A Random Forest on the Titanic

Antonio Skilton

September 12, 2016

The Data

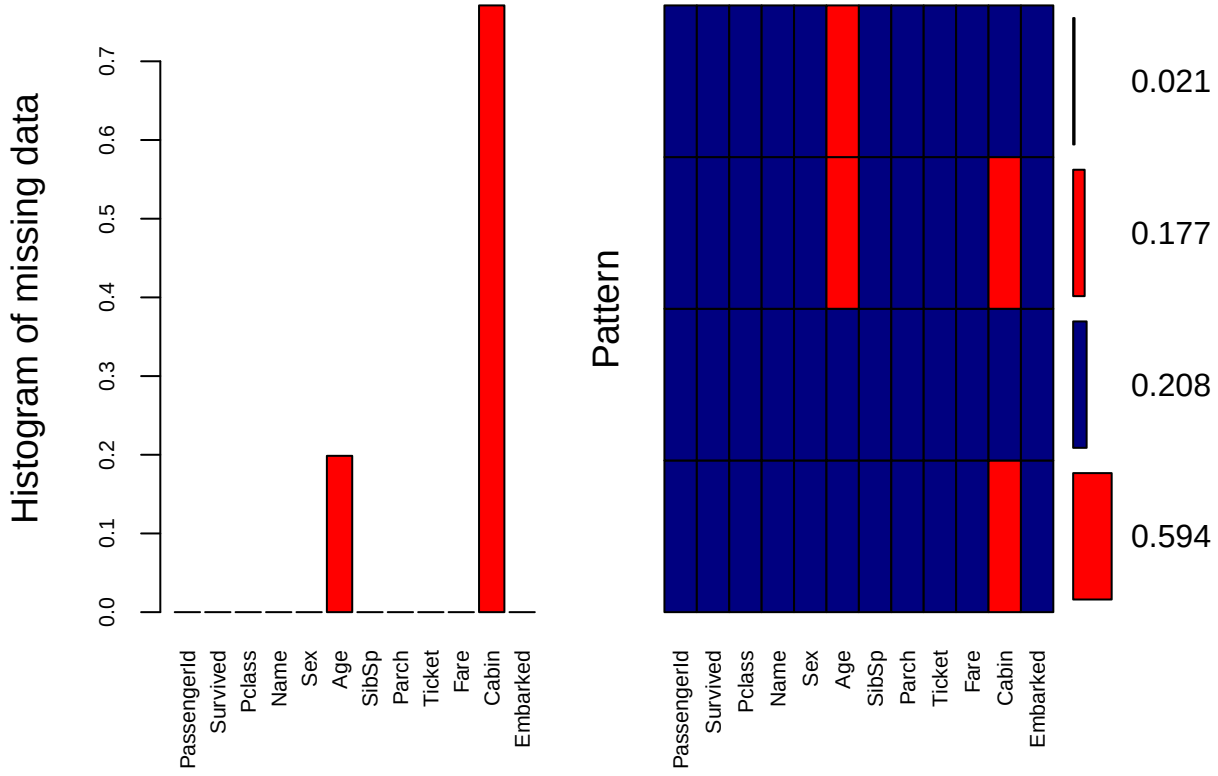
```
data <- read.csv("train.csv")
survived_label <- ifelse(data$Survived==1,"survived","died")
data$Cabin[data$Cabin==""] <- NA
source("cleanAndManipulate.R")
```

Table 1: First 5 Observations in Passenger Data

	1	2	3	4	5
PassengerId	1	2	3	4	5
Survived	0	1	1	1	0
Pclass	3	1	3	1	3
Name	Braund, Mr...	Cumings, M...	Heikkinen,...	Futrelle, ...	Allen, Mr....
Sex	male	female	female	female	male
Age	22	38	26	35	35
SibSp	1	1	0	1	0
Parch	0	0	0	0	0
Ticket	A/5 21171	PC 17599	STON/O2. 3101282	113803	373450
Fare	7.2500	71.2833	7.9250	53.1000	8.0500
Cabin	NA	C85	NA	C123	NA
Embarked	S	C	S	S	S

Missing Data

```
library(VIM)
aggr(data, col=c('navyblue','red'), numbers=TRUE,
     labels=names(data), sortVars=FALSE, cex.axis=.7,
     prop=TRUE, gap=3, ylab=c("Histogram of missing data","Pattern"))
```



Name/Title

As seen above, passenger names all have some title associated with them. I suspect that there is infmation to be gleaned from titles in the names that might not be available in either Sex or Age.

```
data$title <- NA
for(string in c("master","mr","miss","mrs")){
  data$title[grepl(string,data$Name,ignore.case=T)] <- string}
```

After creating a new column for titles, I fill in missing variables. I assume that those missing a traditional title are adults and supply them a replacement title based upon their age and sex.

```
data$title[is.na(data$title)] <- ifelse(data$Sex[is.na(data$title)]=="female","mrs","mr")
```

Table 2: The new 'title' column, made from 'Name' and 'Sex'

title	Name	Sex
mr	Braund, Mr. Owen Harris	male
mrs	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female
miss	Heikkinen, Miss. Laina	female
mrs	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female
mr	Allen, Mr. William Henry	male
mr	Moran, Mr. James	male
mr	McCarthy, Mr. Timothy J	male
master	Palsson, Master. Gosta Leonard	male
mrs	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female
mrs	Nasser, Mrs. Nicholas (Adele Achem)	female

There are clear differences in survival rates that are associated with title.

	master	miss	mr	mrs
died	17	54	451	27
survived	23	126	86	107

Other variables

Child

Anybody who is either below or equal to the age of 10, below or equal to the age of 10 *and* is accompanied by a parent, is traveling with more than one sibling, or has the name "master" is recorded as a child.

```
data$child <- ifelse(
  data$Age<=10 | (data$Age <= 15 & data$Parch>0) | data$SibSp > 1 | data$title=="master",
  1,0)
```

Missing Age

Variable is equal to 1 where passenger has no recorded age.

```
data$ageMissing <- ifelse(is.na(data$Age),1,0)
```

Cabin

Variable is equal to 1 where passenger has no recorded cabin.

```
data$cabin <- ifelse(is.na(data$Cabin),1,0)
```

Ticket number

The ticket numbers include both special characters and non-numeric characters before I clean them. I then seperate them into groups based upon the uneven distribution of the ticket numbers.

```
data$Ticket_numeric <- as.numeric(gsub("[^0-9]","",data$Ticket))
data$Ticket_group <- as.factor(ifelse(data$Ticket_numeric < 300000,"lowTicketNumber","highTicketNumber"
```

Family Size

```
data$familySize <- data$Parch + data$SibSp
```

Dummy Variables

with the following code we produce the dummy variables for the Embarked, Pclass, title, and Ticket_group columns.

```
data <- cbind(data,
              predict(
                dummyVars(~Embarked + Pclass + title + Ticket_group, data = data),
                newdata = data)
              )
```

Final Edits

Using the dplyr package I remove columns that through trial and error I have found increase the classification error rate in the random forest model.

```
library(dplyr)
data<-data %>%
    dplyr::select(-Ticket_numeric,-Sex,-title.master,
                  -title.miss,-Cabin,-Pclass,-title) %>%
    dplyr::select(Survived,child,contains("title"),
                  contains("Pclass"),cabin,contains("group"),
                  Fare,Parch,familySize,ageMissing)
```

```
Survived <- select(data, Survived) %>% unlist()  %>% as.factor()
predictors <- select(data, -Survived)
```

Random Forest Model

```
library(randomForest)
set.seed(1)
randomForest(predictors,Survived,importance=T,mtry=3,xtest=)
```

```
##
## Call:
##  randomForest(x = predictors, y = Survived, mtry = 3, importance = T)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 17.27%
## Confusion matrix:
##     0   1 class.error
## 0 412  21  0.04849885
## 1 104 187  0.35738832
```

We can now test the model on test.csv. The above cleaning and manipulation is performed by the cleanAndManipulate() function.

```
test <- read.csv("test.csv")
source("cleanAndManipulate.r")
test2 <- cleanAndManipulate(test,test=TRUE)
library(randomForest)
set.seed(1)
randomForest(predictors,Survived,importance=T,mtry=3,xtest=test2)
```

```
##
## Call:
##  randomForest(x = predictors, y = Survived, xtest = test2, mtry = 3,      importance = T)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 17.54%
## Confusion matrix:
##     0   1 class.error
## 0 410  23  0.05311778
## 1 104 187  0.35738832
```