchange Internet Mail Service
    Version 5.5.2653.13) id NLZTAG1Q; Fri, 29 Jun 2001 03:48:31 +0200
Message-Id: <0000382d3858$0000403d$00007ce9@Artic.net>
To: <174@portugalmail.com>
From: aifrik@corpusmail.com
Subject: FW:
Date: Thu, 28 Jun 2001 16:58:52 -0700
MIME-Version: 1.0
Content-Transfer-Encoding: quoted-printable
X-Priority: 3
X-Msmail-Priority: Normal


<HTML>
<BODY bgColor=3D#000000>

<FONT face=3D"Times New Roman">
<FONT size=3D3>
<FONT color=3D"#FF0000"><B> Would you like to</B></FONT>
<FONT color=3D"#FFFF00"><B>  look and feel 10-20 years younger</B></FONT>
<FONT color=3D"#FF0000"><B> ? <BR>
<BR>
Would you be interested in</B></FONT>
<FONT color=3D"#FFFF00"><B>  increasing energy levels</B></FONT>
<FONT color=3D"#FF0000"><B>  by</B></FONT>
<FONT color=3D"#FFFF00"><B>  84%</B></FONT>
<FONT color=3D"#FF0000"><B> ?</B></FONT>
<FONT size=3D2>
<FONT color=3D"#000080">  </FONT>
<FONT size=3D2>
<FONT color=3D"#804040"><I> 15x</I></FONT>
<FONT size=3D2>
<FONT color=3D"#804040"><B> <BR>
</B></FONT>
<FONT size=3D3>
<FONT color=3D"#FF0000"><B> How about</B></FONT>
<FONT size=3D2>
<FONT color=3D"#FFFF00"><B>  Increasing Sexual Potency Frequency</B></FONT=
>
<FONT color=3D"#FF0000"><B>  by </B></FONT>
<FONT color=3D"#FFFF00"><B> 75%</B></FONT>
<FONT color=3D"#FF0000"><B> ? <BR>
```

Figure 2:

```
From fork-admin@xent.com  Fri Aug 23 11:09:00 2002
Return-Path: <fork-admin@xent.com>
Delivered-To: zzzz@localhost.netnoteinc.com
Received: from localhost (localhost [127.0.0.1])
        by phobos.labs.netnoteinc.com (Postfix) with ESMTP id 6C8BB44161
        for <zzzz@localhost>; Fri, 23 Aug 2002 06:06:57 -0400 (EDT)
Received: from phobos [127.0.0.1]
        by localhost with IMAP (fetchmail-5.9.0)
        for zzzz@localhost (single-drop); Fri, 23 Aug 2002 11:06:57 +0100 (IST)
Received: from xent.com ([64.161.22.236]) by dogma.slashnull.org
    (8.11.6/8.11.6) with ESMTP id g7N8CWZ15864 for <zzzz@spamassassin.taint.org>;
    Fri, 23 Aug 2002 09:12:32 +0100
Received: from lair.xent.com (localhost [127.0.0.1]) by xent.com (Postfix)
    with ESMTP id 2DDA329418E; Fri, 23 Aug 2002 01:10:10 -0700 (PDT)
Delivered-To: fork@spamassassin.taint.org
Received: from hughes-fe01.direcway.com (hughes-fe01.direcway.com
    [66.82.20.91]) by xent.com (Postfix) with ESMTP id 0EC36294099 for
    <fork@xent.com>; Fri, 23 Aug 2002 01:09:30 -0700 (PDT)
Received: from spinnaker ([64.157.38.84]) by hughes-fe01.direcway.com
    (InterMail vK.4.04.00.00 201-232-137 license
    dcc4e84cb8fc01ca8f8654c982ec8526) with ESMTP id
    <20020823081149.JPJZ17240.hughes-fe01@spinnaker> for <fork@xent.com>;
    Fri, 23 Aug 2002 04:11:49 -0400
Subject: Re: Entrepreneurs
Content-Type: text/plain; charset=US-ASCII; format=flowed
MIME-Version: 1.0 (Apple Message framework v482)
From: Chuck Murcko <chuck@topsail.org>
To: fork@spamassassin.taint.org
Content-Transfer-Encoding: 7bit
In-Reply-To: <20020822205834.D7039C44E@argote.ch>
Message-Id: <DD4216FF-B66F-11D6-837F-003065F93D3A@topsail.org>
X-Mailer: Apple Mail (2.482)
Sender: fork-admin@xent.com
Errors-To: fork-admin@xent.com
X-Beenthere: fork@spamassassin.taint.org
X-Mailman-Version: 2.0.11
Precedence: bulk
List-Help: <mailto:fork-request@xent.com?subject=help>
List-Post: <mailto:fork@spamassassin.taint.org>
List-Subscribe: <http://xent.com/mailman/listinfo/fork>, <mailto:fork-request@xent.com?subject=subscribe>
List-Id: Friends of Rohit Khare <fork.xent.com>
List-Unsubscribe: <http://xent.com/mailman/listinfo/fork>,
    <mailto:fork-request@xent.com?subject=unsubscribe>
List-Archive: <http://xent.com/pipermail/fork/>
Date: Fri, 23 Aug 2002 01:11:02 -0700

According to my son, it was actually Homer Simpson, who claimed the
French had no word for victory.

Chuck

On Thursday, August 22, 2002, at 01:58 PM, Robert Harley wrote:

> An apparent quote from Dubya, from the Times (sent to me by my Dad):
>
> http://www.timesonline.co.uk/printFriendly/0,,1-43-351083,00.html
```

Figure 3:

Analysis

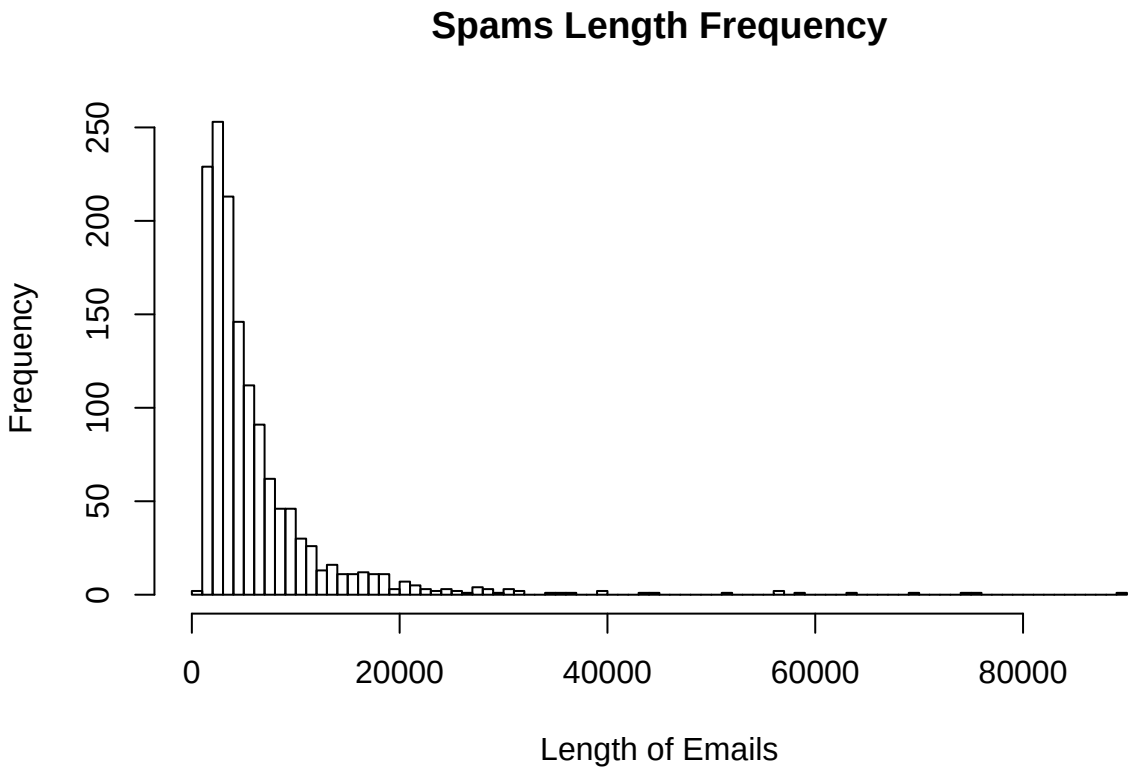
Lenght of Email

Spams Statistics

Summary

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     725    2458    4004    6183    7020   89210
```

Distribution

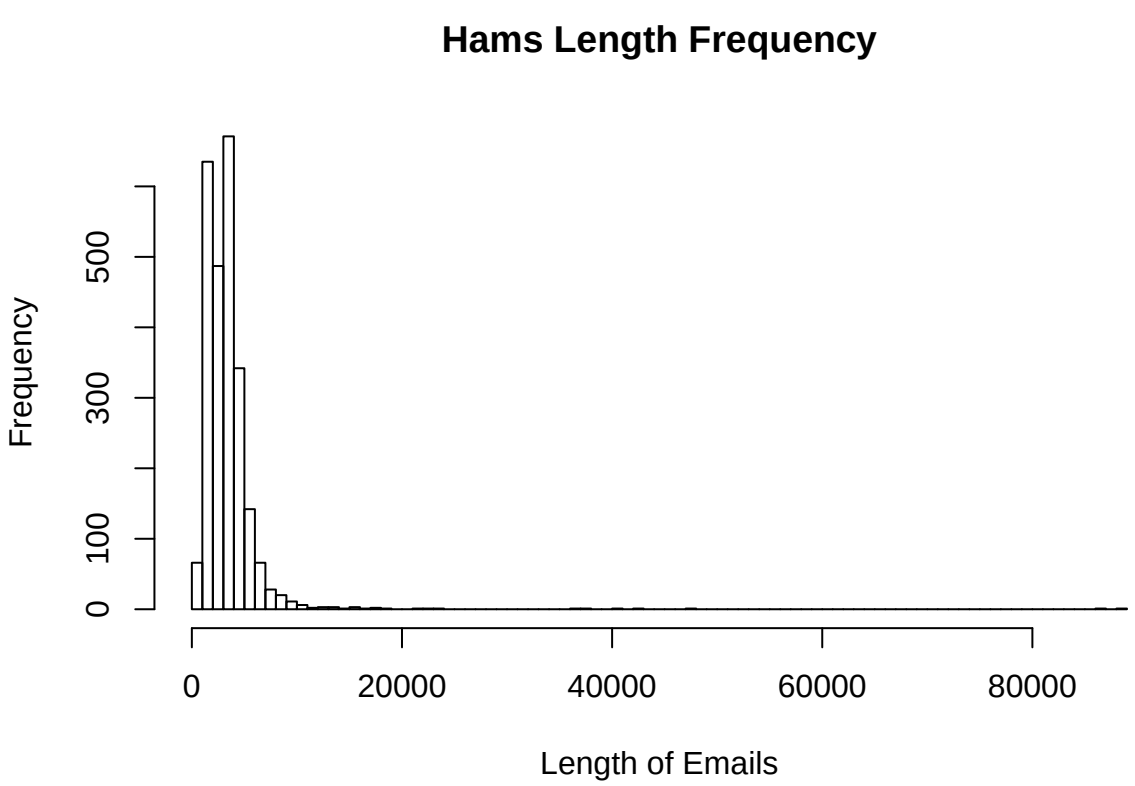


Hams Summary Statistics

Summary

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     355    1644    3081    3364    4039   88590
```

Distribution



Median Length

By running this analysis we can find out that in our pool of known ham spam emails; the Spam emails tend to have a longer Median length compared to Ham emails; that is as follows:

Median Length of Spams: 4004.

Median Length of Hams: 3081.

Difference of medians: 923.

Percentage difference: 29.96%.

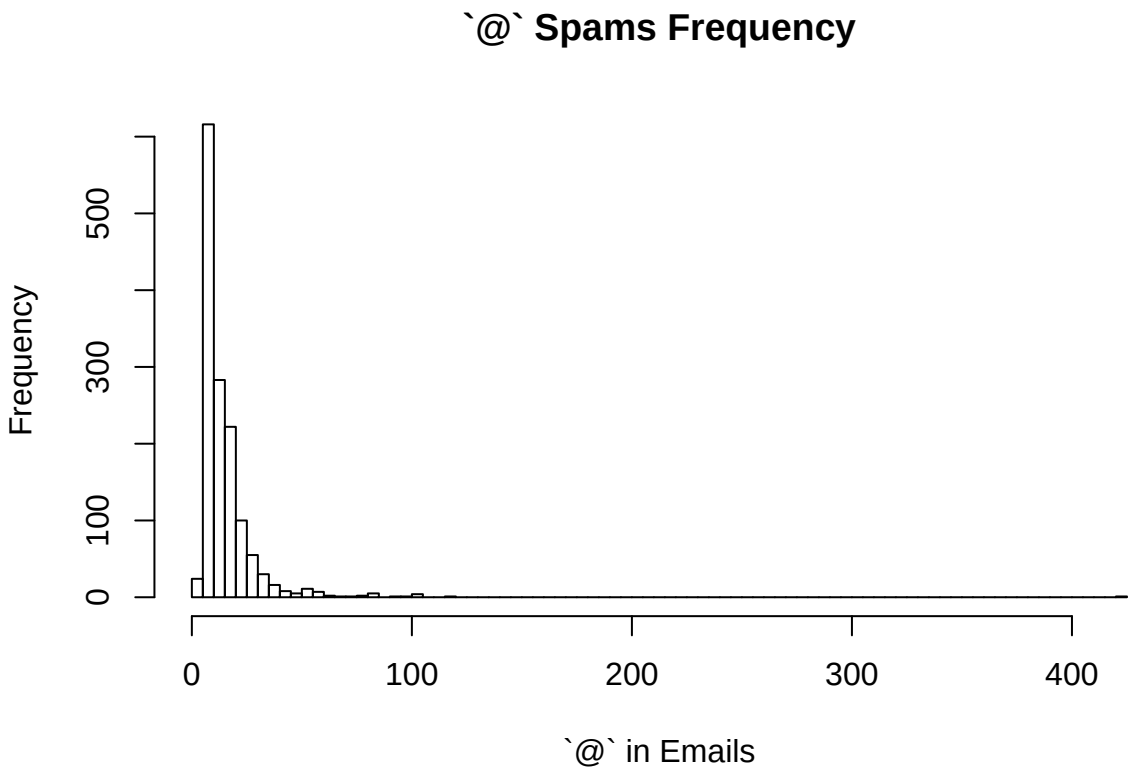
@ Analysis

@ Spams

Summary

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     3.0     9.0    11.0    15.6    19.0   423.0
```

Distribution

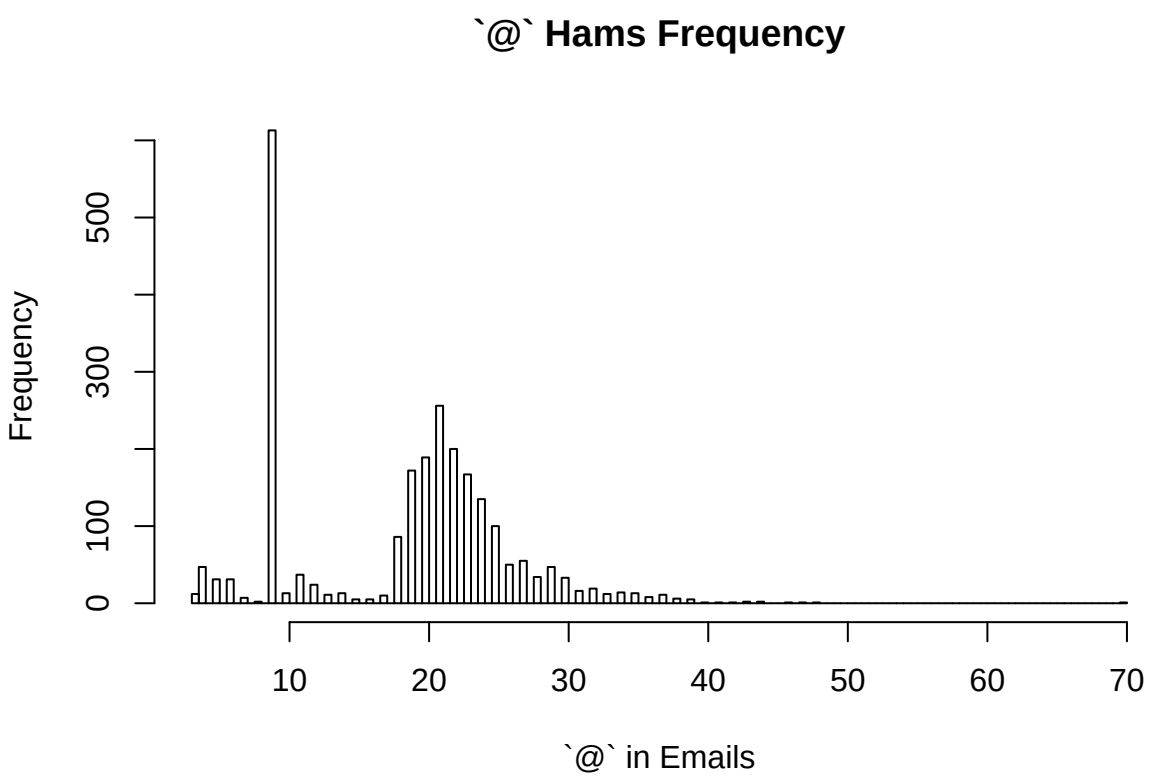


@ Hams

Summary

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.00    9.00   20.00   18.29   23.00   70.00
```

Distribution



@ Median analysis

By running this analysis we can find out that in our pool of known ham spam emails; the Spam emails tend to have a lower Median count of "@" compared to Ham emails; that is as follows:

Median Length of Spams: 11.

Median Length of Hams: 20.

Difference of medians: -9.

Percentage difference: -45%.

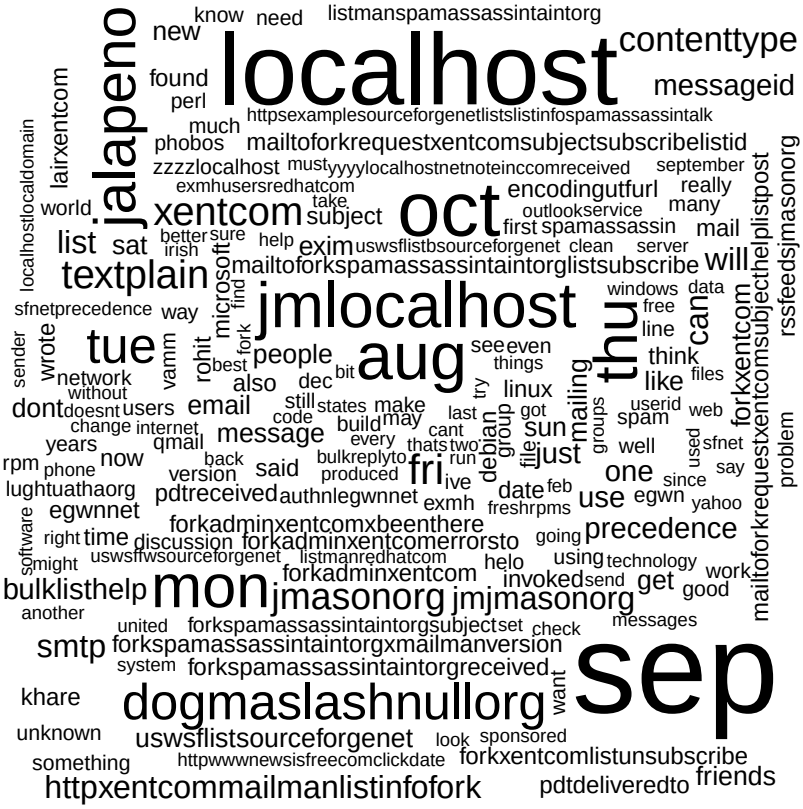
This can be probably concluded as accurate since work and personal emails tend to cc a lot of people while spams are targeted to small audiences in the beginning.

Wordclouds

Spam


facedverdana
cypherpunksdspronsnet
good
fetchmail
report
sat
life
aligndcenterfont
helvetica
date
credit
webnotenet
companies
outlook
valignd top
dont
much
cash
check
now
internet
colspand
business
colord
mailing
div
need
start
ffffff
widthdfont
sans
one
removed
new
einsteinsszcom
government
cdt
product
products
yyyylocalhostnetnoteinccomreceived
exchange
rates
dogmaslashnullorg
mail
alignleft
money
bgcolorffffff
mimeversion
sized
aug
many
xentcom
insurance
roman
wantweb
fri
inputfaceverdanafont
without
center
take
remove
service
envelopefrom
contact
border
see
like
work
future
html
pfont
times
first
way
typedtext
height
click
reply
postfix
save
right
bits
email
made
unknown
days
even
within
imap
arial
get
years
site
list
form
per
help
colordffffff
link
version
call
fax
low
cellpaddingd
card
special
cellspacingd
size
sales
will
make
img
sizeda
message
best
express
amp
jul
colspan
color
edtreceived
tue
today
find
colordff
name
order
facedarial
mon
phone
just
colorff
search
tbody
widthd
available
year
emails
people
subject
sites
million
software
marketing
cellpadding
font
also
option
locustmindernet
bfont
sizedfont
sansseriffont
wish
sent
month
online
heightd
fontfont
sansserif
localhost
number
orders
back
wastemindernet

Ham



Training data

Divide corpus into training and test data

Use 75% training and 25% test.

```
# Randomize emails order
random_emails <- emails_df[sample(nrow(emails_df)),]
NEmailsQ <- dim(random_emails)[1]/4*3
NEmails <- dim(random_emails)[1]

random_emails_train <- random_emails[1:NEmailsQ,]
random_emails_test <- random_emails[NEmailsQ+1:NEmails,]

# Document-term matrix and clean corpus
emails_corpus_train <- clean_corpus[1:NEmailsQ]
emails_corpus_test <- clean_corpus[NEmailsQ+1:NEmails]


# Text to Matrix in order to Tokenize the corpus
emails_dtm_train <- DocumentTermMatrix(emails_corpus_train)
emails_dtm_train <- removeSparseTerms(emails_dtm_train, 1-(10/length(release_corpus)))

emails_dtm_test <- DocumentTermMatrix(emails_corpus_test)
emails_dtm_test <- removeSparseTerms(emails_dtm_test, 1-(10/length(release_corpus)))
```

```
emails_tdm_train <- TermDocumentMatrix(emails_corpus_train)
emails_tdm_train <- removeSparseTerms(emails_tdm_train, 1-(10/length(release_corpus)))

emails_tdm_test <- TermDocumentMatrix(emails_corpus_test)
emails_tdm_test <- removeSparseTerms(emails_tdm_test, 1-(10/length(release_corpus)))



five_times_words <- findFreqTerms(emails_dtm_train, 5)
```

Create document-term matrices using frequent words

```
emails_train <- DocumentTermMatrix(emails_corpus_train, control=list(dictionary = five_times_words))
emails_test <- DocumentTermMatrix(emails_corpus_test, control=list(dictionary = five_times_words))
```

Convert count information to "Yes", "No"

Naive Bayes classification needs present or absent info on each word in a message. We have counts of occurrences. Convert the document-term matrices.

```
convert_count <- function(x) {
  y <- ifelse(x > 0, 1,0)
  y <- factor(y, levels=c(0,1), labels=c("No", "Yes"))
  y
}
```

```
emails_train <- apply(emails_train, 2, convert_count)
emails_test <- apply(emails_test, 2, convert_count)
```

The Naive Bayes function

We'll use a Naive Bayes classifier provided in the package e1071.

```
emails_classifier <- naiveBayes(emails_train, factor(random_emails_train$type))
class(emails_classifier)
```

```
## [1] "naiveBayes"
```

```
# emails_test_pred <- predict(emails_classifier, newdata=emails_test)
```

Unfortunatelly this requires a lot of resources from my PC and ran out of memory; hense I can't present the final reults.