Can we predict whether a received email message is spam or not?

Choose the data set

The ideal data set would be every email sent and received, but this is impossible, due to lack of time, space, and security issues. So, instead, let us use the sample of spam and notspam emails collected by a UCI study, available in the kernlab package of R:

```
library(kernlab)
data(spam)
```

Perform the subsampling

First, let's subsample, so we take the dataset and split it into a training and a test set, using a binomial distribution (i.e. flipping a coin)

```
set.seed(3435)
trainIndicator = rbinom(4601, size=1, prob = 0.5)
table(trainIndicator)
```

```
## trainIndicator
##    0    1
## 2314 2287
```

Let's call the 0 set the training set, and the 1 set the test set.

```
trainSpam = spam[trainIndicator == 1, ]
testSpam = spam[trainIndicator == 0, ]
```

Exploratory data analysis

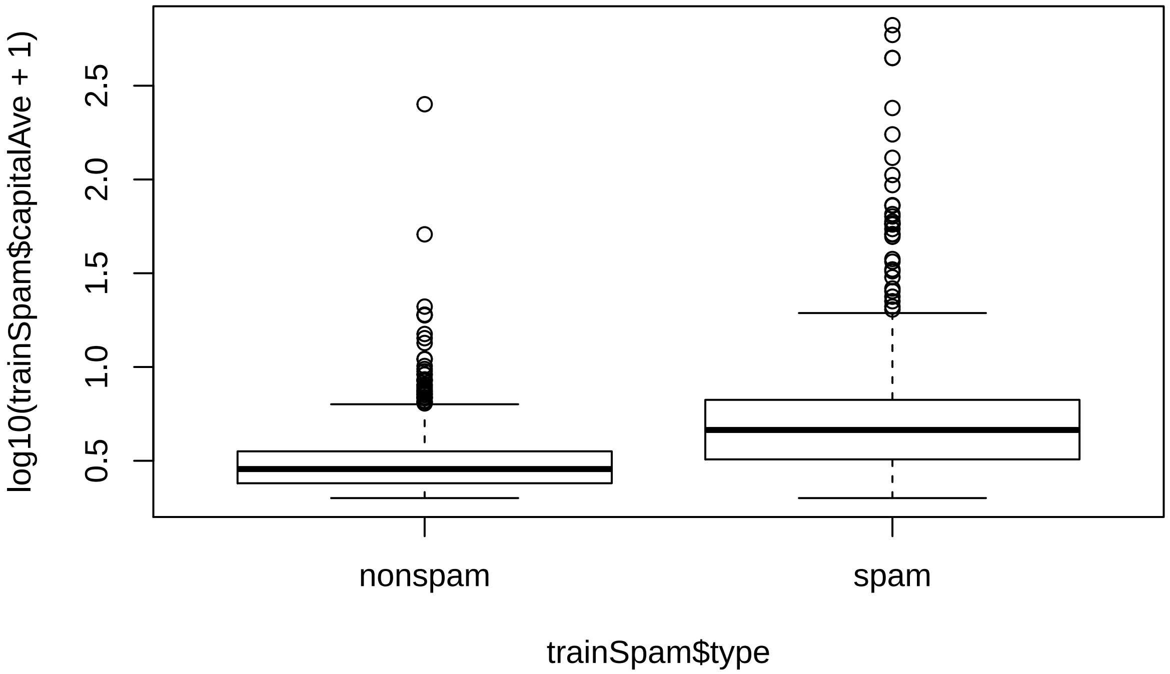
Looking at the training data set, let's see how it is divided into spam and notspam:

```
table(trainSpam$type)
```

```
##
## nonspam    spam
##    1381     906
```

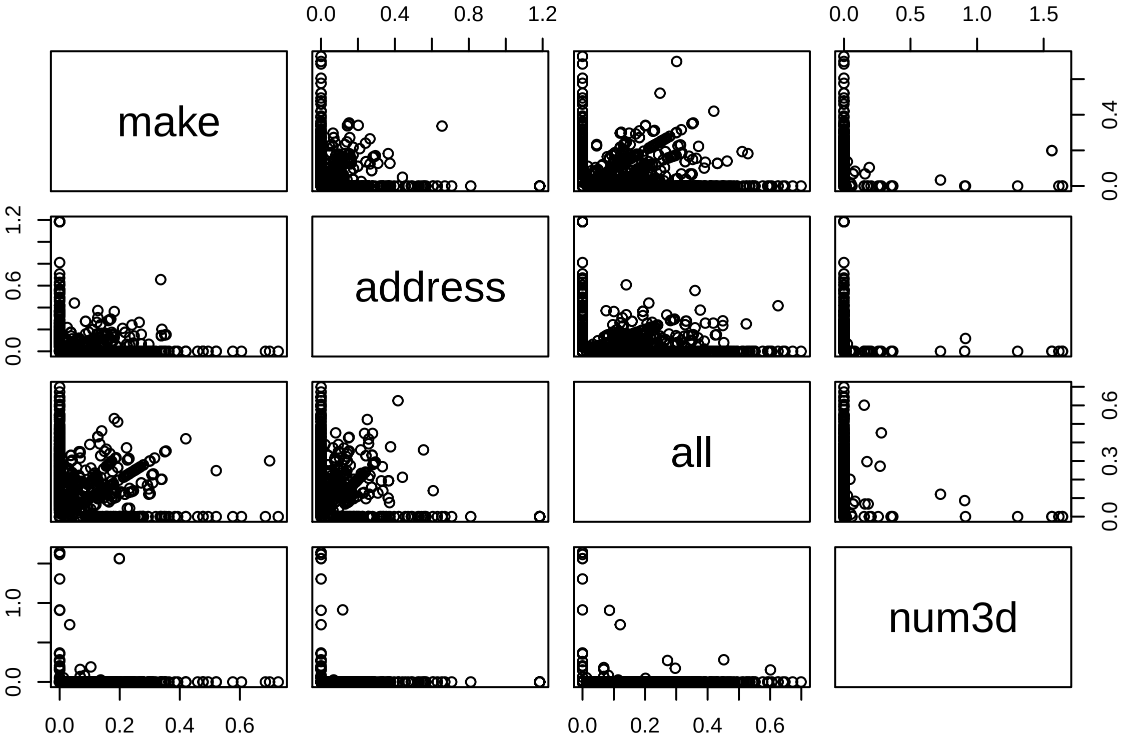
What is a good first indicator of a spam message... the number of capital letters? Let's see: does the averagee number of capital letters correlate with whether the email is spam?

```
plot(log10(trainSpam$capitalAve+1)~ trainSpam$type)
```



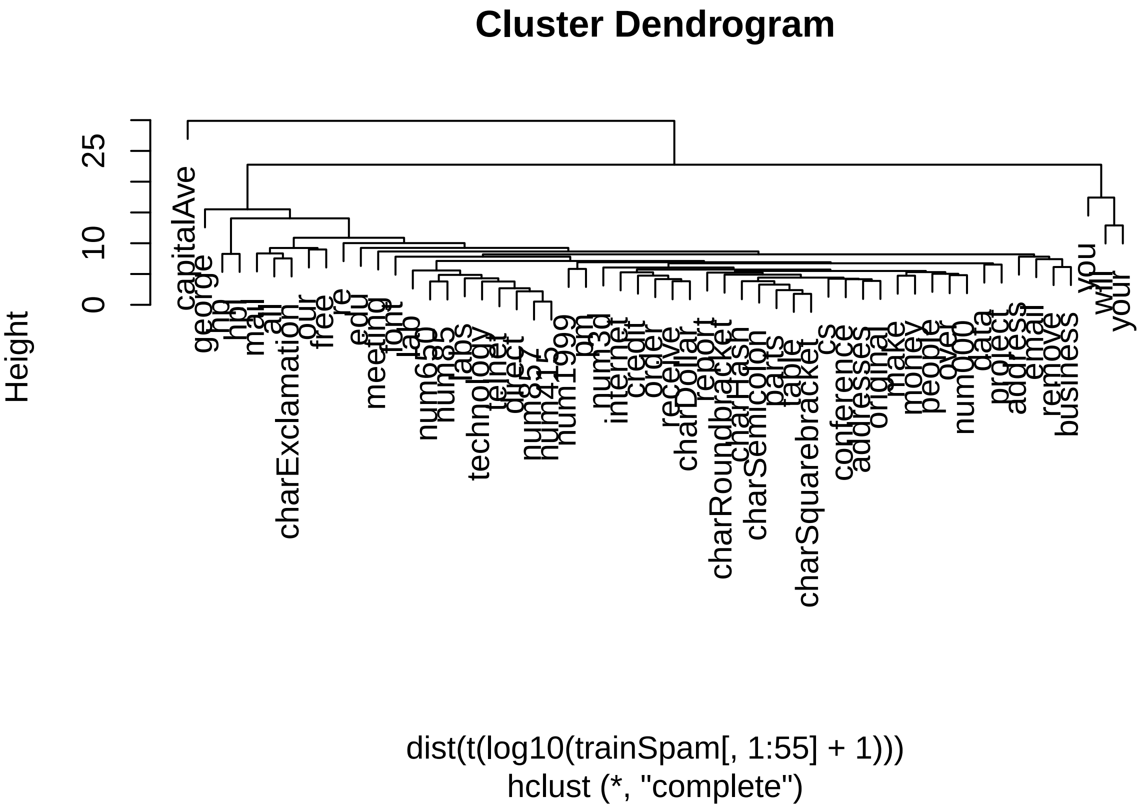
Yes, it seems so. We can look at other variables, and see some correlations, and some with no correlation.

```r
plot(log10(trainSpam[, 1:4]+1))
```



Let us see how the variables tend to cluster together:

```r
hClusterUpdated = hclust(dist(t(log10(trainSpam[, 1:55] + 1))))
plot(hClusterUpdated)
```



We see some clustering, for example “you,” “we”, “your”, tend to go hand-in-hand. # Statistical prediction/modeling We choose the following model as a first estimate, a generalized linear model, to cycle through the variables one at a time, and see if we can predict whether or not an email is spam by using just a single variable. Then, we calculate the cross validated error rate of predicting spam emails from a single variable.

```
trainSpam$numType = as.numeric(trainSpam$type) - 1
costFunction = function(x,y) sum(x != (y > 0.5))
cvError = rep(NA, 55)
library(boot)
for (i in 1:55){
    lmFormula = reformulate(names(trainSpam)[i], response = "numType")
    glmFit = glm(lmFormula, family = "binomial", data = trainSpam)
    cvError[i] = cv.glm(trainSpam, glmFit, costFunction, 2)$delta[2]
}
```

The predictor with the minumum cross-validated error is

```
names(trainSpam)[which.min(cvError)]
```

```
## [1] "charDollar"
```

Now, let’s use the best predictor model from the group, namely the count of dollar sign characters in the email

```
predictionModel = glm(numType ~ charDollar, family = "binomial", data = trainSpam)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

We now get predictions on the test set

```
predictionTest = predict(predictionModel, testSpam)
predictedSpam = rep("nonspam", dim(testSpam)[1])
```

and classify “as spam” for those with prob > 0.5

```
predictedSpam[predictionModel$fitted > 0.5] = "spam"
```

How did we do? Compare the predictions versus the known spam/nospam

```
table(predictedSpam, testSpam$type)
```

```
##
## predictedSpam nonspam spam
##       nonspam    1346  458
##       spam         61  449
```

Error rate is then

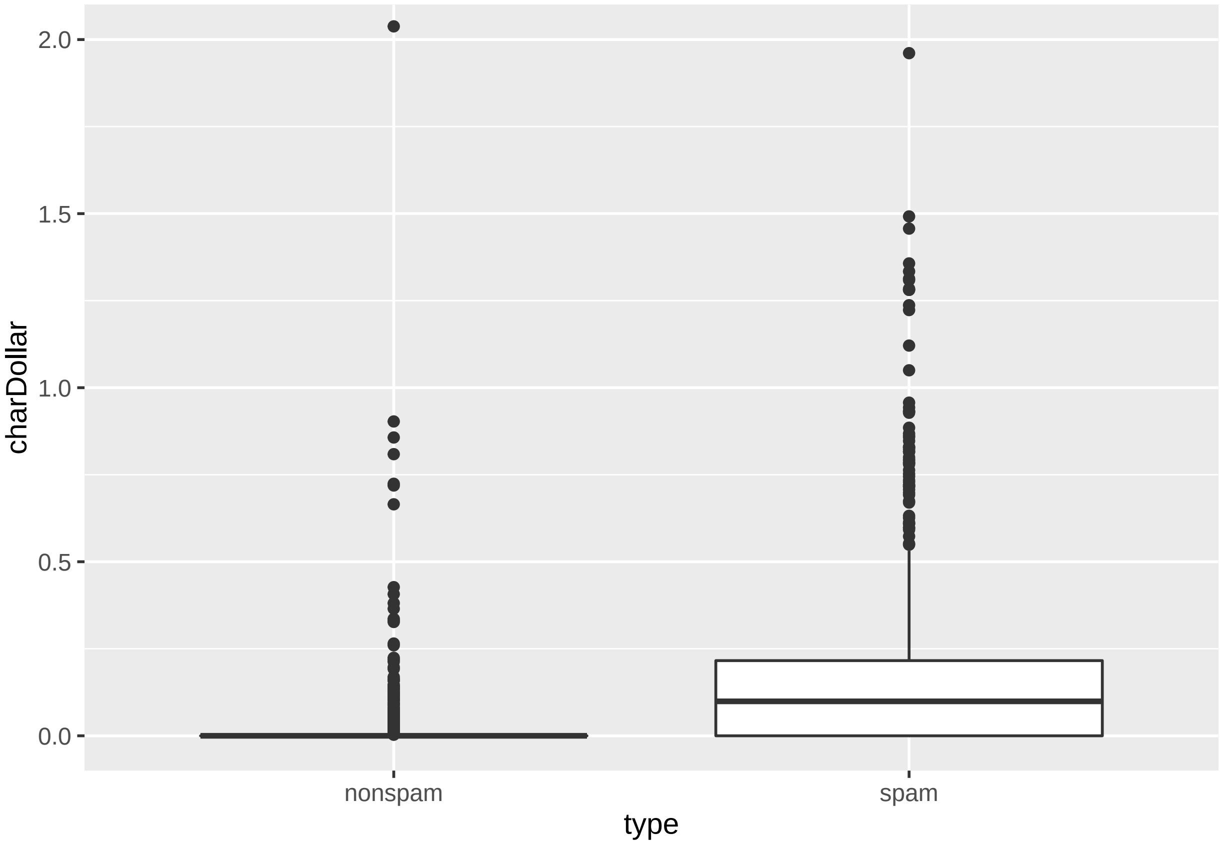
```
(61+458)/(1346+458+61+449)
```

```
## [1] 0.2242869
```

Interpret results

- The fraction of characters that are dollar signs can predict if email is spam
- An email with more dollar signs in the body is more likely to be spam, as can be seen in the following boxplot:

```
library(ggplot2)
ggplot(testSpam, aes(type, charDollar))+geom_boxplot()+coord_cartesian(ylim=c(0,2))
```



The median and interquartile range of the number of dollar signs in an email is zero for the emails that we know are not spam, but for spam emails is non-zero. This shows that the number of dollar signs in an email gives, at least, some indication of whether an email is spam or not.

- Our test set error was 22.4%. This is not great, and shows that more work is required to obtain a better indicator.