Data Exploration Using R

Data Science Dojo

Data Exploration Using R

Titanic tragedy data

Reading RAW training data

- Download the data set "Titanic_train.csv" from https://raw.githubusercontent.com/datasciencedojo/datasets/master/Titanic_train.csv
- Set working directory of R to the directory of the file using setwd()

```
titanic = read.csv("Titanic_train.csv"); colnames(titanic)
```

```
##  [1] "PassengerId" "Survived"    "Pclass"      "Name"        "Sex"
##  [6] "Age"         "SibSp"       "Parch"       "Ticket"      "Fare"
## [11] "Cabin"       "Embarked"
```

```
colnames(titanic)[5] <- "Gender" ## revise the name of a column
```

Look at the first few rows

```
head(titanic[,1:5], 4)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
##                                                 Name Gender
## 1                            Braund, Mr. Owen Harris   male
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female
## 3                             Heikkinen, Miss. Laina female
## 4       Futrelle, Mrs. Jacques Heath (Lily May Peel) female
```

Look at the first few rows

What would be some good features to be considered?

```
head(titanic[,6:ncol(titanic)], 4)
```

```
##   Age SibSp Parch           Ticket    Fare Cabin Embarked
## 1  22     1     0        A/5 21171  7.2500              S
## 2  38     1     0         PC 17599 71.2833   C85        C
## 3  26     0     0 STON/O2. 3101282  7.9250              S
## 4  35     1     0           113803 53.1000  C123        S
```

Summary of the data frame

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 416 58
##  $ Gender     : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 525 596 662 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
```

Converting class label to a factor

```
titanic$Survived = factor(titanic$Survived, labels=c("died", "survived"))
titanic$Embarked = factor(titanic$Embarked, labels=c("Unkown", "Cherbourg", "Queenstown", "Southampton"))
str(titanic$Survived)
```

```
##  Factor w/ 2 levels "died","survived": 1 2 2 2 1 1 1 1 2 2 ...
```

```
str(titanic$Gender)
```

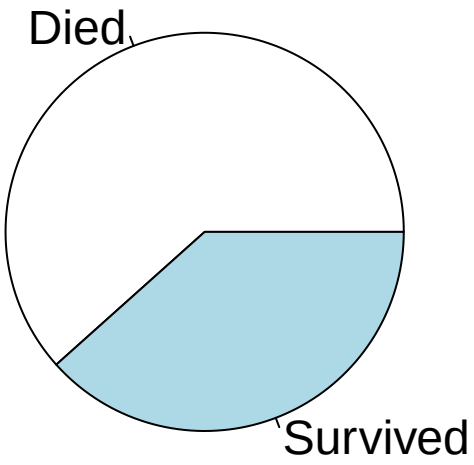
```
##  Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
```

Class (survived or died) distribution - PIE Charts

```
survivedTable = table(titanic$Survived); survivedTable
```

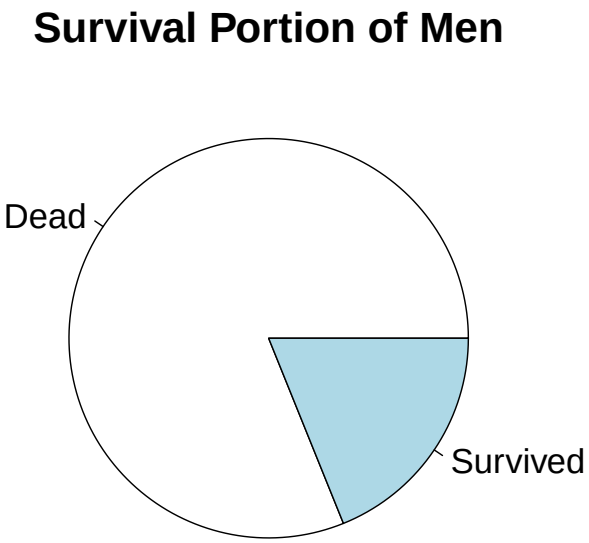
```
##
##     died survived
##      549      342
```

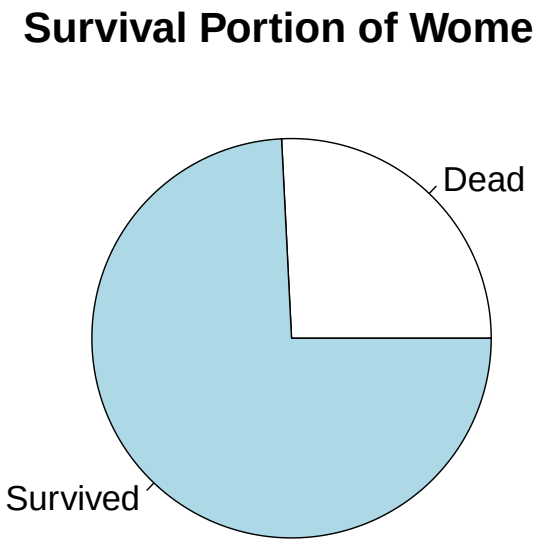
```
par(mar = c(0, 0, 0, 0), oma = c(0, 0, 0, 0), cex=1.5)
pie(survivedTable,labels=c("Died","Survived"))
```



Is Gender a good predictor?

```
par(mfrow = c(1, 2), mar = c(0, 0, 2, 0), oma = c(0, 1, 0, 1), cex=1.5)
pie(table(titanic[titanic$Gender=="male", "Survived"]), labels=c("Dead","Survived"),  main="Survival Po
pie(table(titanic[titanic$Gender=="female", "Survived"]), labels=c("Dead","Survived"), main="Survival P
```





Is Age a good predictor?

```
Age <- titanic$Age; summary(Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

How about summary segmented by survival?

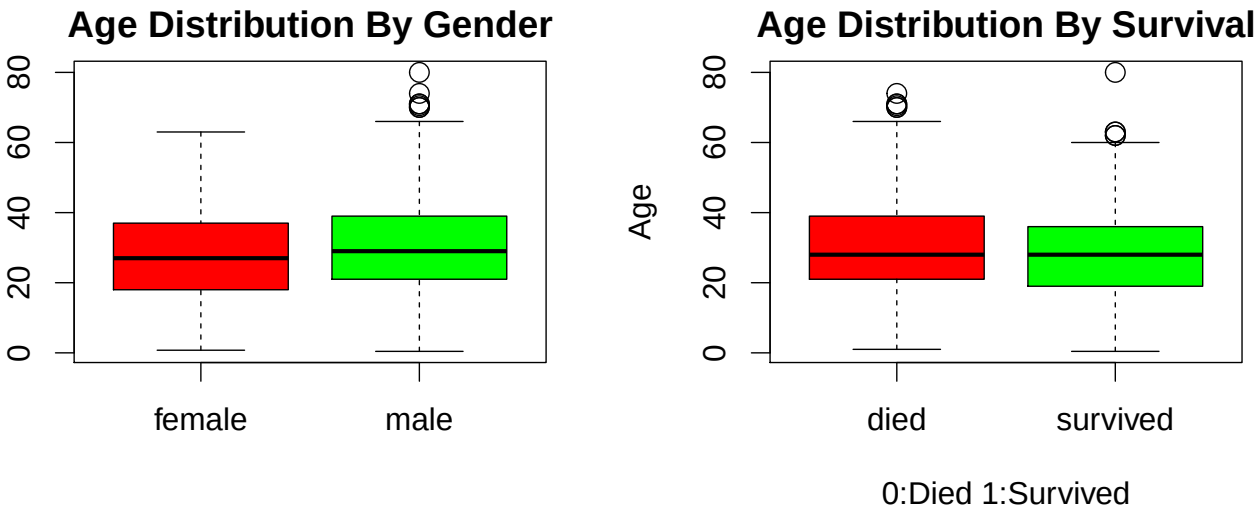
```
summary(titanic[titanic$Survived=="died", "Age"])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    1.00   21.00   28.00   30.63   39.00   74.00     125
```

```
summary(titanic[titanic$Survived=="survived", "Age"])
```

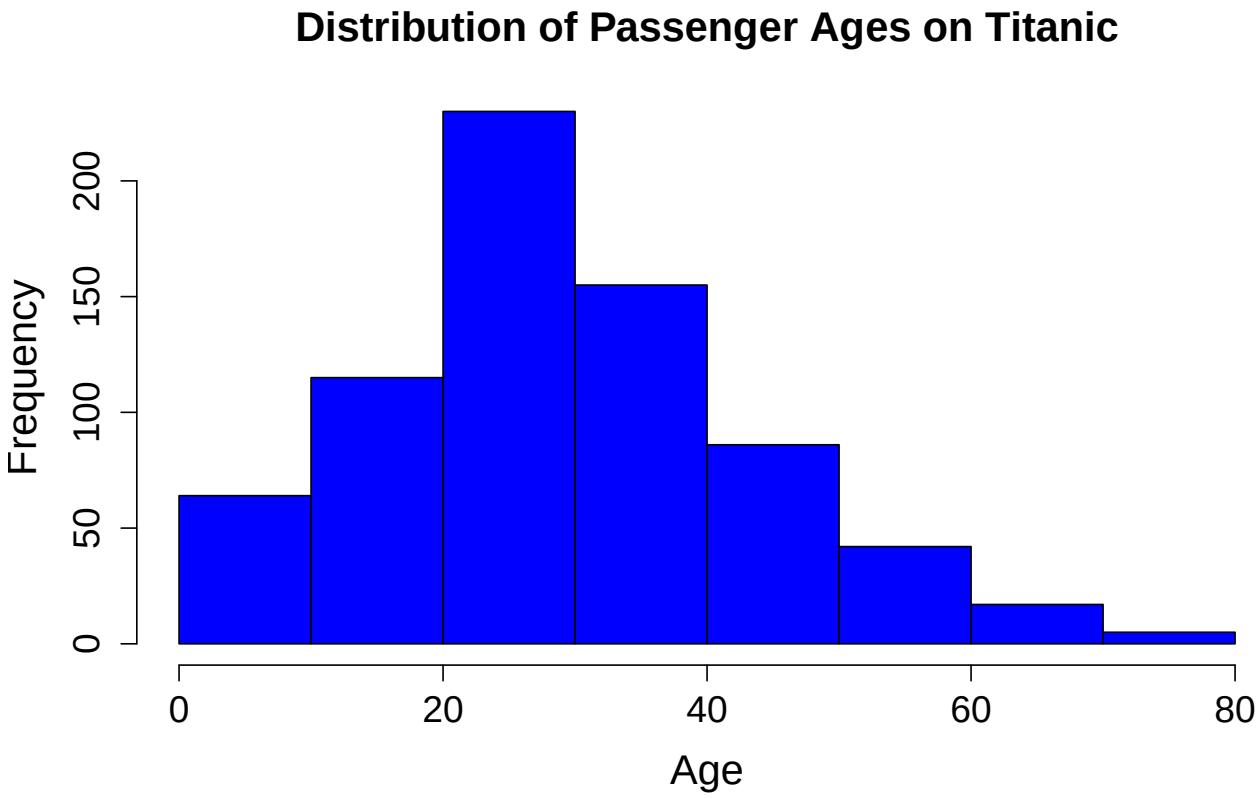
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   19.00   28.00   28.34   36.00   80.00      52
```

Age distribution by Survival and Gender



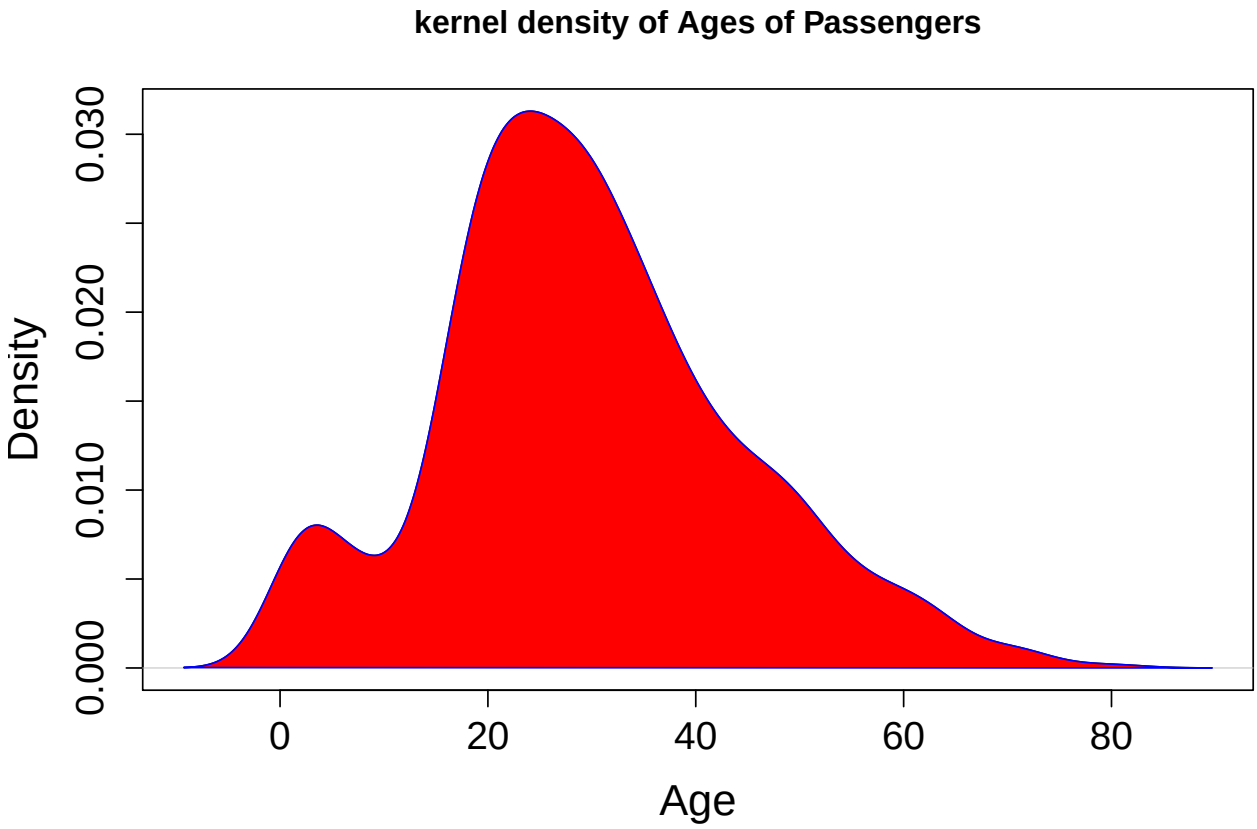
Histogram of Age

```r
hist(Age, col="blue", xlab="Age", ylab="Frequency",
     main = "Distribution of Passenger Ages on Titanic",
     cex.lab=1.6, cex.axis=1.4, cex.main=1.6)
```



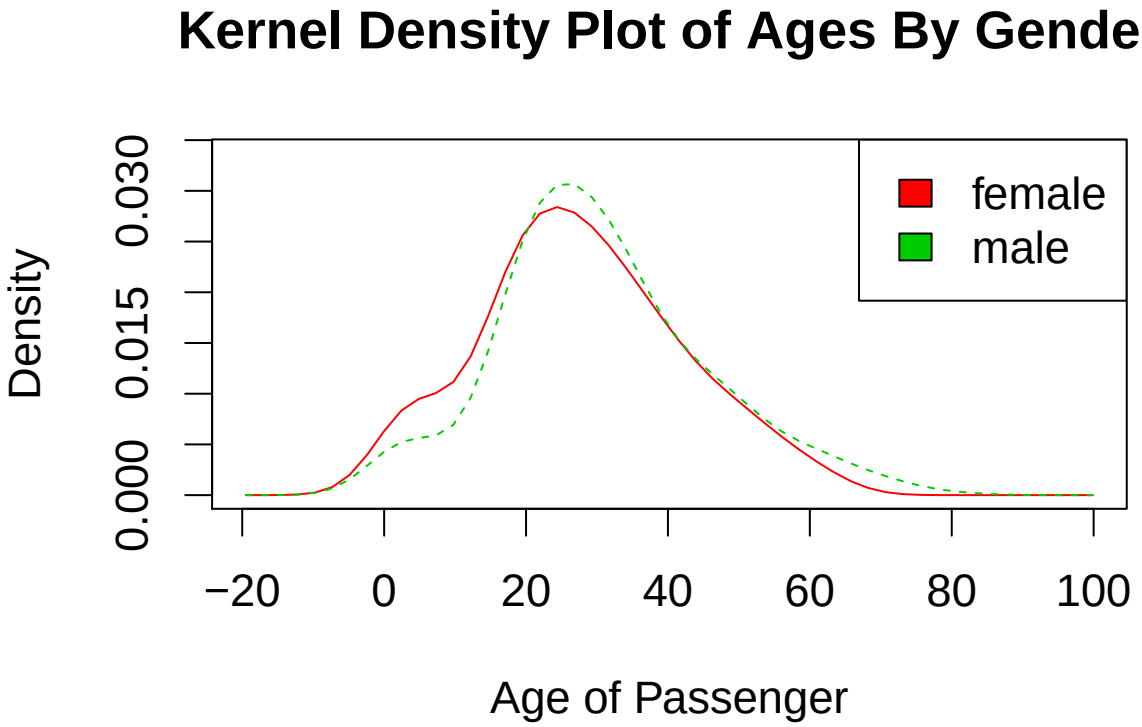
Kernel density plot of Age

```r
d = density(na.omit(Age)) # density() requires all NAs to be removed
plot(d, main = "kernel density of Ages of Passengers", xlab="Age", cex.lab=1.6, cex.axis=1.4); polygon(
```

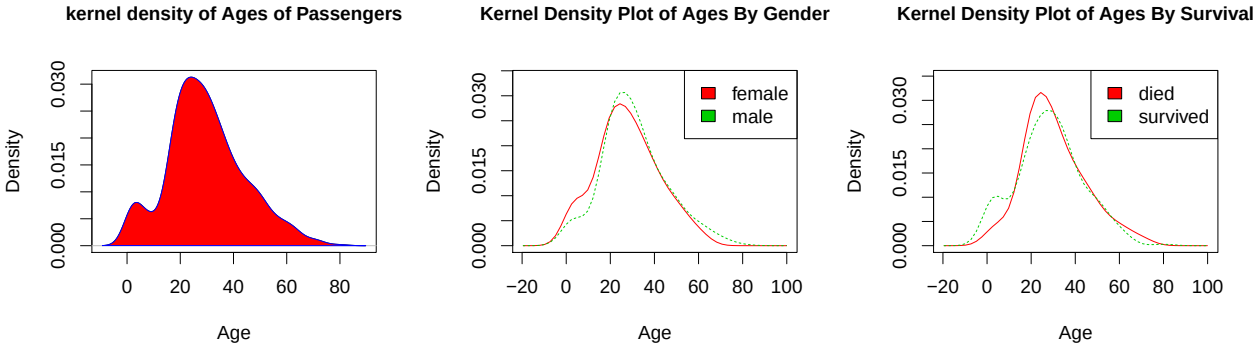


Comparison of density plots of Age with different Gender

```
## Package 'sm', version 2.2-5.4: type help(sm) for summary information
```



Did Age have an impact on survival?



Create categorical groups: Adult vs Child

An example of **feature engineering**!

```
## Multi dimensional comparison
Child <- titanic$Age # Isolating age.
## Now we need to create categories: NA = Unknown, 1 = Child, 2 = Adult
## Every age below 13 (exclusive) is classified into age group 1
Child[Child<13] <- 1
## Every child 13 or above is classified into age group 2
```

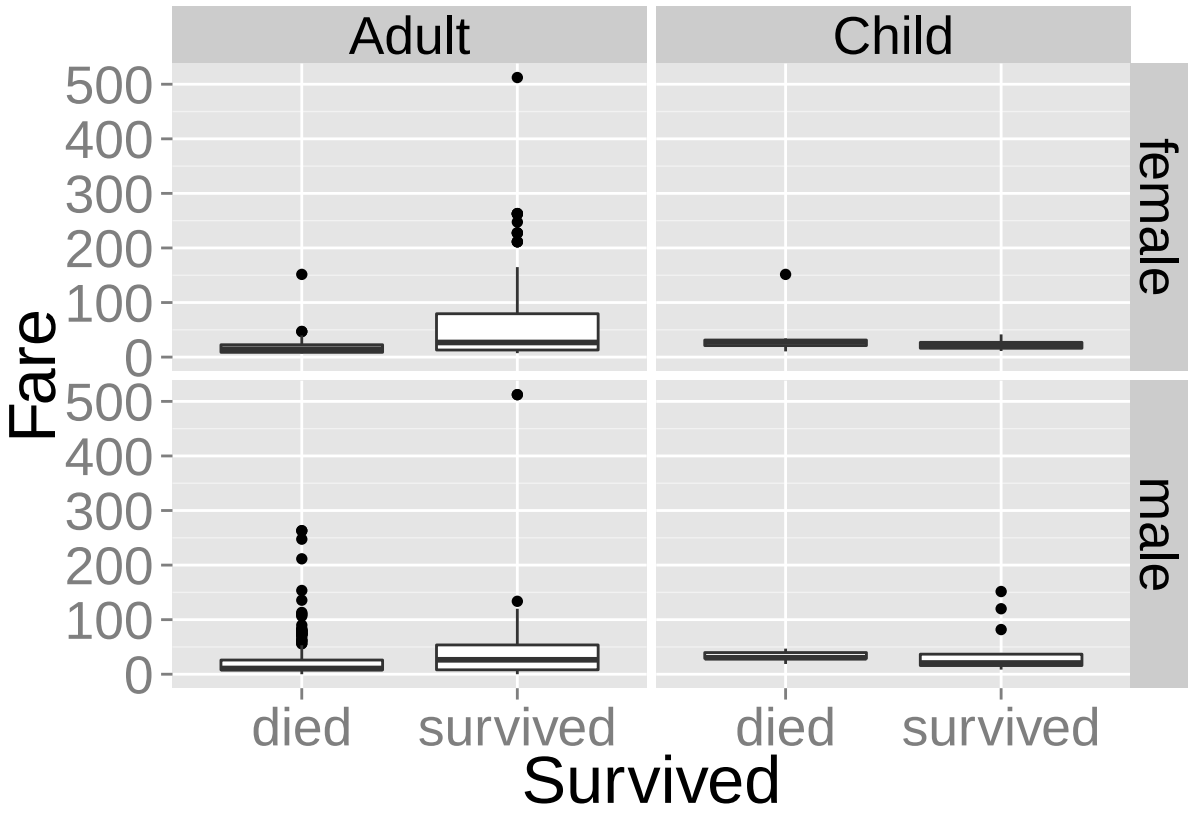
```
Child[Child>=13] <- 2
## Use labels instead of 0's and 1's
Child[Child==1] <- "Child"
Child[Child==2] <- "Adult"
```

Create categorical groups: Adult vs Child

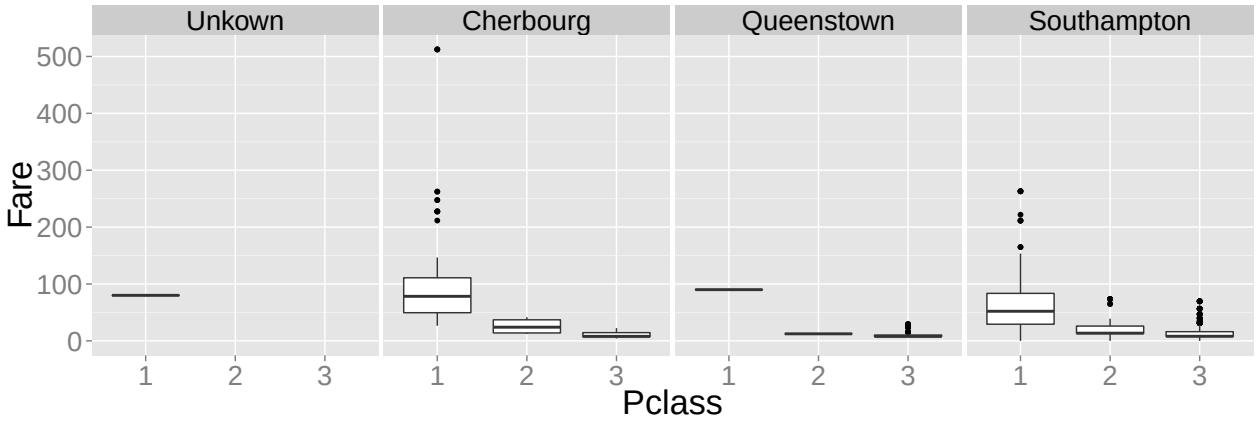
```
# Appends the new column to the titanic dataset
titanic_with_child_column <- cbind(titanic, Child)
# Removes rows where age is NA
titanic_with_child_column <- titanic_with_child_column[!is.na(titanic_with_child_column$Child),]
## Show part of the data frame with new feature: Child
head(titanic_with_child_column[,c(1:3, 13)], 4)
```

```
##   PassengerId Survived Pclass Child
## 1           1     died      3 Adult
## 2           2 survived      1 Adult
## 3           3 survived      3 Adult
## 4           4 survived      1 Adult
```

Fare matters?



How about fare, ship class, port embarkation?



Diamond data

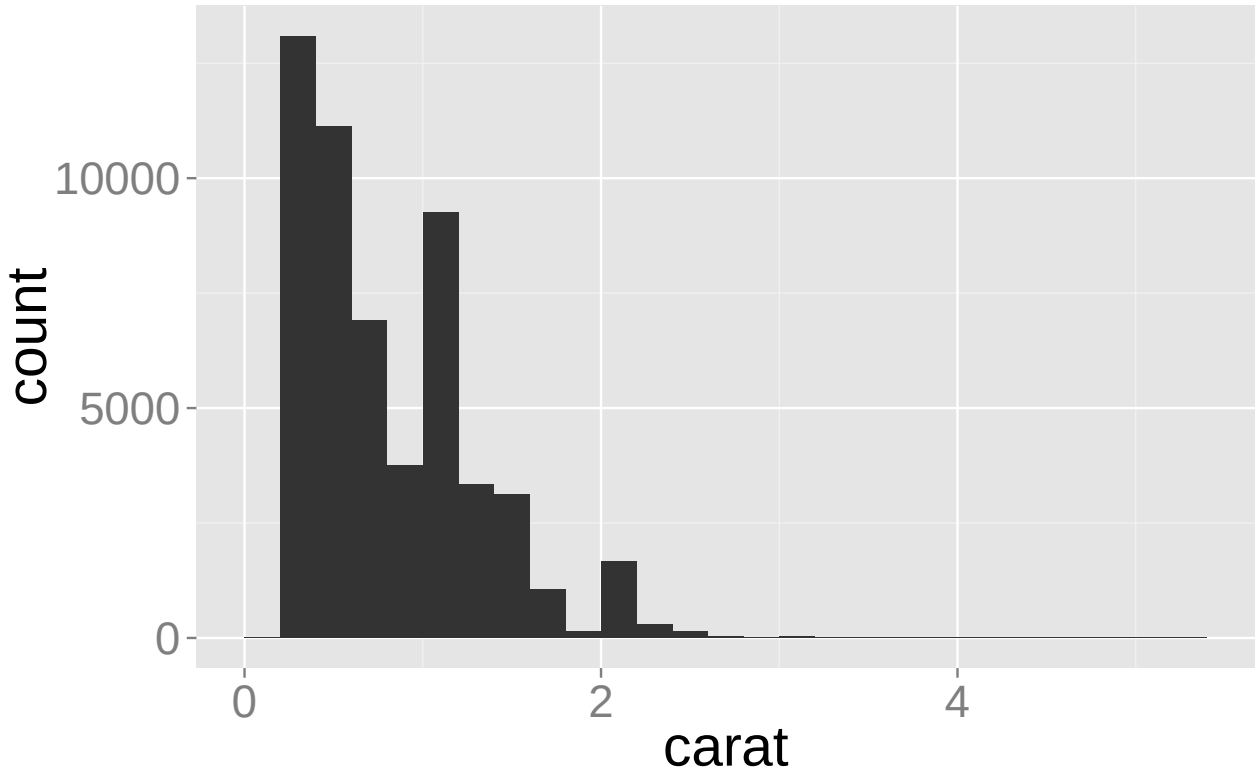
Overview of the diamond data

```
data(diamonds) # loading diamonds data set
head(diamonds, 9) # first few rows of diamond data set
```

```
##   carat       cut color clarity depth table price    x    y    z
## 1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
## 2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
## 3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
## 4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
## 5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
## 6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
## 7  0.24 Very Good     I    VVS1  62.3    57   336 3.95 3.98 2.47
## 8  0.26 Very Good     H     SI1  61.9    55   337 4.07 4.11 2.53
## 9  0.22      Fair     E     VS2  65.1    61   337 3.87 3.78 2.49
```

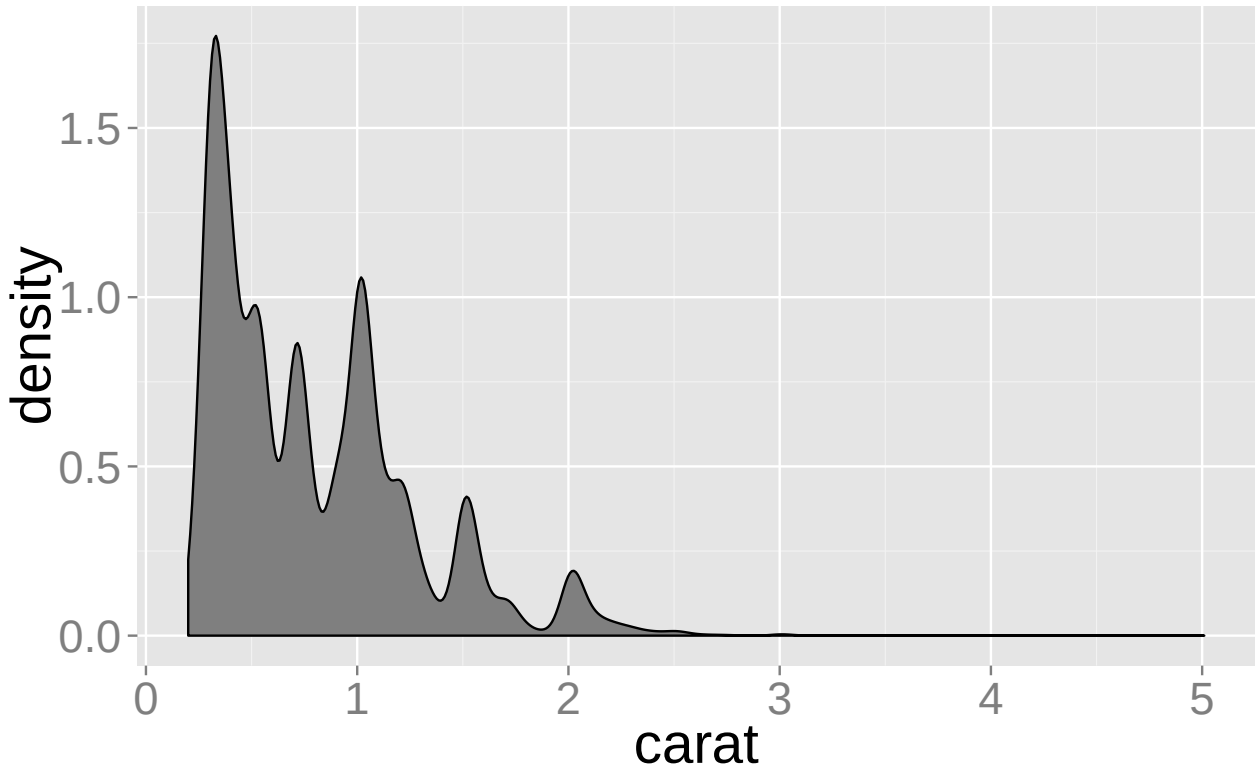
Histogram of carat

```
library(ggplot2)
ggplot(data=diamonds) + geom_histogram(aes(x=carat), binwidth=1/5) +
    theme(text = element_text(size = 25))
```



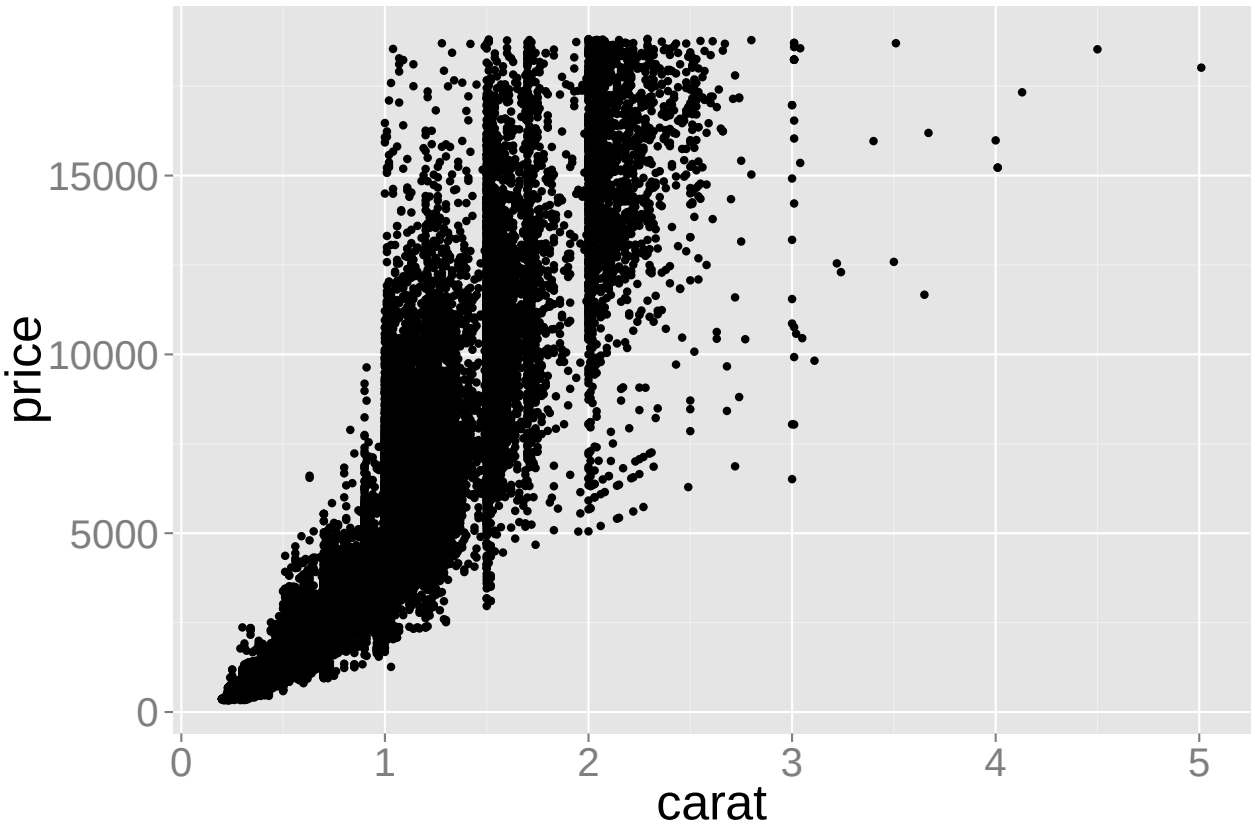
Density plot of carat

```
ggplot(data=diamonds) +
geom_density(aes(x=carat),fill="gray50")+
    theme(text = element_text(size = 25))
```



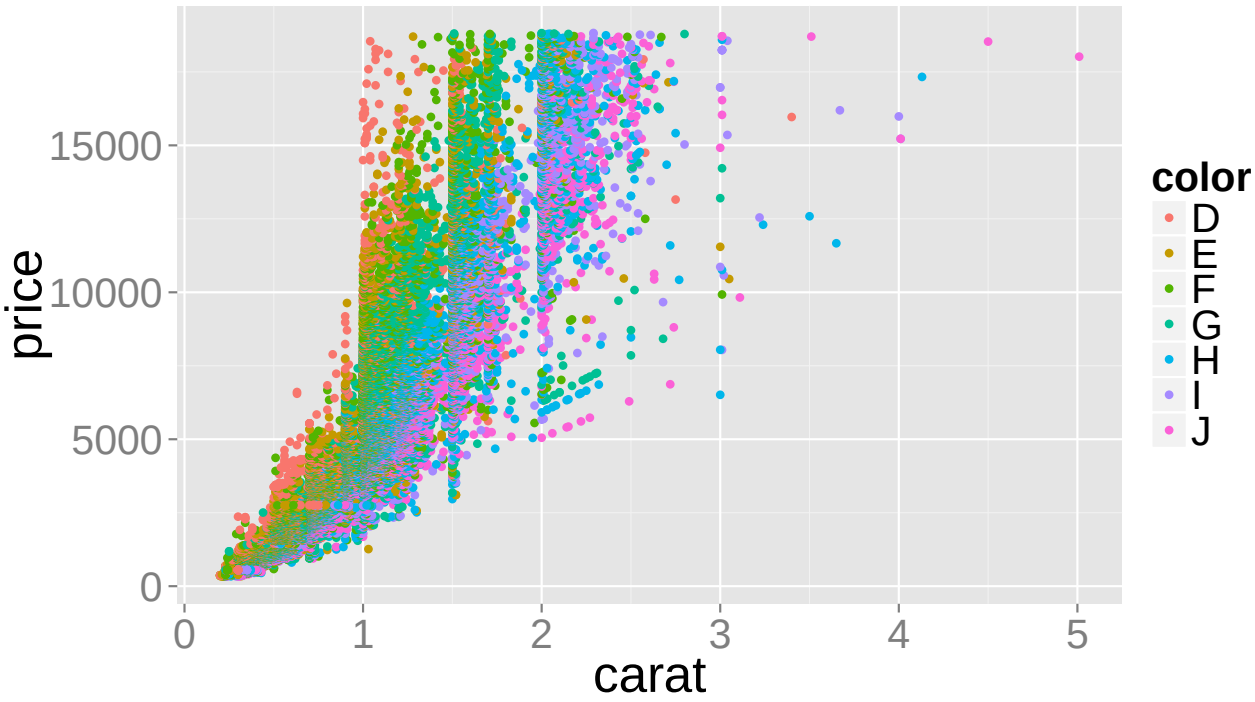
Scatter plots (carat vs. price)

```
ggplot(diamonds, aes(x=carat,y=price)) + geom_point() +
    theme(text = element_text(size = 25))
```



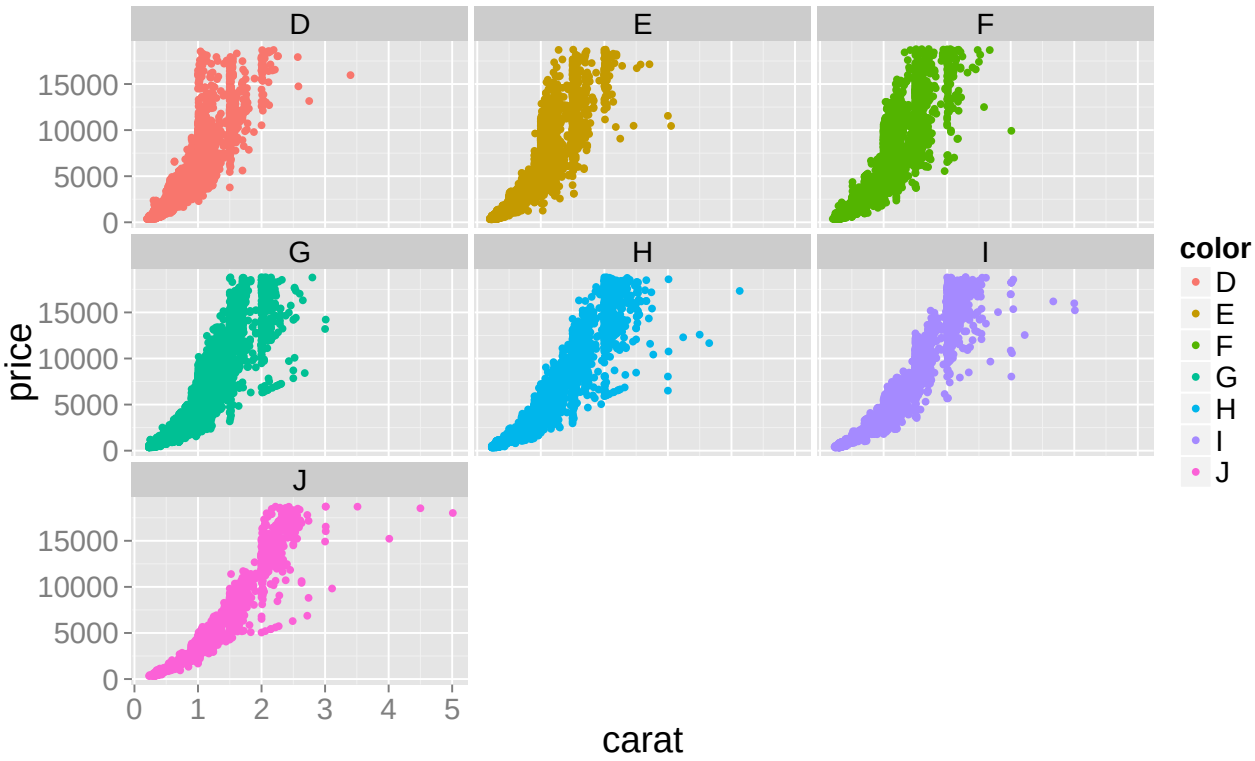
Carat with colors

```
g = ggplot(diamonds, aes(x=carat, y=price)) # saving first layer as variable
g + geom_point(aes(color=color)) + theme(text = element_text(size = 25)) # rendering first layer and ad
```

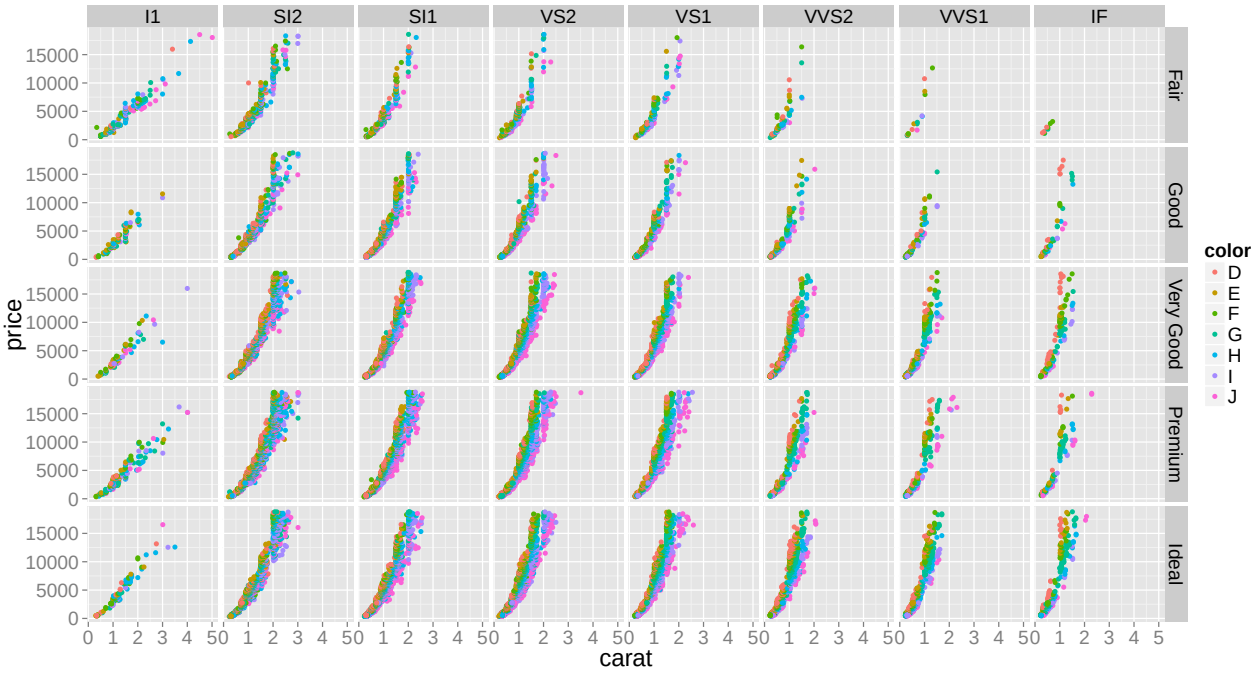


Carat with colors (more details)

```
g + geom_point(aes(color=color)) + facet_wrap(~color) + theme(text = element_text(size = 20))
```



Let's consider cut and clarity



Your trun!

What is your knowledge of diamond's price after exploring this data?