titanic.R

dan_9

Thu Jul 21 18:24:49 2016

```
# VARIABLE DESCRIPTIONS:
#
# survival        Survival
# (0 = No; 1 = Yes)
# pclass          Passenger Class
# (1 = 1st; 2 = 2nd; 3 = 3rd)
# name            Name
# sex             Sex
# age             Age
# sibsp           Number of Siblings/Spouses Aboard
# parch           Number of Parents/Children Aboard
# ticket          Ticket Number
# fare            Passenger Fare
# cabin           Cabin
# embarked        Port of Embarkation
# (C = Cherbourg; Q = Queenstown; S = Southampton)
#
# SPECIAL NOTES:
#   Pclass is a proxy for socio-economic status (SES)
# 1st ~ Upper; 2nd ~ Middle; 3rd ~ Lower
#
# Age is in Years; Fractional if Age less than One (1)
# If the Age is Estimated, it is in the form xx.5
#
# With respect to the family relation variables (i.e. sibsp and parch)
# some relations were ignored.  The following are the definitions used
# for sibsp and parch.
#
# Sibling:  Brother, Sister, Stepbrother, or Stepsister of Passenger Aboard Titanic
# Spouse:   Husband or Wife of Passenger Aboard Titanic (Mistresses and Fiances Ignored)
# Parent:   Mother or Father of Passenger Aboard Titanic
# Child:    Son, Daughter, Stepson, or Stepdaughter of Passenger Aboard Titanic
#
# Other family relatives excluded from this study include cousins,
# nephews/nieces, aunts/uncles, and in-laws.  Some children travelled
# only with a nanny, therefore parch=0 for them.  As well, some
# travelled with very close friends or neighbors in a village, however,
# the definitions do not support such relations.

rm(list = ls())
setwd("C:/Users/dan_9/Desktop/COURSERA + SELF STUDY/Kaggle/Titanic")

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(gridExtra)
library(rpart) # grow classification tree
library(randomForest)
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(stringr)
library(missForest) # imputation
```

```
## Loading required package: foreach
```

```
## Loading required package: itertools
```

```
## Loading required package: iterators
```

```
##
## Attaching package: 'itertools'
```

```
## The following object is masked from 'package:dplyr':
##
##     chain
```

```
library(mice) # imputation
```

```
## Loading required package: Rcpp
```

```
## mice 2.25 2015-11-09
```

```
library(VIM) # visualization missing value
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## Loading required package: data.table
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, last
```

```
## VIM is ready to use.
##  Since version 4.0.0 the GUI is in its own package VIMGUI.
##
##           Please use the package to use the new (and old) GUI.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/alexkowa/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
genderclassmodel = read.csv("genderclassmodel.csv")
gendermodel = read.csv("gendermodel.csv")
training = read.csv("train.csv")
testing = read.csv("test.csv")

glimpse(training)
```

```
## Observations: 891
## Variables: 12
## $ PassengerId (int) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,...
## $ Survived    (int) 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0,...
## $ Pclass      (int) 3, 1, 3, 1, 3, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3,...
## $ Name        (fctr) Braund, Mr. Owen Harris, Cumings, Mrs. John Bradl...
## $ Sex         (fctr) male, female, female, female, male, male, male, m...
## $ Age         (dbl) 22, 38, 26, 35, 35, NA, 54, 2, 27, 14, 4, 58, 20, ...
## $ SibSp       (int) 1, 1, 0, 1, 0, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4,...
## $ Parch       (int) 0, 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1,...
## $ Ticket      (fctr) A/5 21171, PC 17599, STON/O2. 3101282, 113803, 37...
## $ Fare        (dbl) 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 8.4583, ...
## $ Cabin       (fctr) , C85, , C123, , , E46, , , , G6, C103, , , , , ,...
## $ Embarked    (fctr) S, C, S, S, S, Q, S, S, S, C, S, S, S, S, S, S, Q...
```

```
head(training)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Sex Age SibSp
## 1                             Braund, Mr. Owen Harris   male  22     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
## 3                              Heikkinen, Miss. Laina female  26     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
## 5                            Allen, Mr. William Henry   male  35     0
## 6                                    Moran, Mr. James   male  NA     0
##   Parch           Ticket    Fare Cabin Embarked
## 1     0        A/5 21171  7.2500              S
## 2     0         PC 17599 71.2833   C85        C
## 3     0 STON/O2. 3101282  7.9250              S
## 4     0           113803 53.1000  C123        S
## 5     0           373450  8.0500              S
## 6     0           330877  8.4583              Q
```

```
tail(training)
```

```
##     PassengerId Survived Pclass                                     Name
## 886         886        0      3     Rice, Mrs. William (Margaret Norton)
## 887         887        0      2                    Montvila, Rev. Juozas
## 888         888        1      1             Graham, Miss. Margaret Edith
## 889         889        0      3 Johnston, Miss. Catherine Helen "Carrie"
## 890         890        1      1                     Behr, Mr. Karl Howell
## 891         891        0      3                       Dooley, Mr. Patrick
##        Sex Age SibSp Parch     Ticket   Fare Cabin Embarked
## 886 female  39     0     5     382652 29.125              Q
## 887   male  27     0     0     211536 13.000              S
## 888 female  19     0     0     112053 30.000   B42        S
## 889 female  NA     1     2 W./C. 6607 23.450              S
## 890   male  26     0     0     111369 30.000  C148        C
## 891   male  32     0     0     370376  7.750              Q
```

```
glimpse(testing)
```

```
## Observations: 418
## Variables: 11
## $ PassengerId (int) 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, ...
## $ Pclass      (int) 3, 3, 2, 3, 3, 3, 3, 2, 3, 3, 3, 1, 1, 2, 1, 2, 2,...
## $ Name        (fctr) Kelly, Mr. James, Wilkes, Mrs. James (Ellen Needs...
## $ Sex         (fctr) male, female, male, male, female, male, female, m...
## $ Age         (dbl) 34.5, 47.0, 62.0, 27.0, 22.0, 14.0, 30.0, 26.0, 18...
## $ SibSp       (int) 0, 1, 0, 0, 1, 0, 0, 1, 0, 2, 0, 0, 1, 1, 1, 1, 0,...
## $ Parch       (int) 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Ticket      (fctr) 330911, 363272, 240276, 315154, 3101298, 7538, 33...
## $ Fare        (dbl) 7.8292, 7.0000, 9.6875, 8.6625, 12.2875, 9.2250, 7...
## $ Cabin       (fctr) , , , , , , , , , , , , B45, , E31, , , , , , , ,...
## $ Embarked    (fctr) Q, S, Q, S, S, S, Q, S, C, S, S, S, S, S, S, C, Q...
```

```
head(testing)
```

```
##   PassengerId Pclass                                          Name    Sex
## 1         892      3                              Kelly, Mr. James   male
## 2         893      3              Wilkes, Mrs. James (Ellen Needs) female
## 3         894      2                     Myles, Mr. Thomas Francis   male
## 4         895      3                              Wirz, Mr. Albert   male
## 5         896      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist) female
## 6         897      3                    Svensson, Mr. Johan Cervin   male
##    Age SibSp Parch  Ticket    Fare Cabin Embarked
## 1 34.5     0     0  330911  7.8292              Q
## 2 47.0     1     0  363272  7.0000              S
## 3 62.0     0     0  240276  9.6875              Q
## 4 27.0     0     0  315154  8.6625              S
## 5 22.0     1     1 3101298 12.2875              S
## 6 14.0     0     0    7538  9.2250              S
```

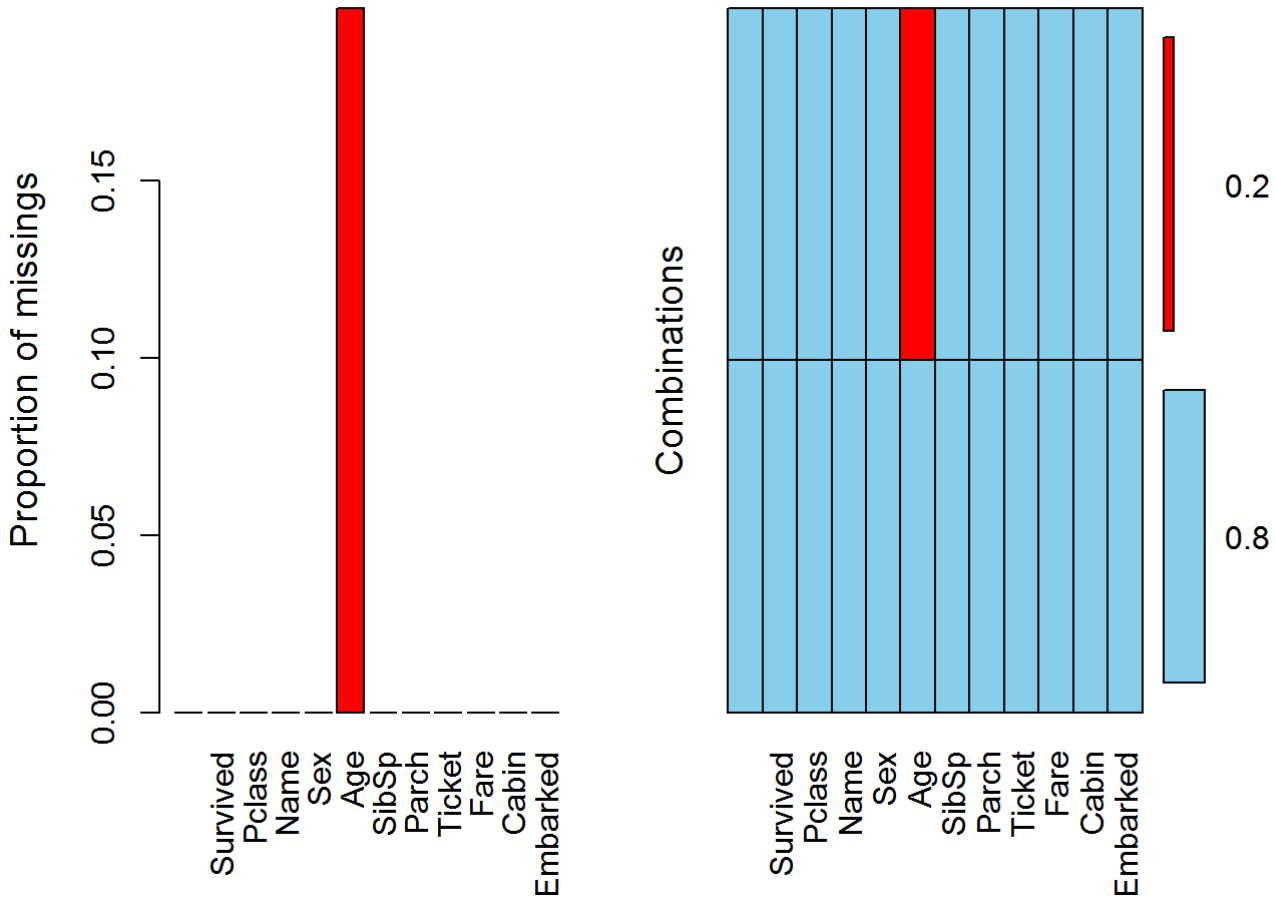
```
tail(testing)
```

```
##     PassengerId Pclass                           Name    Sex  Age SibSp
## 413        1304      3 Henriksson, Miss. Jenny Lovisa female 28.0     0
## 414        1305      3             Spector, Mr. Woolf   male   NA     0
## 415        1306      1   Oliva y Ocana, Dona. Fermina female 39.0     0
## 416        1307      3   Saether, Mr. Simon Sivertsen   male 38.5     0
## 417        1308      3            Ware, Mr. Frederick   male   NA     0
## 418        1309      3       Peter, Master. Michael J   male   NA     1
##     Parch             Ticket     Fare Cabin Embarked
## 413     0             347086   7.7750              S
## 414     0          A.5. 3236   8.0500              S
## 415     0           PC 17758 108.9000  C105        C
## 416     0 SOTON/O.Q. 3101262   7.2500              S
## 417     0             359309   8.0500              S
## 418     1               2668  22.3583              C
```

```
# adding column "Survived" on the testing dataset
testing$Survived = NA
# move this column to the 2nd for binding purpose
testing = testing[,c(1, grep("Survived", colnames(testing)), 2 :  (grep("Survived", colnames(tes
ting))-1))]

full = rbind(training, testing)
glimpse(full)
```

```
## Observations: 1,309
## Variables: 12
## $ PassengerId (int) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,...
## $ Survived    (int) 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0,...
## $ Pclass      (int) 3, 1, 3, 1, 3, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3,...
## $ Name        (fctr) Braund, Mr. Owen Harris, Cumings, Mrs. John Bradl...
## $ Sex         (fctr) male, female, female, female, male, male, male, m...
## $ Age         (dbl) 22, 38, 26, 35, 35, NA, 54, 2, 27, 14, 4, 58, 20, ...
## $ SibSp       (int) 1, 1, 0, 1, 0, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4,...
## $ Parch       (int) 0, 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1,...
## $ Ticket      (fctr) A/5 21171, PC 17599, STON/O2. 3101282, 113803, 37...
## $ Fare        (dbl) 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 8.4583, ...
## $ Cabin       (fctr) , C85, , C123, , , E46, , , , G6, C103, , , , , ,...
## $ Embarked    (fctr) S, C, S, S, S, Q, S, S, S, C, S, S, S, S, S, S, Q...
```

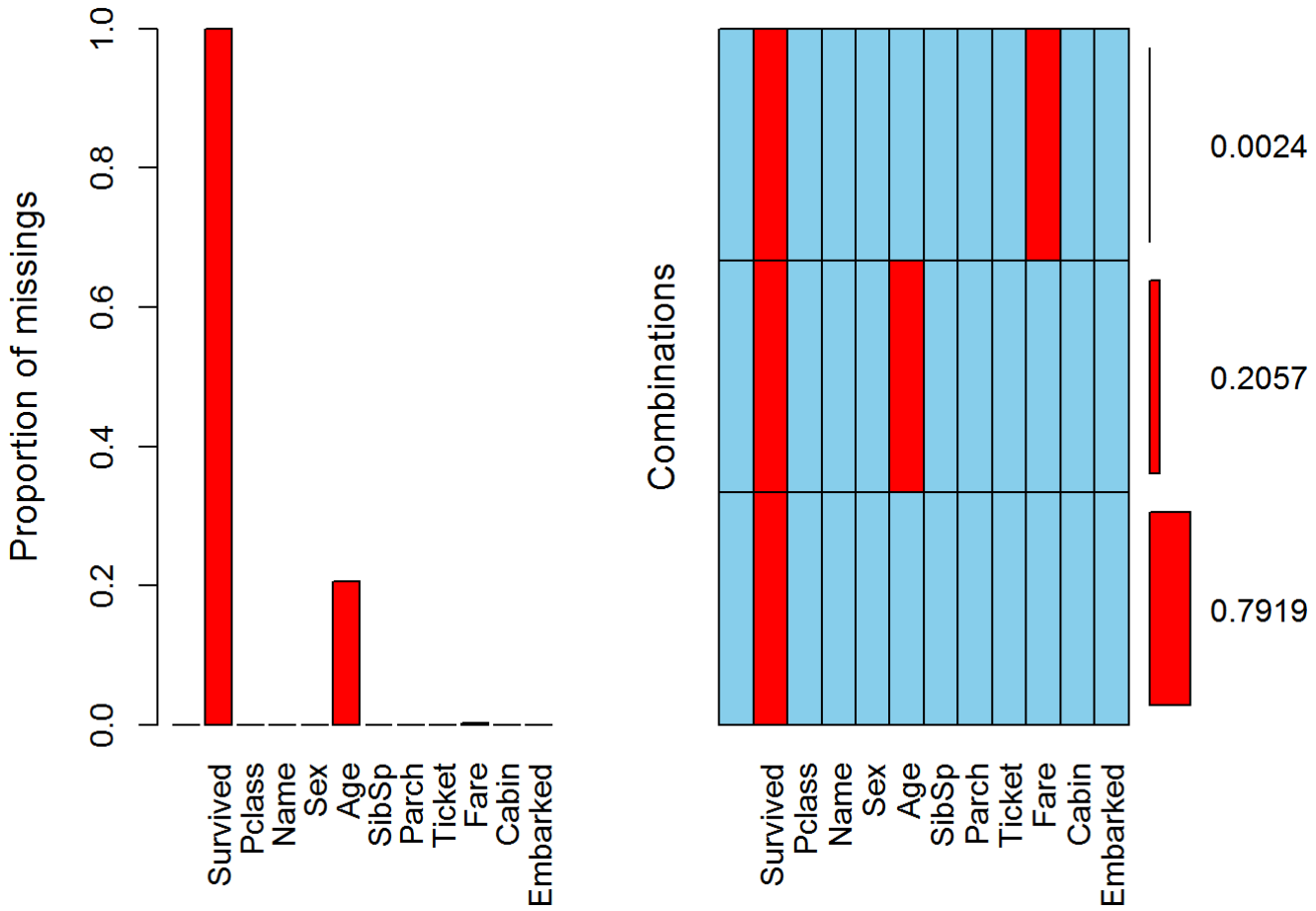
```
#############################################################################
# PART 1 - DATA CLEANING, MANIPULATION, AND EXPLORATORY ANALYSIS
#############################################################################

# column name should be character instead of factor
full$Name = as.character(full$Name)

# checking how many NA for each column for both training and testing data
trainingNAs = aggr(training, numbers=T, sortVars=F)
```




```
trainingNAs
```

```
##
##  Missings in variables:
##  Variable Count
##       Age   177
```

```
testingNAs = aggr(testing, numbers=T, sortVars=F)
```

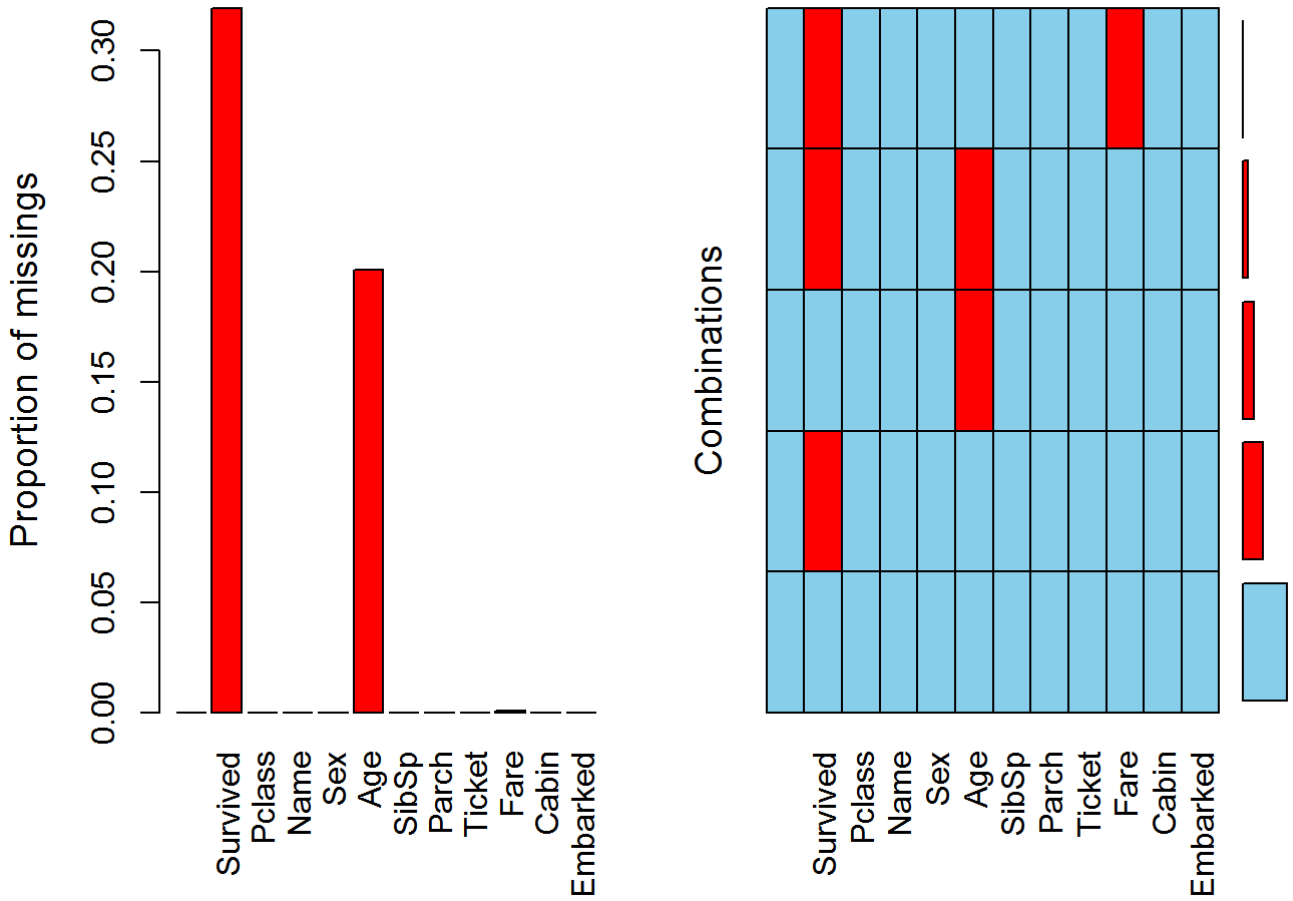



```
testingNAs
```

```
##
##  Missings in variables:
##  Variable Count
##  Survived   418
##       Age    86
##      Fare     1
```

```
fullNAs = aggr(full, numbers=T, sortVars=F)
```

```
## Warning in plot.aggr(res, ...): not enough horizontal space to display
## frequencies
```




```
fullNAs
```

```
##
##  Missings in variables:
##  Variable Count
##  Survived   418
##       Age   263
##      Fare     1
```

```
# We need to see Fare column and fill it with the value that make sense,
# in this case, we will fill the cell with based on the passanger's other attributes, compared w
ith other passanger,
# which is Pclass, Age >=50, Embarked from Southampton

which(is.na(full$Fare))
```

```
## [1] 1044
```

```
# [1] 1044

full[1044,]
```

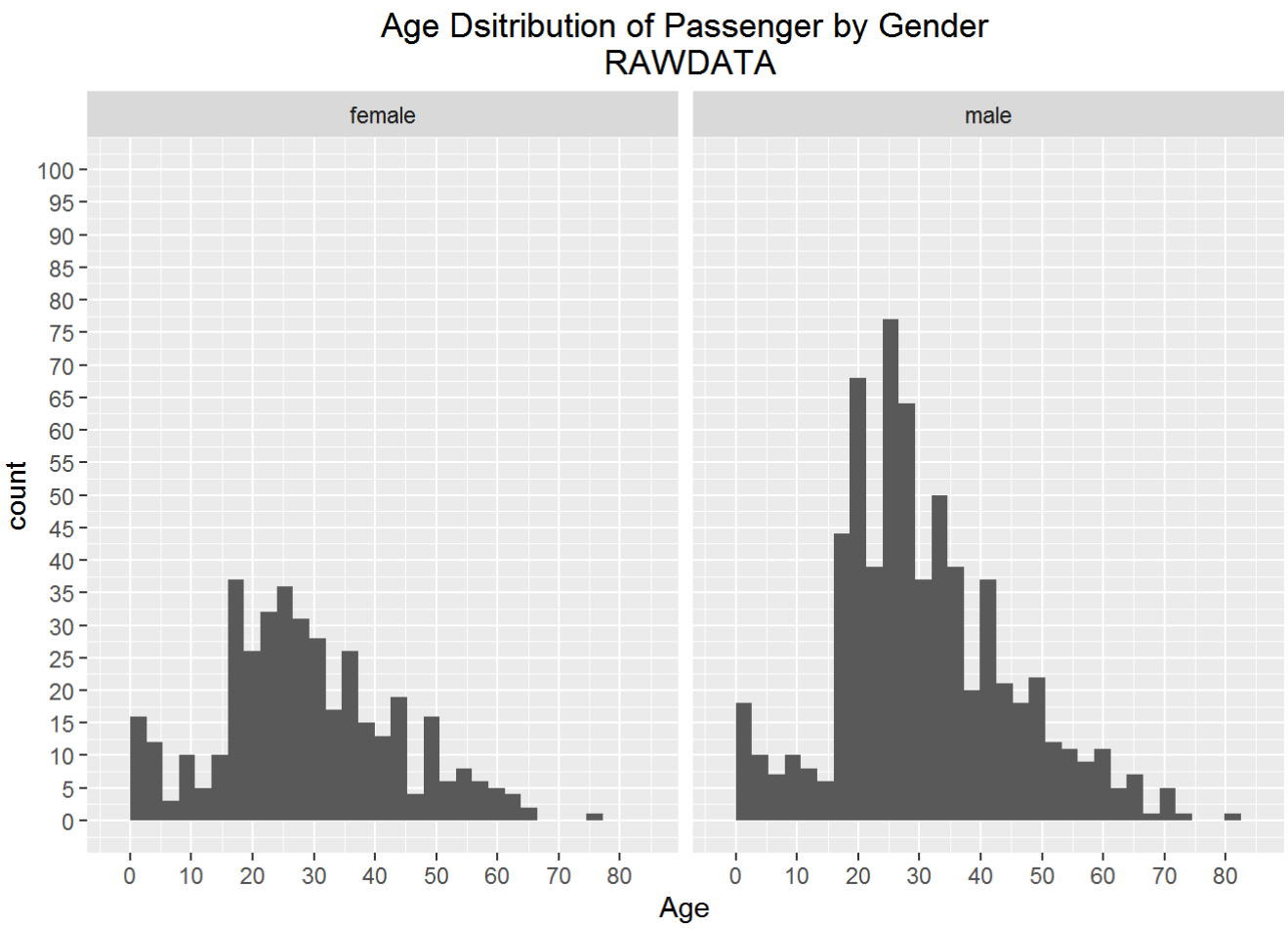
```
##      PassengerId Survived Pclass               Name  Sex  Age SibSp Parch
## 1044        1044       NA      3 Storey, Mr. Thomas male 60.5     0     0
##      Ticket Fare Cabin Embarked
## 1044   3701   NA              S
```

```
# PassengerId Survived Pclass               Name  Sex  Age SibSp Parch Ticket Fare Cabin    Emba
rked honorific GenderAgeClass FareClass Surname
# 1044        1044       NA      3 Storey, Mr. Thomas male 60.5     0     0   3701   NA       S
        Mr.     Adult Male      <NA>  Storey

# to estimate the fare of passenger ID   1044, we need to find the mean value of people similar
 to his pattern:
full %>% filter(!is.na(Fare), Pclass == 3, Age >= 50, Embarked == "S" ) %>%
summarise(mean(Fare))
```

```
##   mean(Fare)
## 1    8.43042
```

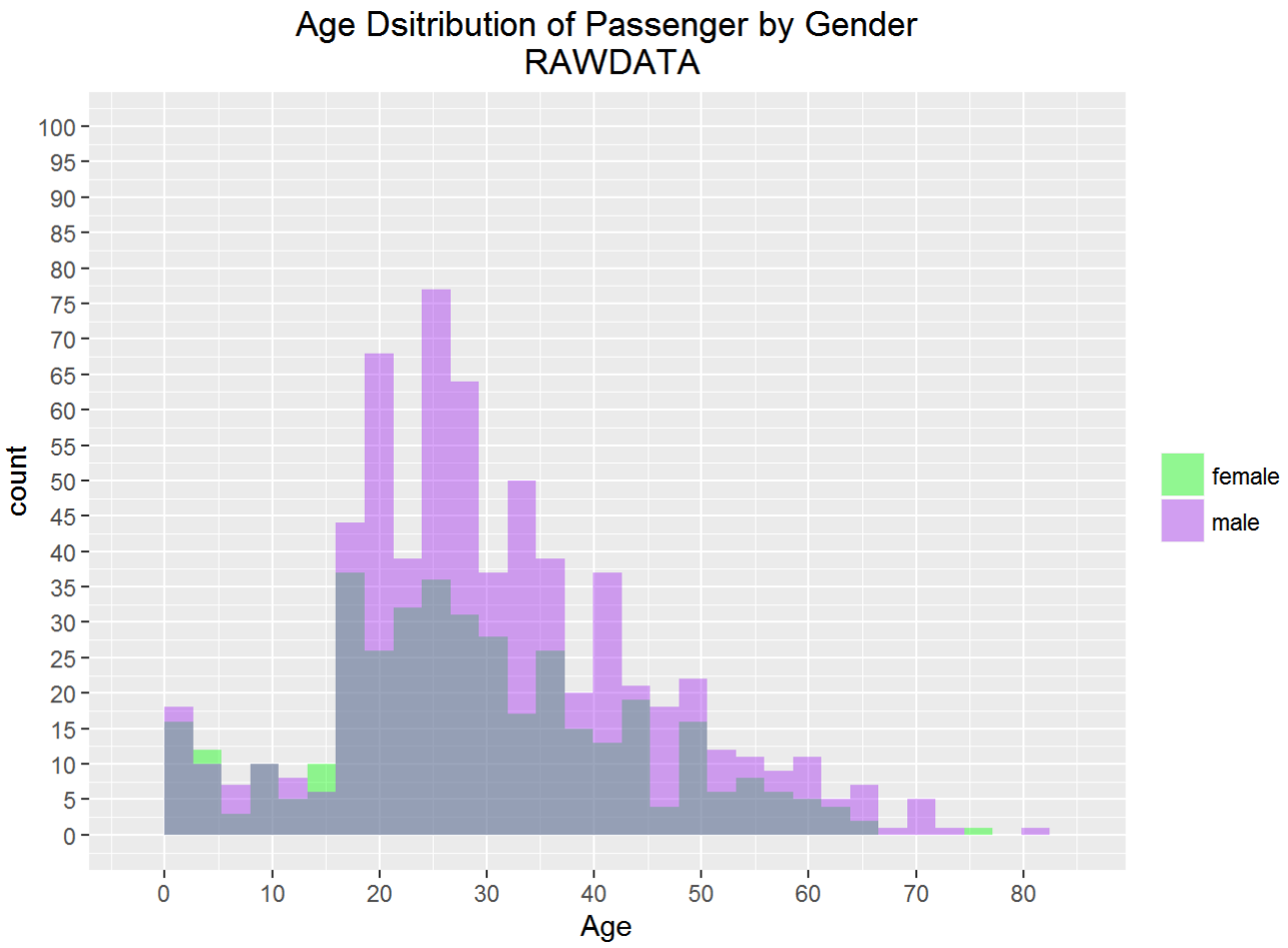
```
# 8.43042 <- we need to fill the NA of passanger #1044's fare with this value

full$Fare[1044] = 8.43042

# Since column Age contains the most of NAs, 177 NAs, we better to do exploratory analysis with
 these raw data, before we perform
# imputation of missing values. Because our imputation may alter our view towards raw dataset if
 we perform the exploratory later

# In these exploratory analysis, we will exclude the observations with NAs in Age

plotAgeRaw1 = ggplot(full[!is.na(full$Age),], aes(x = Age )) +
  geom_histogram() +
  facet_grid(~Sex) +
  coord_cartesian(ylim =  seq(0,100,5))+
  scale_y_continuous(breaks = seq(0,100,5)) +
  scale_x_continuous(breaks = seq(0,90,10)) +
  ggtitle("Age Dsitribution of Passenger by Gender\n RAWDATA")

plotAgeRaw2 = ggplot(full[!is.na(full$Age),], aes(x = Age, fill = as.factor(Sex))) +
  geom_histogram(position = "identity", alpha = .4) +
  scale_fill_manual(values=c("green", "purple")) +
  coord_cartesian(ylim =  seq(0,100,5))+
  scale_y_continuous(breaks = seq(0,100,5)) +
  scale_x_continuous(breaks = seq(0,90,10)) +
  ggtitle("Age Dsitribution of Passenger by Gender\n RAWDATA") +
  labs(fill="")

medians = aggregate(Age ~  Sex, full, median)

plotAgeRaw3 = ggplot(full[!is.na(full$Age),], aes(x = Sex, y = Age)) +
  geom_boxplot() +
  scale_y_continuous(breaks = seq(0,80,5)) +
  geom_text( data = medians, aes( label = Age, y = Age), vjust = -.5) +
  ggtitle("Age Dsitribution of Passenger by Gender\n RAW")

plotAgeRaw1
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

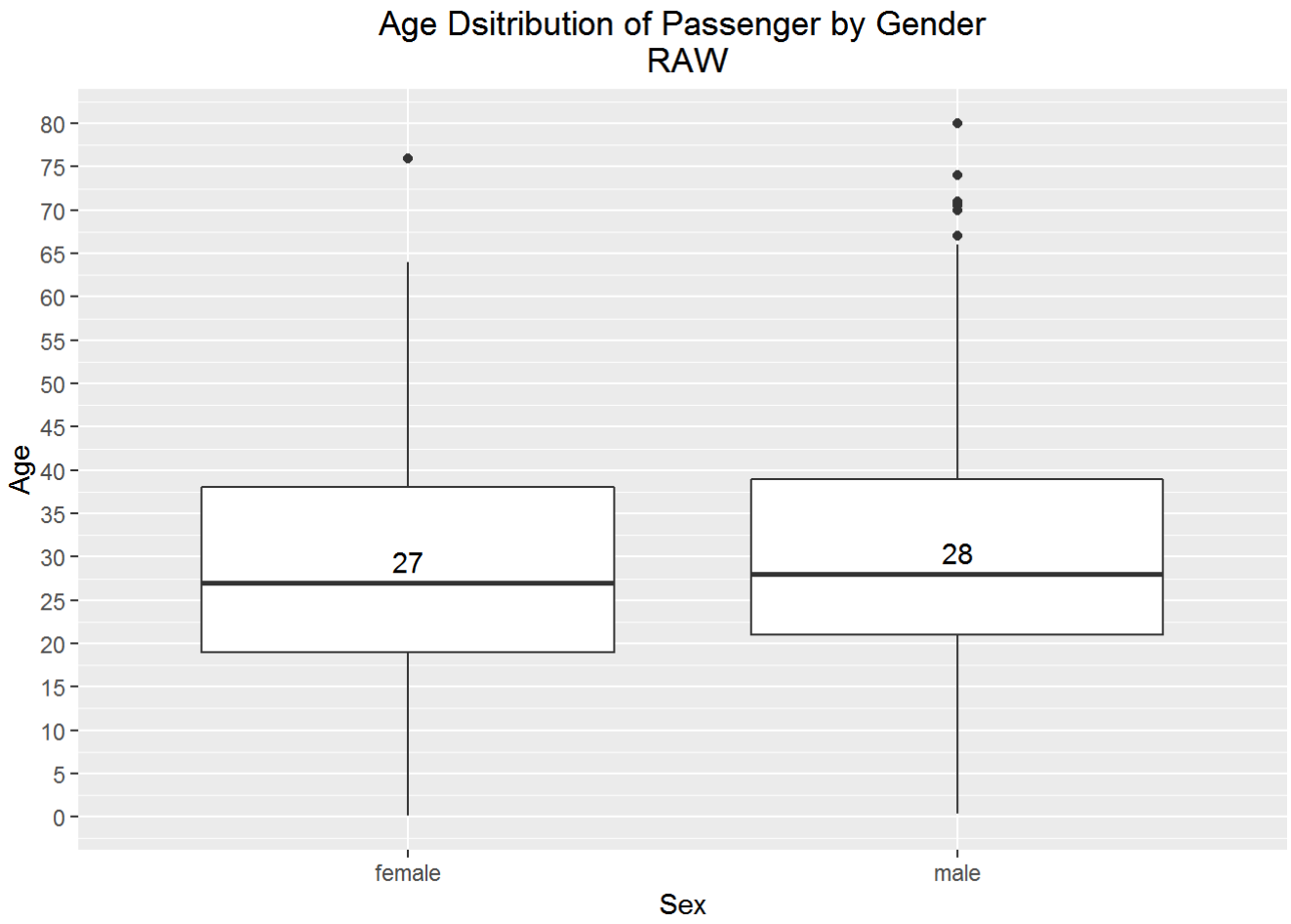



```
plotAgeRaw2
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
plotAgeRaw3
```




```
# we would like to analyze based on honorifics system on their name,
# and ultimately group it into adult male, female, young male

full$honorific = str_extract(full$Name, pattern = "\\,[ ][A-z]*[ ]*[A-z]*\\.")
full$honorific = gsub(", ","",full$honorific)

full$honorific = as.factor(full$honorific)
table(full$honorific)
```

```
##
##        Capt.         Col.         Don.        Dona.          Dr.
##            1            4            1            1            8
##    Jonkheer.        Lady.       Major.      Master.        Miss.
##            1            1            2           61          260
##        Mlle.        Mme.          Mr.         Mrs.          Ms.
##            2            1          757          197            2
##         Rev.         Sir. the Countess.
##            8            1            1
```

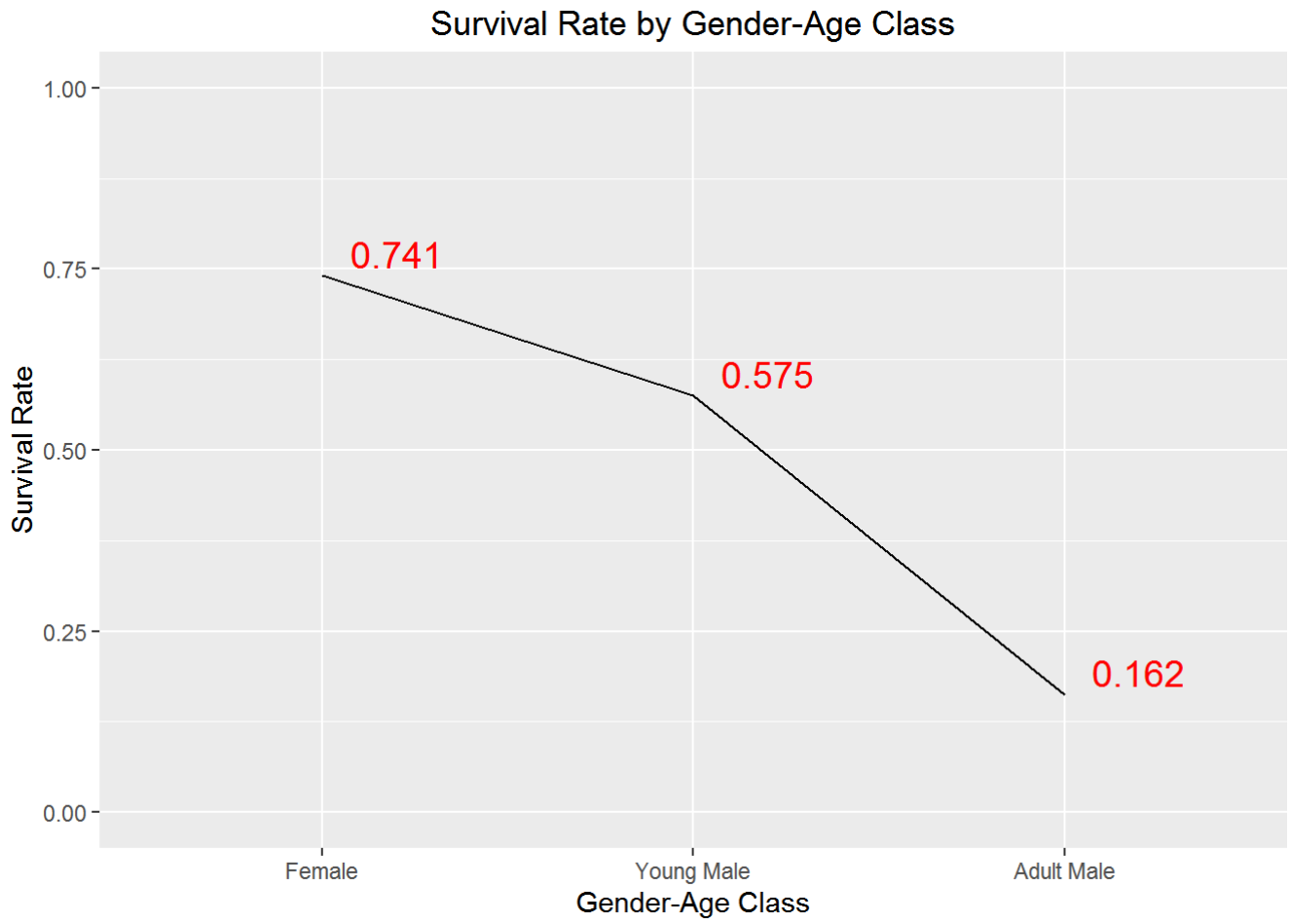
```
# then we need to find the age range of each honorific

SurvivalRateByHonorific = full %>%
  filter(Survived != "NA") %>%
  group_by(honorific) %>%
  summarise(minAge = min(Age, na.rm = T),maxAge = max(Age, na.rm = T), PassengerCount = n(),
            PassengerSurvived =  sum(Survived == 1), survivalRate = round(sum(Survived ==
1)/n(),3)) %>%
  arrange(desc(PassengerSurvived))
SurvivalRateByHonorific
```

```
## Source: local data frame [17 x 6]
##
##        honorific minAge maxAge PassengerCount PassengerSurvived
##           (fctr)  (dbl)  (dbl)          (int)              (int)
## 1          Miss.   0.75     63            182                127
## 2           Mrs.  14.00     63            125                 99
## 3            Mr.  11.00     80            517                 81
## 4        Master.   0.42     12             40                 23
## 5            Dr.  23.00     54              7                  3
## 6          Mlle.  24.00     24              2                  2
## 7           Col.  56.00     60              2                  1
## 8          Lady.  48.00     48              1                  1
## 9         Major.  45.00     52              2                  1
## 10          Mme.  24.00     24              1                  1
## 11           Ms.  28.00     28              1                  1
## 12          Sir.  49.00     49              1                  1
## 13 the Countess.  33.00     33              1                  1
## 14         Capt.  70.00     70              1                  0
## 15          Don.  40.00     40              1                  0
## 16     Jonkheer.  38.00     38              1                  0
## 17          Rev.  27.00     57              6                  0
## Variables not shown: survivalRate (dbl)
```

```
# We can clearly see there are 18 honorific name being used in this dataset,
# However we can classified all 18 into three different group:
# Female, young male, adult male
# The reason is because Miss and Mrs does not really tell the age of the a female.
# Furthermore, it is very clear that female has higher survival rate

full$GenderAgeClass[full$honorific ==  "Miss." | full$honorific ==  "Mrs."| full$honorific ==
"Mlle."|
                      full$honorific ==  "Lady."|full$honorific ==  "Mme."|full$honorific ==  "M
s."|
                      full$honorific ==  "the Countess."|full$honorific ==  "Dona."] = "Female"

full$GenderAgeClass[full$honorific ==  "Mr." |  full$honorific ==  "Dr."| full$honorific ==  "Co
l."|
                      full$honorific ==  "Major."| full$honorific ==  "Sir."|full$honorific ==
"Capt."|
                      full$honorific ==  "Don."| full$honorific ==  "Jonkheer."|full$honorific =
=  "Rev."] = "Adult Male"

full$GenderAgeClass[full$honorific ==  "Master."] = "Young Male"

full$GenderAgeClass = as.factor(full$GenderAgeClass)

SurvivalRateByGenderAge = full %>%
  filter(Survived != "NA") %>%
  group_by(GenderAgeClass) %>%
  summarise(minAge = min(Age, na.rm = T),maxAge = max(Age, na.rm = T), PassengerCount = n(),
            PassengerSurvived =  sum(Survived == 1), survivalRate = round(sum(Survived ==
1)/n(),3)) %>%
  arrange(desc(PassengerSurvived))
SurvivalRateByGenderAge
```

```
## Source: local data frame [3 x 6]
##
##   GenderAgeClass minAge maxAge PassengerCount PassengerSurvived
##           (fctr)  (dbl)  (dbl)          (int)             (int)
## 1         Female   0.75     63            313               232
## 2     Adult Male  11.00     80            538                87
## 3     Young Male   0.42     12             40                23
## Variables not shown: survivalRate (dbl)
```

```
ggplot(SurvivalRateByGenderAge, aes(x = reorder(GenderAgeClass, -survivalRate), y =
survivalRate, group = 1)) +
  geom_line()+
  geom_text(aes(label = survivalRate), vjust = -.3, hjust = -.3 , size = 5, color = "red") +
  ggtitle("Survival Rate by Gender-Age Class")+
  ylim(0,1)+
  xlab("Gender-Age Class") +
  ylab("Survival Rate")
```

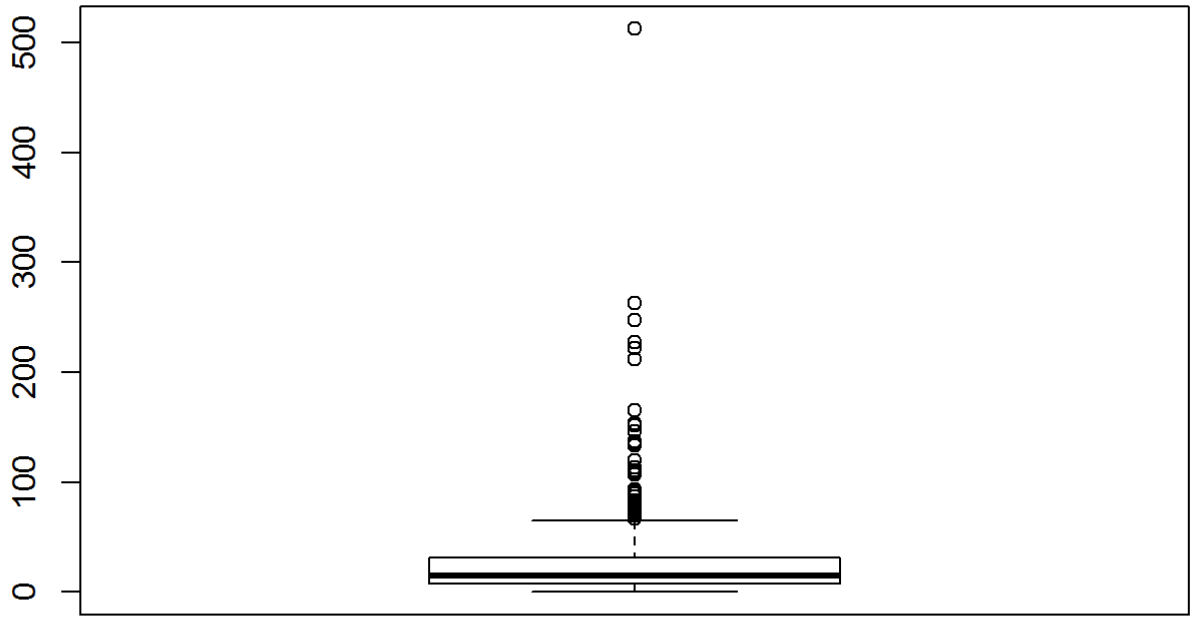



```
ggplot(full[!is.na(full$Survived),], aes(x = Survived, y = GenderAgeClass)) +
  ggtitle("Gender-Age Class survival cases")+
  scale_x_continuous( breaks = seq(0,1,1))+
  xlab("0 = not-Survived \t 1 = Survived ") +
  ylab("Gender Age Class")+
  geom_jitter(width = .65, color = "blue")
```


```
# Intuitively, it seems like there are strong correlation between variable "pclass" and "fare"
# since pclass  = 1 means the highest and tend to have higher fare, then we expect strong negati
ve correlation between there two variables
cor(full$Pclass[!is.na(full$Survived)], full$Fare[!is.na(full$Survived)])
```

```
## [1] -0.5494996
```

```
# turned out its not as high as we expected, so that means it is necessary to keep both variable
s

boxplot(full$Fare, na.rm = T)
```



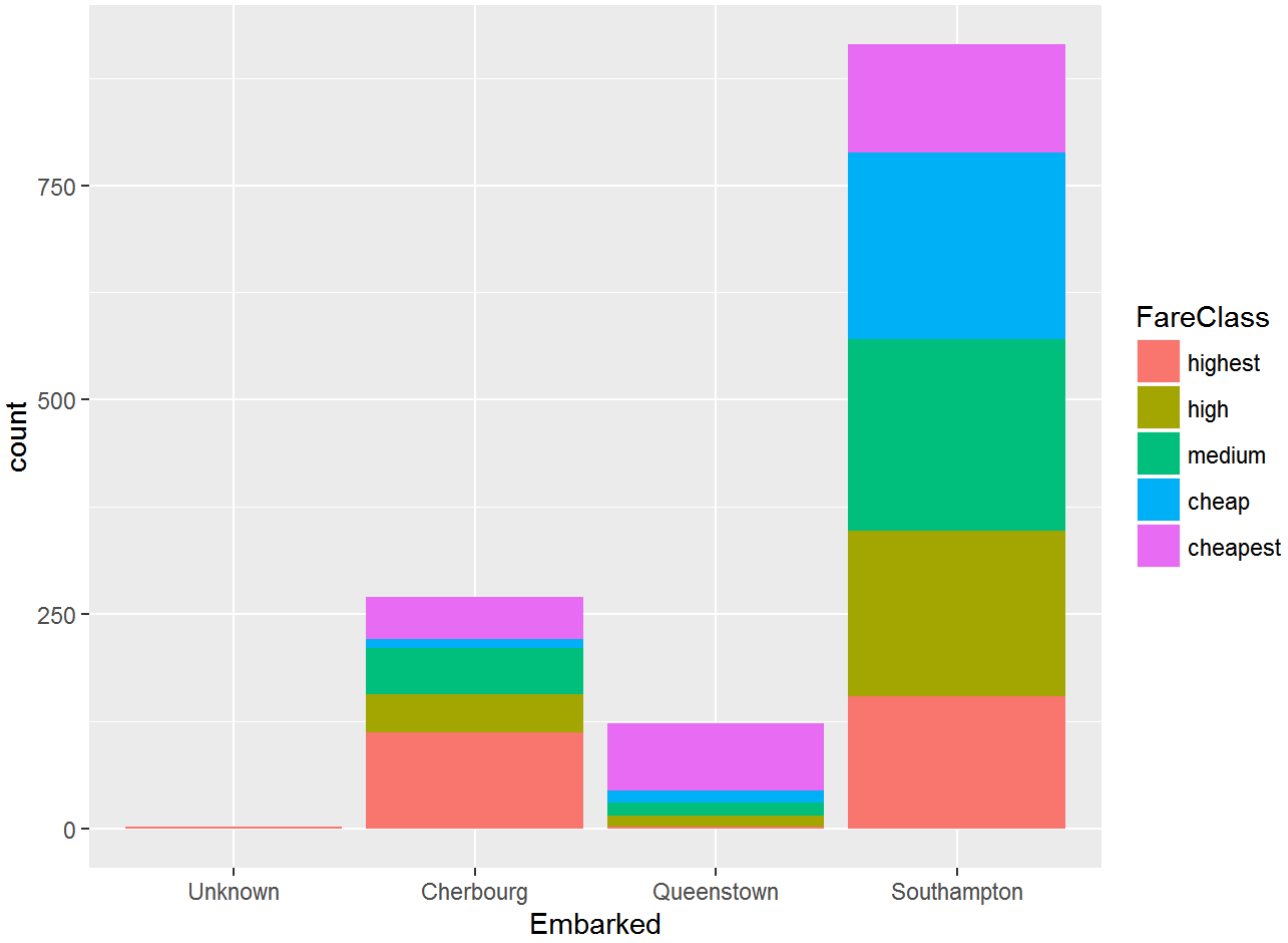

```
# for quantile, we used only the training data
Q = quantile(full$Fare[!is.na(full$Survived)], probs = seq(0,1,.2) )

FareClass = matrix(ncol = 1, nrow = nrow(full))

for( i in 1 : length(Q)){
  FareClass[full$Fare >= Q[i] & full$Fare <= Q[i+1]] = i
}

table(FareClass)
```

```
## FareClass
##   1   2   3   4   5
## 254 241 293 250 271
```

```
FareClass = as.factor(FareClass)
levels(FareClass) = c("cheapest","cheap","medium","high","highest")
FareClass =  factor(FareClass, levels = rev(levels(FareClass)))
full$FareClass = FareClass

levels(full$Embarked) = c("Unknown", "Cherbourg", "Queenstown", "Southampton")
table(full$Embarked)
```

```
##
##     Unknown    Cherbourg  Queenstown Southampton
##           2          270         123         914
```

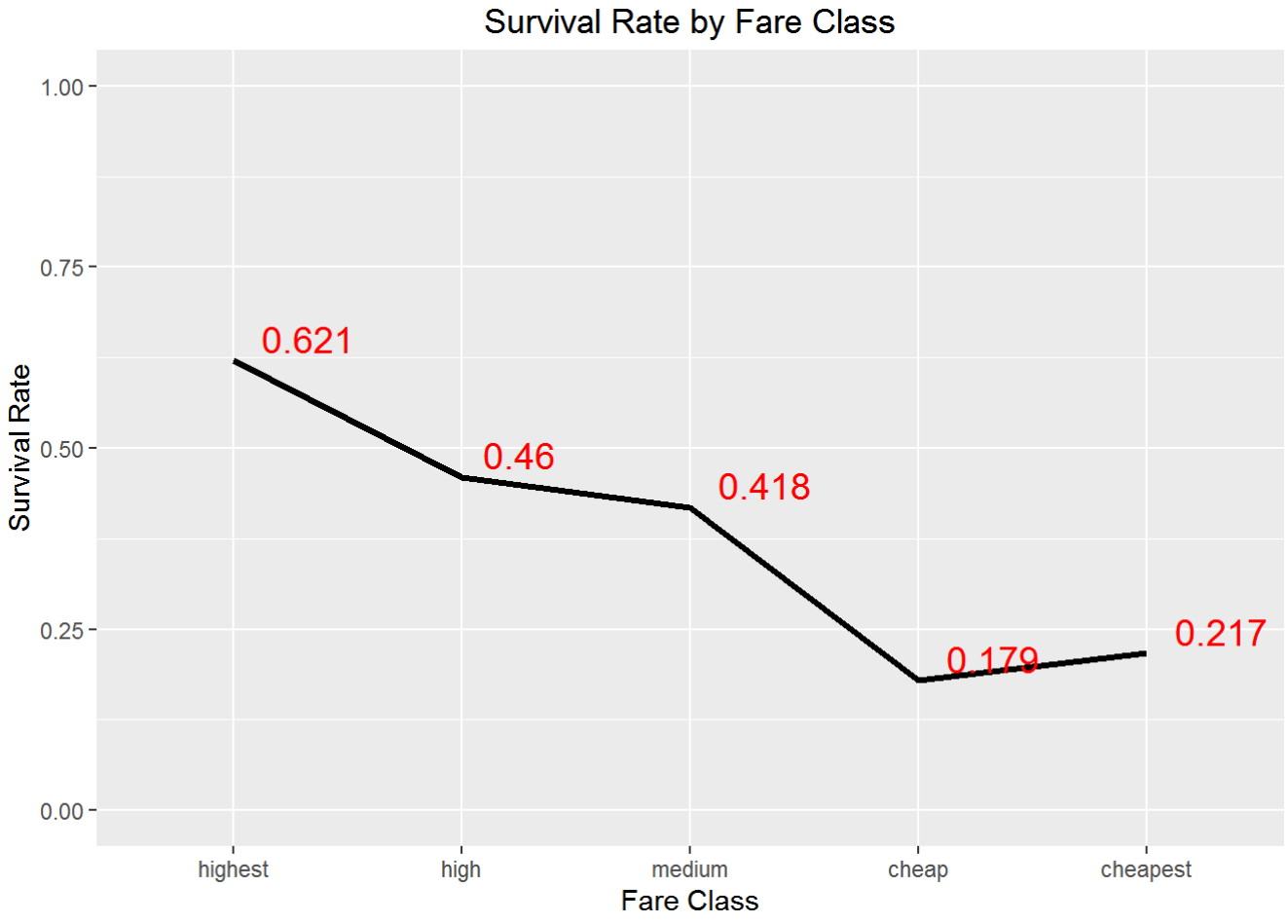
```
ggplot(data = full, aes(x = Embarked, fill = FareClass)) +
  geom_bar()
```




```
SurvivalRateByFareClass = full %>%
  filter(Survived != "NA") %>%
  group_by(FareClass) %>%
  summarise(PassengerCount = n(),PassengerSurvived =  sum(Survived == 1),
            survivalRate = round(sum(Survived == 1)/n(),3)) %>%
  arrange(desc(FareClass))
SurvivalRateByFareClass
```

```
## Source: local data frame [5 x 4]
##
##   FareClass PassengerCount PassengerSurvived survivalRate
##      (fctr)          (int)             (int)        (dbl)
## 1  cheapest            166                36        0.217
## 2     cheap            173                31        0.179
## 3    medium            196                82        0.418
## 4      high            174                80        0.460
## 5   highest            182               113        0.621
```

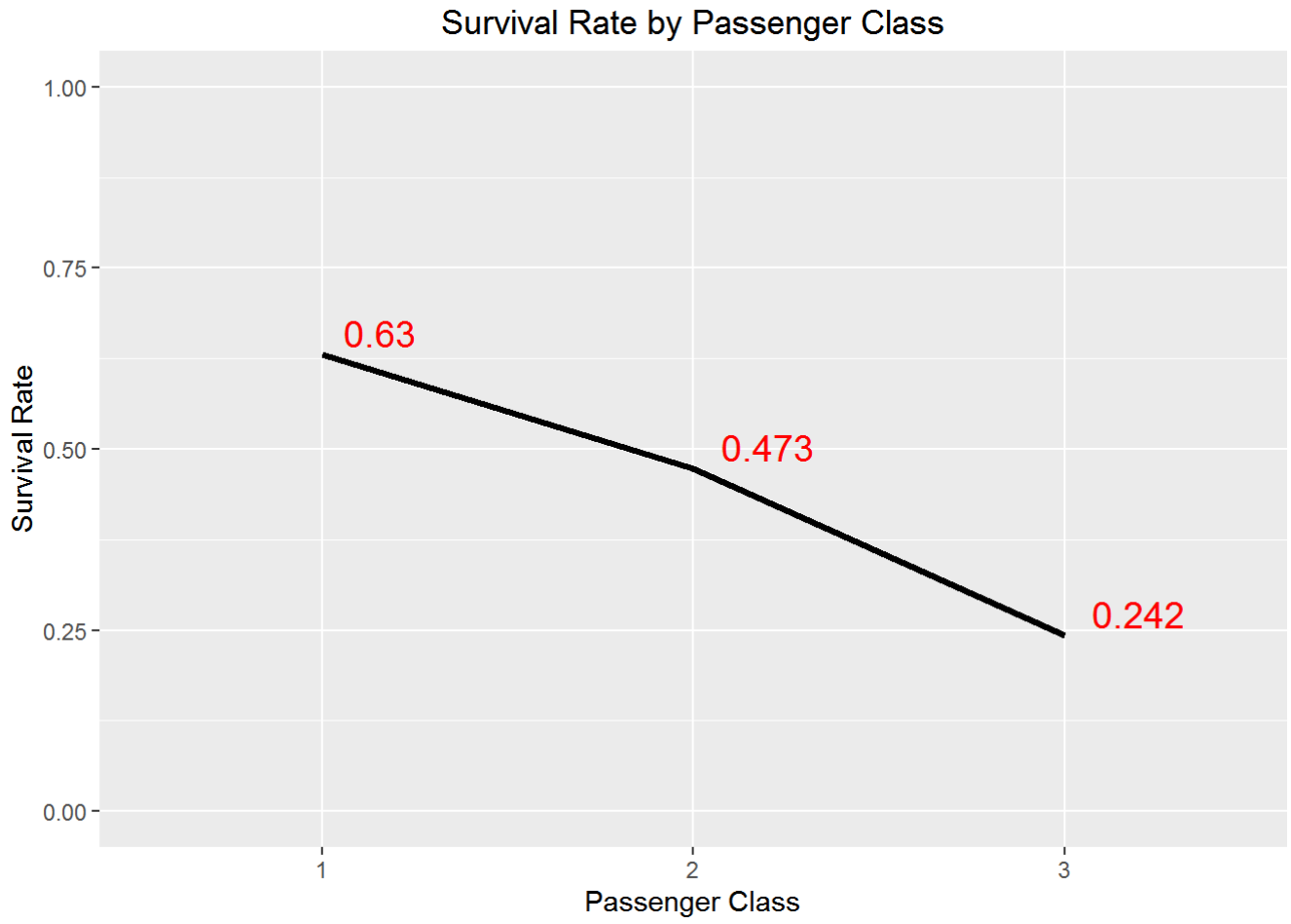
```
ggplot(SurvivalRateByFareClass, aes(x = FareClass, y = survivalRate, group = 1)) +
  geom_line(size = 1.2)+
  geom_text(aes(label = survivalRate), vjust = -.3, hjust = -.3 , size = 5, color = "red") +
  ggtitle("Survival Rate by Fare Class")+
  ylim(0,1)+
  xlab("Fare Class") +
  ylab("Survival Rate")
```



```
SurvivalRateByPclass = full %>%
  filter(Survived != "NA") %>%
  group_by(Pclass) %>%
  summarise(PassengerCount = n(),PassengerSurvived =  sum(Survived == 1),
            survivalRate = round(sum(Survived == 1)/n(),3)) %>%
  arrange(desc(Pclass))
SurvivalRateByPclass
```

```
## Source: local data frame [3 x 4]
##
##   Pclass PassengerCount PassengerSurvived survivalRate
##    (int)          (int)             (int)        (dbl)
## 1      3            491               119        0.242
## 2      2            184                87        0.473
## 3      1            216               136        0.630
```

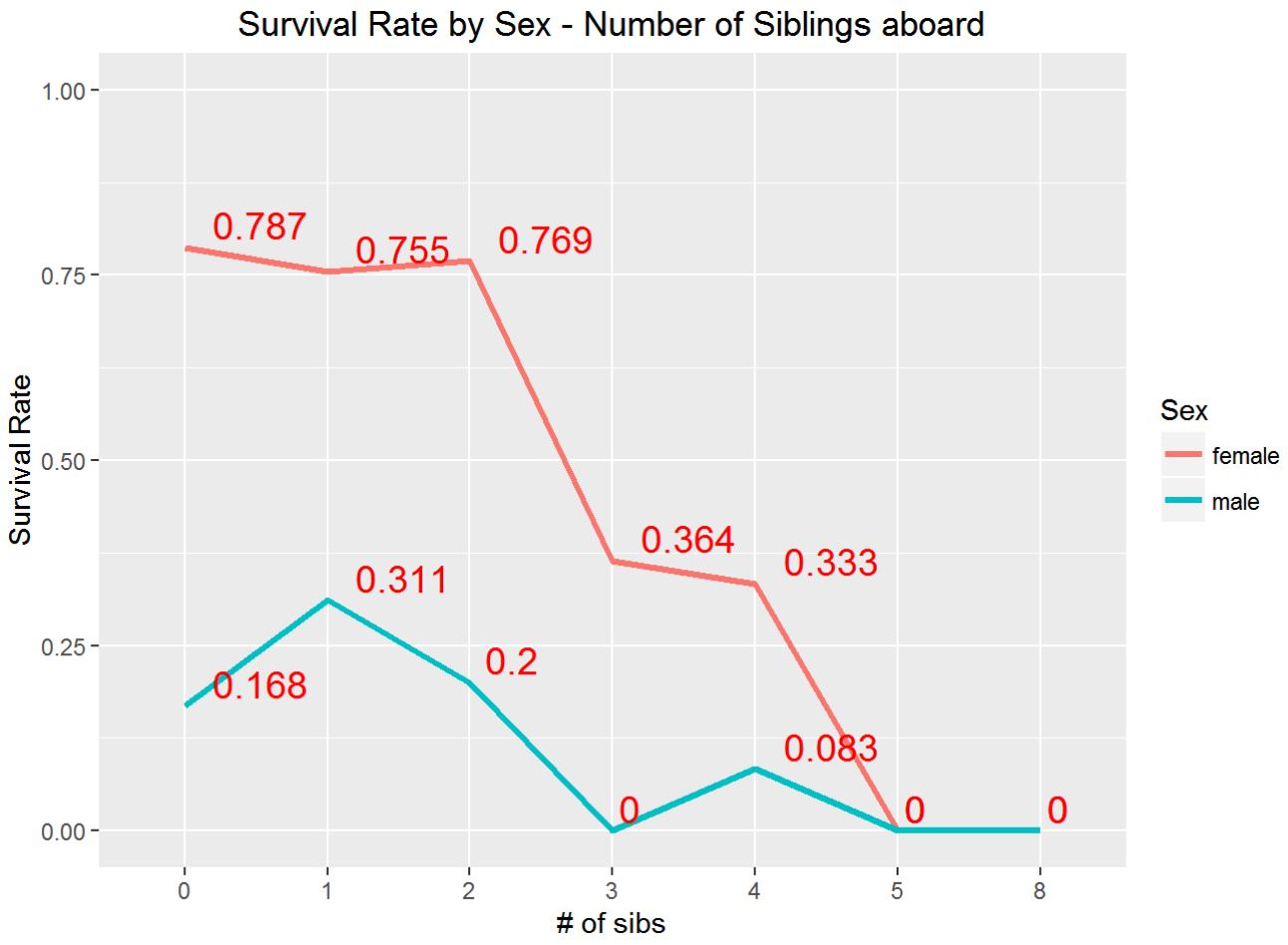
```
ggplot(SurvivalRateByPclass, aes(x = factor(Pclass), y = survivalRate, group = 1)) +
  geom_line(size = 1.2)+
  geom_text(aes(label = survivalRate), vjust = -.3, hjust = -.3 , size = 5, color = "red") +
  ggtitle("Survival Rate by Passenger Class")+
  ylim(0,1)+
  xlab("Passenger Class") +
  ylab("Survival Rate")
```



```
SurvivalRateByGenderSib = full %>%
  filter(Survived != "NA") %>%
  group_by(Sex, SibSp) %>%
  summarise(PassengerCount = n(),PassengerSurvived =  sum(Survived == 1),
            survivalRate = round(sum(Survived == 1)/n(),3))
SurvivalRateByGenderSib
```

```
## Source: local data frame [14 x 5]
## Groups: Sex [?]
##
##       Sex SibSp PassengerCount PassengerSurvived survivalRate
##    (fctr) (int)          (int)             (int)        (dbl)
## 1  female     0            174               137        0.787
## 2  female     1            106                80        0.755
## 3  female     2             13                10        0.769
## 4  female     3             11                 4        0.364
## 5  female     4              6                 2        0.333
## 6  female     5              1                 0        0.000
## 7  female     8              3                 0        0.000
## 8    male     0            434                73        0.168
## 9    male     1            103                32        0.311
## 10   male     2             15                 3        0.200
## 11   male     3              5                 0        0.000
## 12   male     4             12                 1        0.083
## 13   male     5              4                 0        0.000
## 14   male     8              4                 0        0.000
```

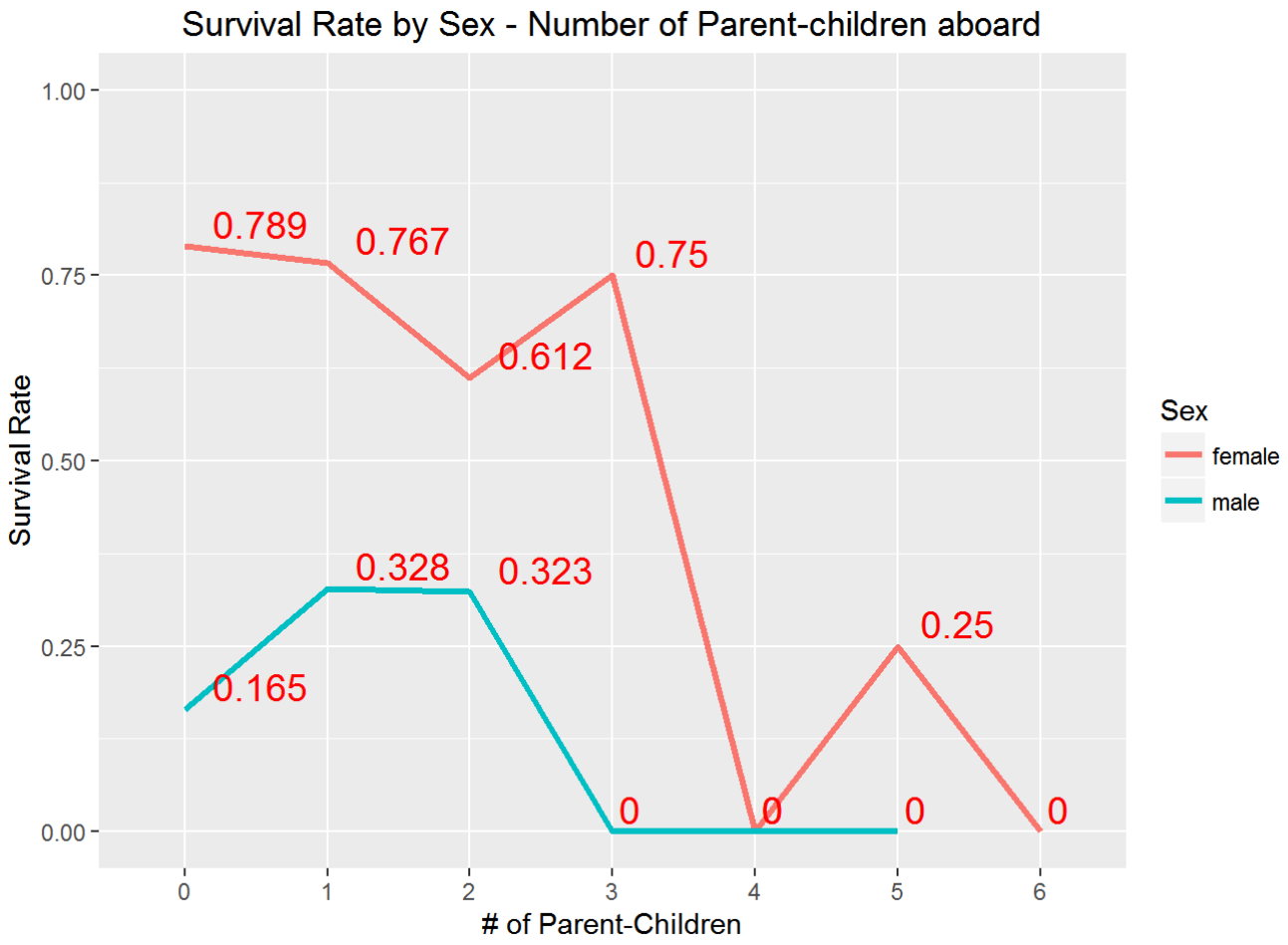
```
ggplot( SurvivalRateByGenderSib, aes(x = factor(SibSp), y = survivalRate, group = interaction(Se
x), color = Sex)) +
  geom_line(size = 1.2) +
  geom_text(aes(label = survivalRate), vjust = -.3, hjust = -.3 , size = 5, color = "red") +
  ggtitle("Survival Rate by Sex - Number of Siblings aboard")+
  ylim(0,1)+
  xlab("# of sibs") +
  ylab("Survival Rate")
```



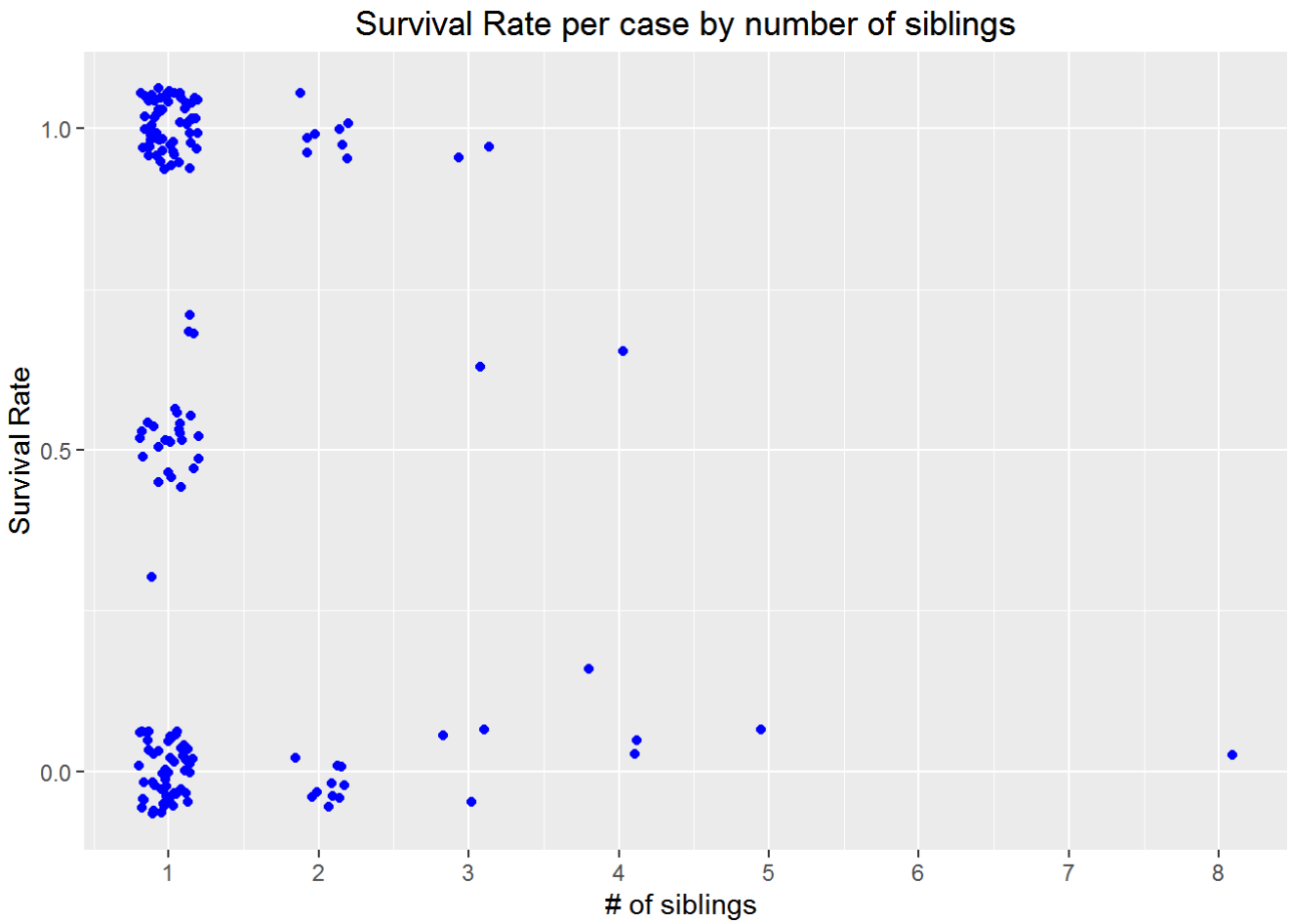

```
SurvivalRateByGenderParch = full %>%
  filter(Survived != "NA") %>%
  group_by(Sex, Parch) %>%
  summarise(PassengerCount = n(),PassengerSurvived =  sum(Survived == 1),
            survivalRate = round(sum(Survived == 1)/n(),3))
SurvivalRateByGenderParch
```

```
## Source: local data frame [13 x 5]
## Groups: Sex [?]
##
##       Sex Parch PassengerCount PassengerSurvived survivalRate
##    (fctr) (int)          (int)             (int)        (dbl)
## 1  female     0            194               153        0.789
## 2  female     1             60                46        0.767
## 3  female     2             49                30        0.612
## 4  female     3              4                 3        0.750
## 5  female     4              2                 0        0.000
## 6  female     5              4                 1        0.250
## 7  female     6              1                 0        0.000
## 8    male     0            484                80        0.165
## 9    male     1             58                19        0.328
## 10   male     2             31                10        0.323
## 11   male     3              1                 0        0.000
## 12   male     4              2                 0        0.000
## 13   male     5              1                 0        0.000
```

```
ggplot( SurvivalRateByGenderParch, aes(x = factor(Parch), y = survivalRate, group =
interaction(Sex), color = Sex)) +
  geom_line(size = 1.2) +
  geom_text(aes(label = survivalRate), vjust = -.3, hjust = -.3 , size = 5, color = "red") +
  ggtitle("Survival Rate by Sex - Number of Parent-children aboard")+
  ylim(0,1)+
  xlab("# of Parent-Children") +
  ylab("Survival Rate")
```



```
# Among the people who travels with siblings/parents, are they survived together or not, this me
thod is not perfect due to limitation of
# knowing whos travel with who, the method that we used here is just based on the surname, a spo
use may have different surname which may cause
# error in this analysis

full$Surname = str_extract(full$Name, pattern = "[A-z]*" )

SurvivalRateSibSp = full %>%   filter(Survived != "NA" & SibSp != 0) %>%
  group_by(SibSp, Surname) %>%
  summarise(PassengerCount = n(),PassengerSurvived =sum(Survived == 1),
            survivalRate = round(sum(Survived == 1)/n(),3)) %>%
  arrange(desc(survivalRate))

ggplot(SurvivalRateSibSp, aes(x = SibSp, y = survivalRate)) +
  scale_x_continuous( breaks = seq(0,8,1))+
  scale_y_continuous( breaks = seq(0,1,.5))+
  xlab("# of siblings ") +
  ylab("Survival Rate") +
  ggtitle("Survival Rate per case by number of siblings") +
  geom_jitter(width =.5, color = "blue")
```



```
# observation : Most people with siblings/ traveling together chose to either survive together o
r not survive together

#########################################################################
# PART 2 - IMPUTATION OF MISSING VALUES USING MULTIPLE ALGORITHM (MEDIAN, MISSFORREST, AND MICE)
# We ended up using missforrest algorithm, which yield us the best score among three
#########################################################################


# Next, we need to work on NAs in Age column, we will use library(missRandom) and library(mice)
 for the imputation of missing values
# in Age column

# To compare before and after missing value imputation, we will do exploratory analysis before a
nd after missing value imputation

# 1st approach we predict the missing Age using median by gender/sex
# this approach yield to the best score we can achieve so far

# medians = aggregate(Age ~  Sex, full, median)
# medians

# full$Age[is.na(full$Age) & full$Sex == "female"] = 27
# full$Age[is.na(full$Age) & full$Sex == "male"] = 28

# 2nd approach we predict the missing Age using missForrest Package, which predict missing value
s using random forrest
# eliminating columns: PassengerId,Survived,Name,Parch, Ticket,Cabin, because Age should not be
 correlated to these columns

temp = full[-c(1,2,4,8,9,11,16)]
head(temp)
```

```
##   Pclass    Sex Age SibSp    Fare    Embarked honorific GenderAgeClass
## 1      3   male  22     1  7.2500 Southampton       Mr.     Adult Male
## 2      1 female  38     1 71.2833   Cherbourg      Mrs.         Female
## 3      3 female  26     0  7.9250 Southampton     Miss.         Female
## 4      1 female  35     1 53.1000 Southampton      Mrs.         Female
## 5      3   male  35     0  8.0500 Southampton       Mr.     Adult Male
## 6      3   male  NA     0  8.4583  Queenstown       Mr.     Adult Male
##   FareClass
## 1  cheapest
## 2   highest
## 3     cheap
## 4   highest
## 5     cheap
## 6     cheap
```

```
set.seed(100)
Age.rf = missForest(temp)
```

```
##   missForest iteration 1 in progress...done!
##   missForest iteration 2 in progress...done!
##   missForest iteration 3 in progress...done!
##   missForest iteration 4 in progress...done!
```

```
full$Age = Age.rf$ximp$Age
summary(temp)
```

```
##      Pclass          Sex           Age            SibSp
##  Min.   :1.000   female:466   Min.   : 0.17   Min.   :0.0000
##  1st Qu.:2.000   male  :843   1st Qu.:21.00   1st Qu.:0.0000
##  Median :3.000                Median :28.00   Median :0.0000
##  Mean   :2.295                Mean   :29.88   Mean   :0.4989
##  3rd Qu.:3.000                3rd Qu.:39.00   3rd Qu.:1.0000
##  Max.   :3.000                Max.   :80.00   Max.   :8.0000
##                               NA's   :263
##       Fare                Embarked     honorific      GenderAgeClass
##  Min.   :  0.000   Unknown    :  2   Mr.    :757   Adult Male:783
##  1st Qu.:  7.896   Cherbourg  :270   Miss.  :260   Female    :465
##  Median : 14.454   Queenstown :123   Mrs.   :197   Young Male: 61
##  Mean   : 33.276   Southampton:914   Master.: 61
##  3rd Qu.: 31.275                     Dr.    :  8
##  Max.   :512.329                     Rev.   :  8
##                                      (Other): 18
##     FareClass
##  highest :271
##  high    :250
##  medium  :293
##  cheap   :241
##  cheapest:254
##
##
```

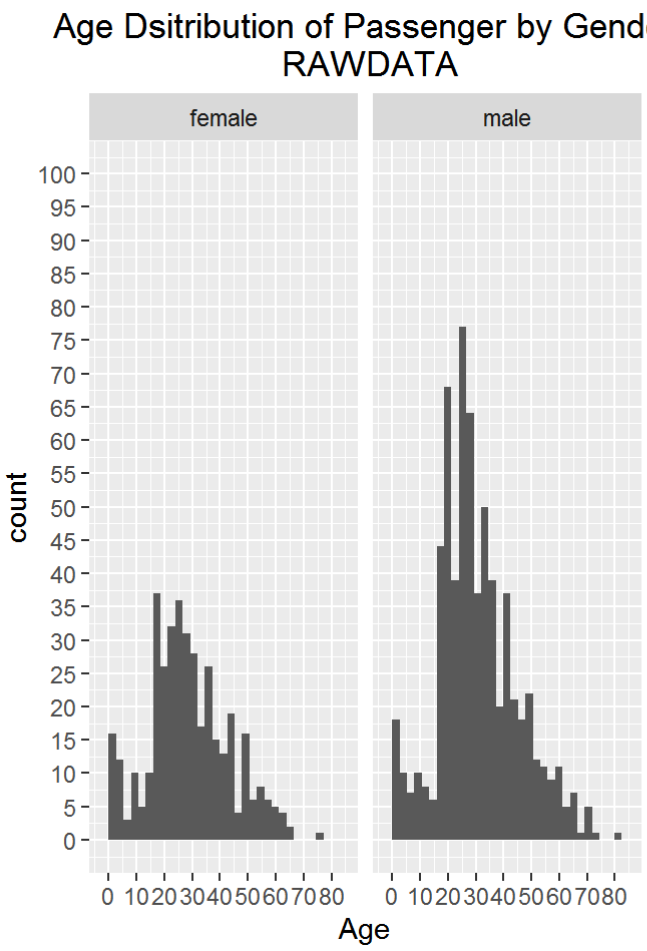
```
plotAge.rf1 = ggplot(full[!is.na(full$Age),], aes(x = Age )) +
  geom_histogram() +
  facet_grid(~Sex) +
  coord_cartesian(ylim =  seq(0,150,5))+
  scale_y_continuous(breaks = seq(0,150,5)) +
  scale_x_continuous(breaks = seq(0,90,10)) +
  ggtitle("Age Dsitribution of Passenger by Gender\n MISSFORREST")

plotAge.rf2 = ggplot(full[!is.na(full$Age),], aes(x = Age, fill = as.factor(Sex))) +
  geom_histogram(position = "identity", alpha = .4) +
  scale_fill_manual(values=c("green", "purple")) +
  scale_y_continuous(breaks = seq(0,150,5))+
  scale_x_continuous(breaks = seq(0,90,10)) +
  ggtitle("Age Dsitribution of Passenger by Gender\n MISSFORREST") +
  labs(fill="")

grid.arrange(plotAgeRaw2, plotAge.rf2 , ncol=2)
```

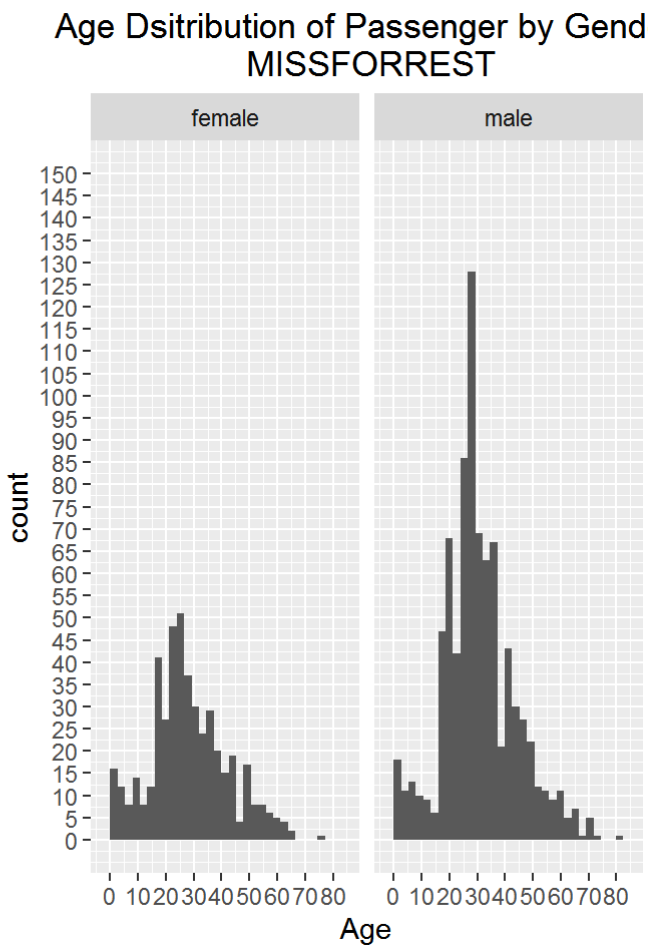
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


```
grid.arrange(plotAgeRaw1, plotAge.rf1 , ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


Age Dsitribution of Passenger by Gende
RAWDATA
female
male
count
Age


Age Dsitribution of Passenger by Gende
MISSFORREST
female
male
count
Age

```
# 3rd approach we predict the missing Age using MICE Package, this time we will use "pmm" metho
d, (predictive mean matching)
# eliminating columns: PassengerId,Survived,Name,Parch, Ticket,Cabin, because Age should not be
 correlated to these columns
# temp2 = full[-c(1,2,4,9,11,16)]
# head(temp2)
# set.seed(100)
# micePmm = mice(temp2 , method = "norm")
# # micePmm$imp$Age$`3`
# micePmm$method
# temp2  = complete(micePmm,3)
# full$Age = temp2$Age
#
# plotAge.mice1 = ggplot(full[!is.na(full$Age),], aes(x = Age )) +
#   geom_histogram() +
#   facet_grid(~Sex) +
#   coord_cartesian(ylim =  seq(0,100,5))+
#   scale_y_continuous(breaks = seq(0,100,5)) +
#   scale_x_continuous(breaks = seq(0,90,10)) +
#   ggtitle("Age Dsitribution of Passenger by Gender\n MICE")
#
# plotAge.mice2 = ggplot(full[!is.na(full$Age),], aes(x = Age, fill = as.factor(Sex))) +
#   geom_histogram(position = "identity", alpha = .4) +
#   scale_fill_manual(values=c("green", "purple")) +
#   scale_y_continuous(breaks = seq(0,100,5))+
#   scale_x_continuous(breaks = seq(0,90,10)) +
#   ggtitle("Age Dsitribution of Passenger by Gender\n MICE") +
#   labs(fill="")
#
# medians = aggregate(Age ~  Sex, full, median)
#
# plotAge.mice3 = ggplot(full[!is.na(full$Age),], aes(x = Sex, y = Age)) +
#   geom_boxplot() +
#   scale_y_continuous(breaks = seq(0,80,5)) +
#   geom_text( data = medians, aes( label = Age, y = Age), vjust = -.5) +
#   ggtitle("Age Dsitribution of Passenger by Gender\n MICE")
#
#
# grid.arrange(plotAgeRaw1, plotAge.mice1 , ncol=2)
# grid.arrange(plotAgeRaw2, plotAge.mice2 , ncol=2)
# grid.arrange(plotAgeRaw3, plotAge.mice3 , ncol=2)


train = full[!is.na(full$Survived),]
test = full[is.na(full$Survived),]

summary(train)
```

```
##   PassengerId       Survived          Pclass          Name
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Length:891
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median :446.0   Median :0.0000   Median :3.000   Mode  :character
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##      Sex           Age            SibSp           Parch
##  female:314   Min.   : 0.42   Min.   :0.000   Min.   :0.0000
##  male  :577   1st Qu.:21.00   1st Qu.:0.000   1st Qu.:0.0000
##               Median :28.58   Median :0.000   Median :0.0000
##               Mean   :29.58   Mean   :0.523   Mean   :0.3816
##               3rd Qu.:36.70   3rd Qu.:1.000   3rd Qu.:0.0000
##               Max.   :80.00   Max.   :8.000   Max.   :6.0000
##
##       Ticket         Fare                Cabin            Embarked
##  1601    :  7   Min.   :  0.00             :687   Unknown    :  2
##  347082  :  7   1st Qu.:  7.91   B96 B98    :  4   Cherbourg  :168
##  CA. 2343:  7   Median : 14.45   C23 C25 C27:  4   Queenstown : 77
##  3101295 :  6   Mean   : 32.20   G6         :  4   Southampton:644
##  347088  :  6   3rd Qu.: 31.00   C22 C26    :  3
##  CA 2144 :  6   Max.   :512.33   D          :  3
##  (Other) :852                    (Other)    :186
##    honorific      GenderAgeClass    FareClass     Surname
##  Mr.    :517   Adult Male:538    highest :182   Length:891
##  Miss.  :182   Female    :313    high    :174   Class :character
##  Mrs.   :125   Young Male: 40    medium  :196   Mode  :character
##  Master.: 40                     cheap   :173
##  Dr.    :  7                     cheapest:166
##  Rev.   :  6
##  (Other): 14
```

```
summary(test)
```

```
##   PassengerId        Survived       Pclass          Name
##  Min.   : 892.0   Min.   : NA   Min.   :1.000   Length:418
##  1st Qu.: 996.2   1st Qu.: NA   1st Qu.:1.000   Class :character
##  Median :1100.5   Median : NA   Median :3.000   Mode  :character
##  Mean   :1100.5   Mean   :NaN   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.: NA   3rd Qu.:3.000
##  Max.   :1309.0   Max.   : NA   Max.   :3.000
##                   NA's   :418
##      Sex           Age            SibSp            Parch
##  female:152   Min.   : 0.17   Min.   :0.0000   Min.   :0.0000
##  male  :266   1st Qu.:22.00   1st Qu.:0.0000   1st Qu.:0.0000
##               Median :28.18   Median :0.0000   Median :0.0000
##               Mean   :30.01   Mean   :0.4474   Mean   :0.3923
##               3rd Qu.:36.88   3rd Qu.:1.0000   3rd Qu.:0.0000
##               Max.   :76.00   Max.   :8.0000   Max.   :9.0000
##
##       Ticket         Fare                      Cabin            Embarked
##  PC 17608:  5   Min.   :  0.000                  :327   Unknown    :  0
##  113503  :  4   1st Qu.:  7.896   B57 B59 B63 B66:  3   Cherbourg  :102
##  CA. 2343:  4   Median : 14.454   A34            :  2   Queenstown : 46
##  16966   :  3   Mean   : 35.562   C101           :  2   Southampton:270
##  220845  :  3   3rd Qu.: 31.472   C23 C25 C27    :  2
##  347077  :  3   Max.   :512.329   C78            :  2
##  (Other) :396                     (Other)        : 80
##    honorific      GenderAgeClass    FareClass    Surname
##  Mr.    :240   Adult Male:245    highest :89   Length:418
##  Miss.  : 78   Female    :152    high    :76   Class :character
##  Mrs.   : 72   Young Male: 21    medium  :97   Mode  :character
##  Master.: 21                     cheap   :68
##  Col.   :  2                     cheapest:88
##  Rev.   :  2
##  (Other):  3
```

```
# we need to tune our random forrest, we need to find the best number of variables "mtry"
set.seed(100)
tuneRF(x = train[,c(3,5,7,8,10,12,13,14,15,6)],
       y = train[,2],
       stepFactor=1.5,
       improve=1e-5)
```

```
## Warning in randomForest.default(x, y, mtry = mtryStart, ntree = ntreeTry, :
## The response has five or fewer unique values. Are you sure you want to do
## regression?
```

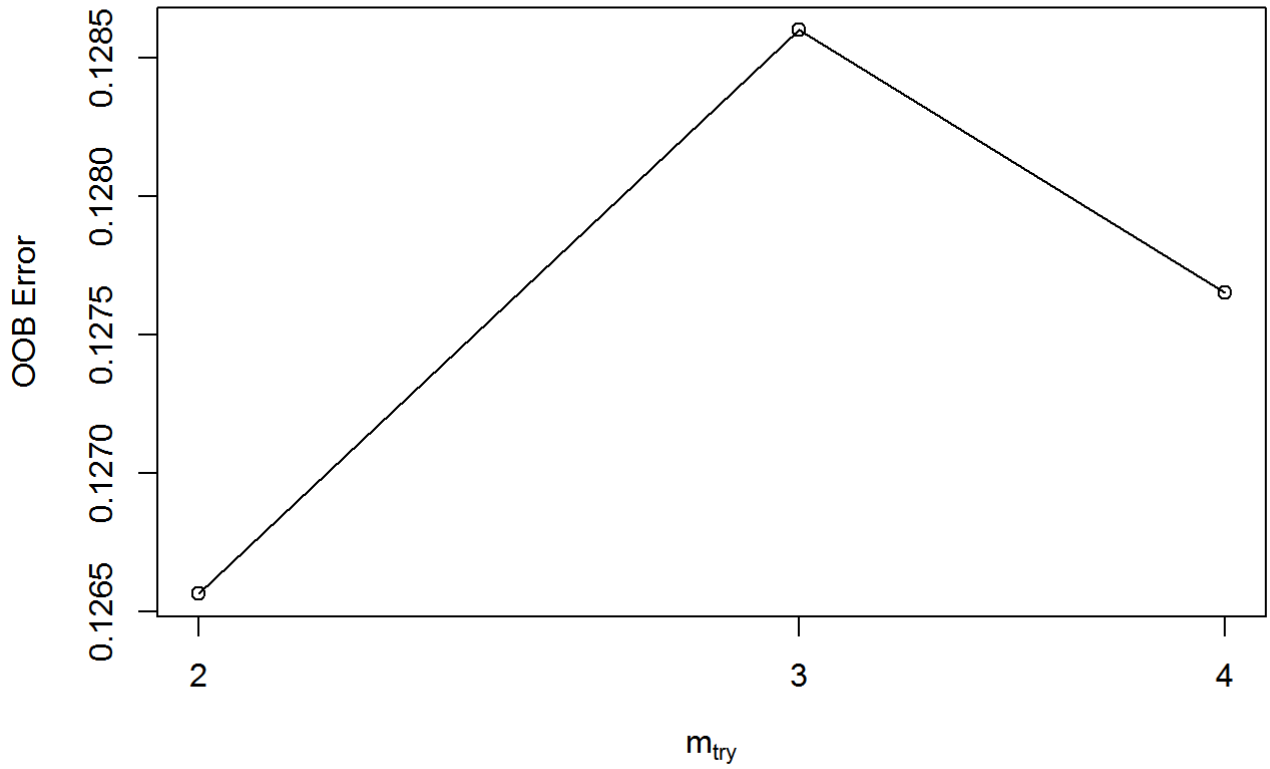
```
## mtry = 3  OOB error = 0.1285991
## Searching left ...
```

```
## Warning in randomForest.default(x, y, mtry = mtryCur, ntree = ntreeTry, :
## The response has five or fewer unique values. Are you sure you want to do
## regression?
```

```
## mtry = 2     OOB error = 0.1265639
## 0.01582604 1e-05
## Searching right ...
```

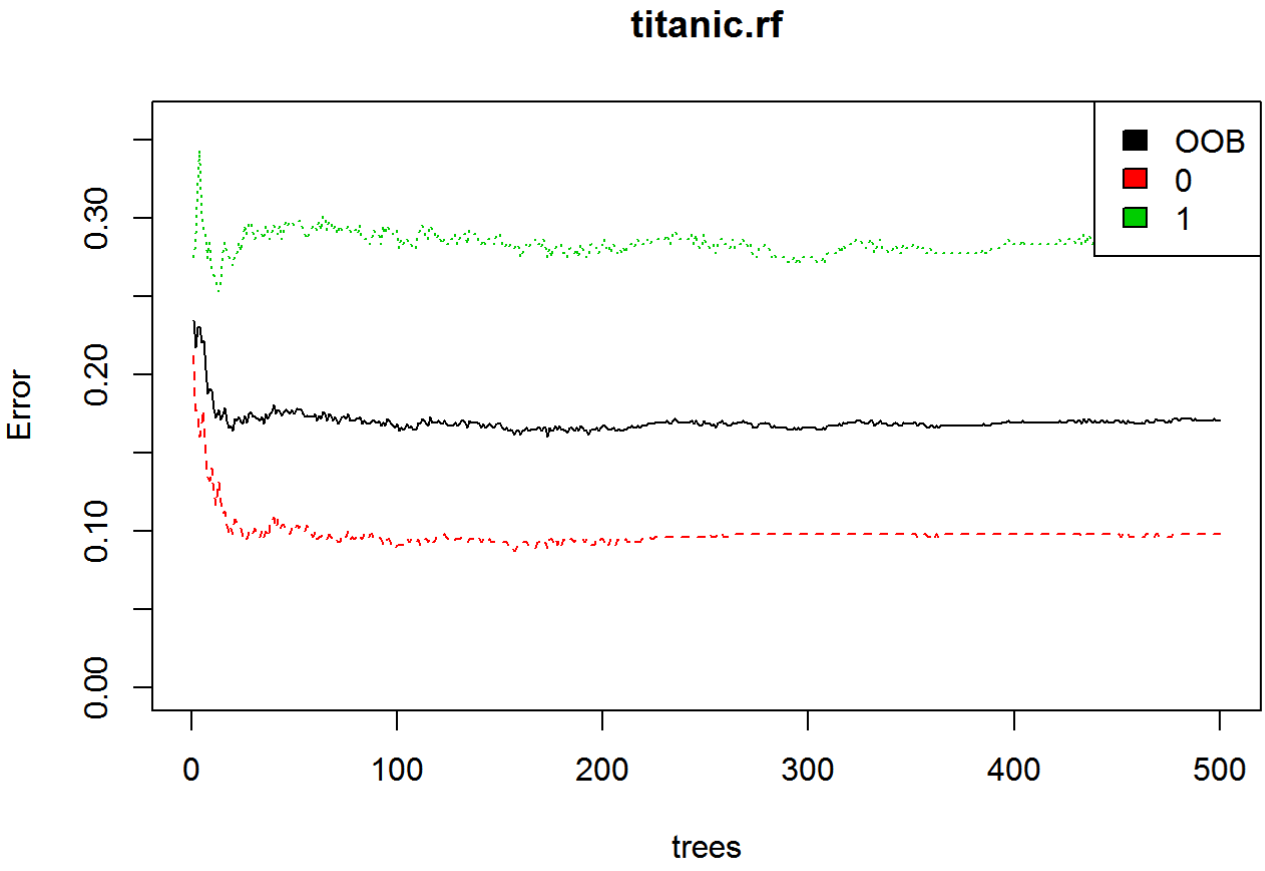
```
## Warning in randomForest.default(x, y, mtry = mtryCur, ntree = ntreeTry, :
## The response has five or fewer unique values. Are you sure you want to do
## regression?
```

```
## mtry = 4     OOB error = 0.1276504
## -0.008584952 1e-05
```




```
##   mtry  OOBError
## 2    2 0.1265639
## 3    3 0.1285991
## 4    4 0.1276504
```

```
set.seed(100)
titanic.rf = randomForest(factor(Survived) ~ Pclass + Sex + SibSp  + Parch + Fare + Embarked + h
onorific + GenderAgeClass + FareClass + Age ,
                          data = train,
                          mtry = 2)

plot(titanic.rf, ylim=c(0,0.36))
legend('topright', colnames(titanic.rf$err.rate), col=1:3, fill=1:3)
```




```
prediction = predict(titanic.rf, test)
submission = data.frame(PassengerId = test$PassengerId, Survived = prediction)
write.csv(submission, "submission.csv", row.names = FALSE)

importance(titanic.rf)
```

```
##                MeanDecreaseGini
## Pclass                28.804295
## Sex                   32.341933
## SibSp                 15.709107
## Parch                  8.861323
## Fare                  41.388014
## Embarked               8.501074
## honorific             56.224614
## GenderAgeClass        43.983455
## FareClass             16.228756
## Age                   34.340542
```

```
#############################################################################
# END
#############################################################################









#############################################################################
# Experiment using Trees - rpart package
#############################################################################

# library(rpart)
# set.seed(100)
# titanic.rp = rpart(factor(Survived) ~ Pclass + Sex + SibSp  + Parch + Fare + Embarked + honori
fic + GenderAgeClass + FareClass + Age ,
#                           data = train)
#
# plot(titanic.rp, uniform =TRUE)
# text(titanic.rp, use.n = TRUE, all = TRUE)
# prediction = predict(titanic.rp, test, type = "class")
# submission = data.frame(PassengerId = test$PassengerId, Survived = prediction)
# write.csv(submission, "submission.csv", row.names = FALSE)
```