Titanic Data Analysis

Ahmed Ashraf Mohamed   Omar Gamal Abdelaziz   Ahmed Yousri   Ahmed Dawood
Alia Medhat

Contents

1 Introdcution

In this project we are trying first to explore our data to get a better understanding To answer our questions we first need to have a preliminary look at our data, so that we can get a better a idea what we are dealing with, as well as the possible missing data and relationships that exist ## Preliminary Look at the data

We need first to define the data we have.

Variable	Definition	Key
survival	Survival	0 = No, 1 = yes
pclass	ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	sex	
age	Age in year	
sibsp	Number of siblings/spouses aboard the titanic	
parch	Number of parents/children aboard the Titanic	
ticket	ticket number(unique)	

Variable	Definition	Key
fare	Passenger fare	
cabin	Cabin number	
embarked	port of embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

1.1 Loading the packages and the Data

```
# Loading Packages
## tidyverse loads dplyr and readr
library(tidyverse)

## To have different color maps
library(viridis)

## ggplot2 to produce different plots
library(ggplot2)

## uses ggplot2 to produce a correlation matrix -- the data must be in the correct form
library(ggcorrplot)

## Gives us better themes
library(hrbrthemes)

## to use skewness fun. to calculate skewness of the distribution
library(e1071)

## Multivariate imputation using chained equations -- to impute the missing values in our data
library(mice)

## Loads different statistical functions
library(statsr)

## To produce interactive plot
library(plotly)


# Loading Training Data
train <- read_csv("data/train.csv")

# Loading Testing Data
test <- read_csv("data/test.csv")

# Binding them into a full data frame
df <- bind_rows(train,test)
```

2 Exploration Of The Data

2.1 Description of the Data

2.2 Summary of Data

```
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
##  Length:891         Min.   : 0.42   Min.   :0.000   Min.   :0.0000
##  Class :character   1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000
##  Mode  :character   Median :28.00   Median :0.000   Median :0.0000
##                     Mean   :29.70   Mean   :0.523   Mean   :0.3816
##                     3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000
##                     Max.   :80.00   Max.   :8.000   Max.   :6.0000
##                     NA's   :177
##     Ticket               Fare            Cabin             Embarked
##  Length:891         Min.   :  0.00   Length:891         Length:891
##  Class :character   1st Qu.:  7.91   Class :character   Class :character
##  Mode  :character   Median : 14.45   Mode  :character   Mode  :character
##                     Mean   : 32.20
##                     3rd Qu.: 31.00
##                     Max.   :512.33
##
```

2.3 Categories of Features

Quantitative data are measures of values or counts and are expressed as numbers.

Qualitative data are measures of 'types' and may be represented by a name, symbol, or a number code.

2.3.1 Qualitive

Categorical: Survived, Sex, and Embarked. Ordinal: Pclass. Nominal: Name.

2.3.2 Quantitive

Continuous: Age, Fare. Discrete: SibSp, Parch.

2.3.3 Mix types

Ticket is a mix of numeric and alphanumeric data types Cabin is mix between alpha and numeric

2.4 Exploring Missing Data.

Checking for Missing values in each feature

```
colSums(is.na(train))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         177
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0         687           2
```

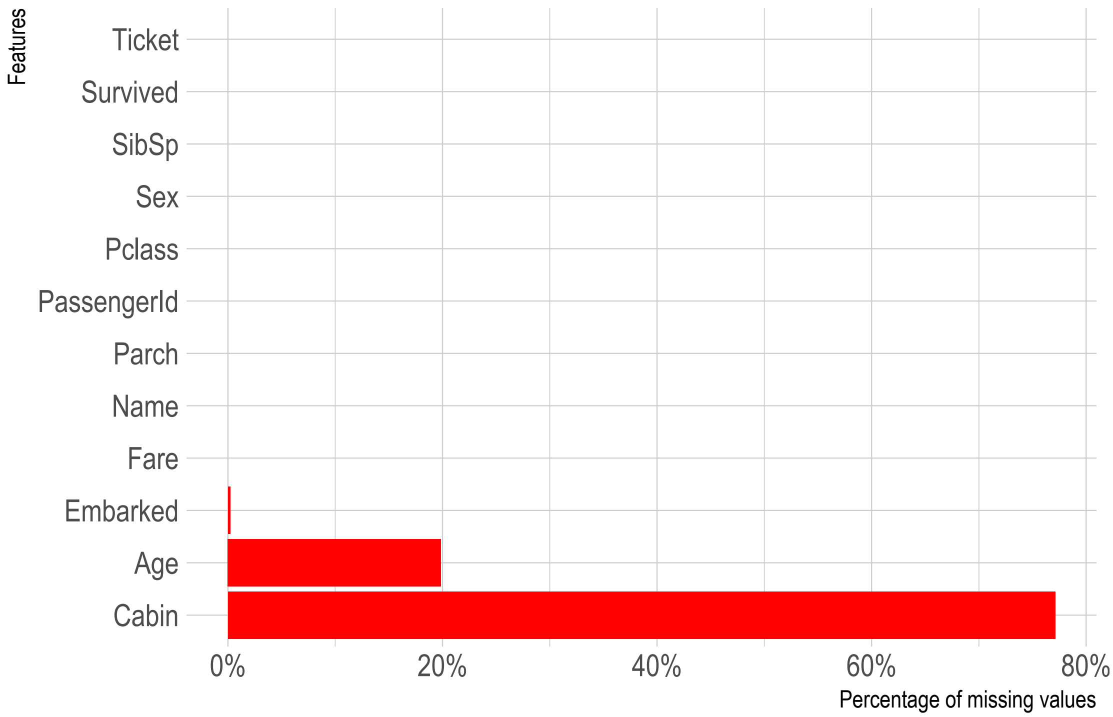
```
missing_values <- train %>% summarize_all(funs(sum(is.na(.))/n()))
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## i Please use a list of either functions or lambdas:
##
## # Simple named list: list(mean = mean, median = median)
##
## # Auto named with `tibble::lst()`: tibble::lst(mean, median)
##
## # Using lambdas list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
```

```
missing_values <- gather(missing_values, key="feature", value="missing_pct")

missing_values
```

```
## # A tibble: 12 x 2
##    feature     missing_pct
##    <chr>             <dbl>
##  1 PassengerId     0
##  2 Survived        0
##  3 Pclass          0
##  4 Name            0
##  5 Sex             0
##  6 Age             0.199
##  7 SibSp           0
##  8 Parch           0
##  9 Ticket          0
## 10 Fare            0
## 11 Cabin           0.771
## 12 Embarked        0.00224
```

```
missing_values %>%
  ggplot(aes(x=reorder(feature,-missing_pct),y=missing_pct)) +
  geom_bar(stat="identity",fill="red") +
  coord_flip() + # to flip the graph
  xlab("Features") +
  ylab("Percentage of missing values")+
  scale_y_continuous(labels=scales::percent) +
  theme_ipsum()
```



2.4.1 Missing data, it is normal?

It is quite normal to see missing data in any data-set as such data is collected by manually which means that there might be some error. Missing data present various problems. First, the absence of data reduces statistical power, which refers to the probability that the test will reject the null hypothesis when it is false. Second, the lost data can cause bias in the estimation of parameters. Third, it can reduce the representativeness of the samples.

2.4.2 The solution to the missing data

2.4.2.1 Removal The removal of the observations that contain missing might cause a bigger issue where it might an even bigger loss of information which cause bias in our estimation. But on close observation of our data we can identify that there are some features that are not extremely useful and contain missing data, so we can drop such features.

```
train <- train %>%
  select(!Cabin)
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       
##  Length:891         Min.   : 0.42   Min.   :0.000   Min.   :0.0000  
##  Class :character   1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000  
##  Mode  :character   Median :28.00   Median :0.000   Median :0.0000  
##                     Mean   :29.70   Mean   :0.523   Mean   :0.3816  
##                     3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000  
##                     Max.   :80.00   Max.   :8.000   Max.   :6.0000  
```

```
##                     NA's   :177
##     Ticket               Fare           Embarked
##  Length:891         Min.   :  0.00   Length:891
##  Class :character   1st Qu.:  7.91   Class :character
##  Mode  :character   Median : 14.45   Mode  :character
##                     Mean   : 32.20
##                     3rd Qu.: 31.00
##                     Max.   :512.33
##
```

2.4.2.2 Imputation Imputing the missing data gives the advantage that we use a learning model to predict such missing data and maintain the distribution of our data.

2.5 Histogram of Age feature.

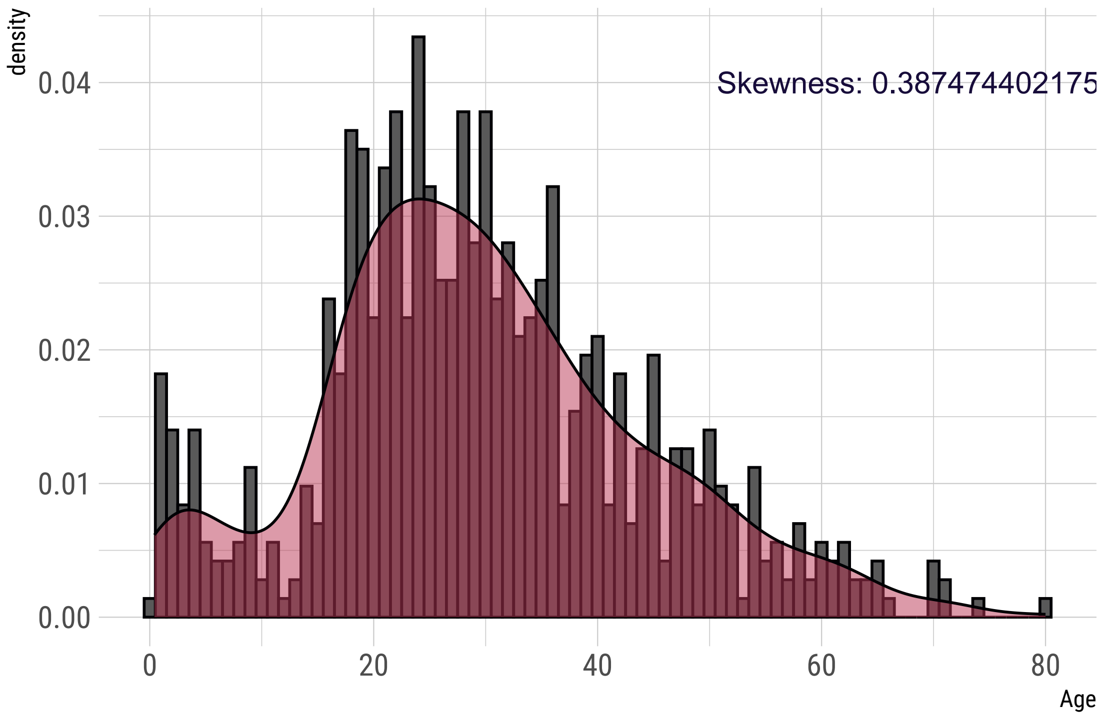
```
gAgeDensity <- train %>%
  select(Age) %>%
  ggplot(aes(Age, y = ..density..)) +
  geom_histogram(bins = 20,binwidth = 1,color=inferno(1,alpha=1)) +
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = inferno(1,begin=0)) +
  annotate(
    "text",
    x = 70,
    y = 0.04,
    label = paste("Skewness:",skewness(train$Age,na.rm = T)),
    colour = inferno(1,begin = 0.1),
    size = 4
  ) +
  theme_ipsum_rc()
```

```
gAgeDensity
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
```

```
## Warning: Removed 177 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 177 rows containing non-finite values (`stat_density()`).
```



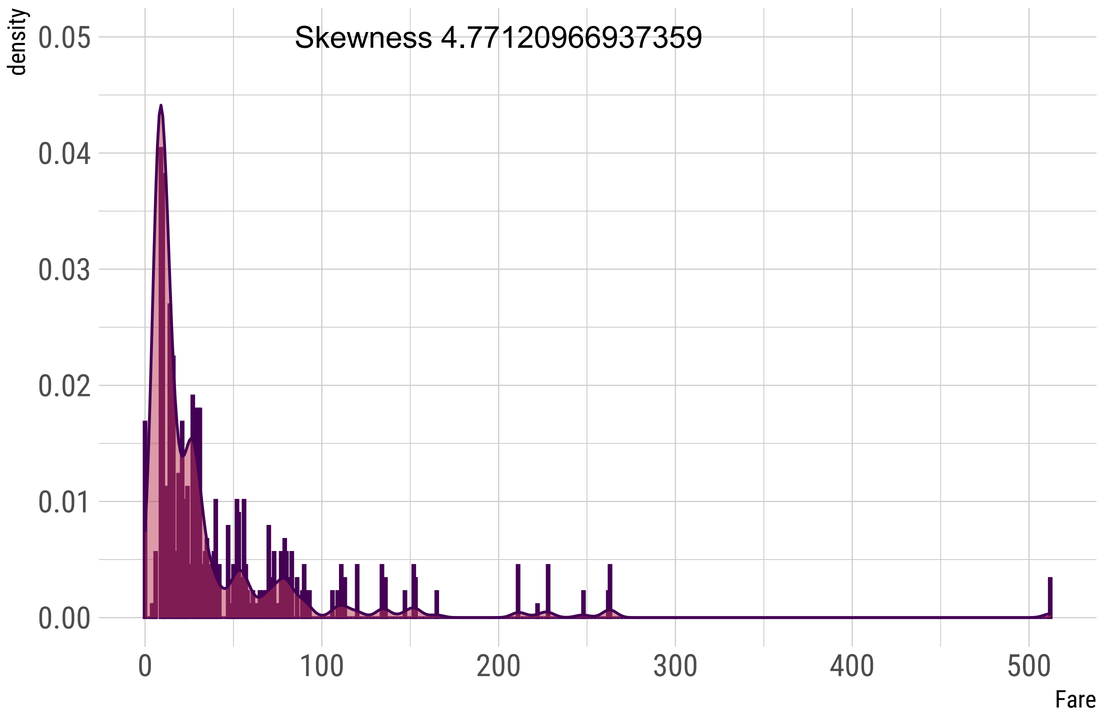

```
train %>%
  summarise(Age_mean = mean(Age,na.rm = T), Age_sd = sd(Age,na.rm = T))
```

```
## # A tibble: 1 x 2
##   Age_mean Age_sd
##      <dbl>  <dbl>
## 1     29.7   14.5
```

2.6 Histogram of Fare feature

```
gFareDensity <- train %>%
  select(Fare) %>%
  ggplot(aes(Fare, y = ..density..)) +
  geom_histogram(bins = 20,binwidth = 1,color=viridis(1,alpha=1)) +
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = viridis(1,begin=0)) +
  scale_y_continuous(limits = c(0,0.05))+
  theme_ipsum_rc() +
  annotate(
    "text",
    x = 200,
    y = 0.05,
    label = paste("Skewness",skewness(train$Fare)),
    colour = "black",
    size = 4
  )

gFareDensity
```

```
## Warning: Removed 4 rows containing missing values (`geom_bar()`).
```




```
train %>%
  summarise(Fare_mean = mean(Fare,na.rm = T), Fare_sd = sd(Fare,na.rm = T))
```

```
## # A tibble: 1 x 2
##   Fare_mean Fare_sd
##       <dbl>   <dbl>
## 1      32.2    49.7
```

2.7 Imputing the missing age feature

```
#---------------MICE--------------
set.seed(129)
mice_mod <- mice(train[,!names(train) %in% c('PassengerId','Name','Ticket','Cabin','Survived')],method =
```

```
##
##  iter imp variable
##   1   1  Age
##   1   2  Age
##   1   3  Age
##   1   4  Age
##   1   5  Age
##   2   1  Age
##   2   2  Age
##   2   3  Age
##   2   4  Age
##   2   5  Age
##   3   1  Age
##   3   2  Age
##   3   3  Age
##   3   4  Age
##   3   5  Age
##   4   1  Age
```

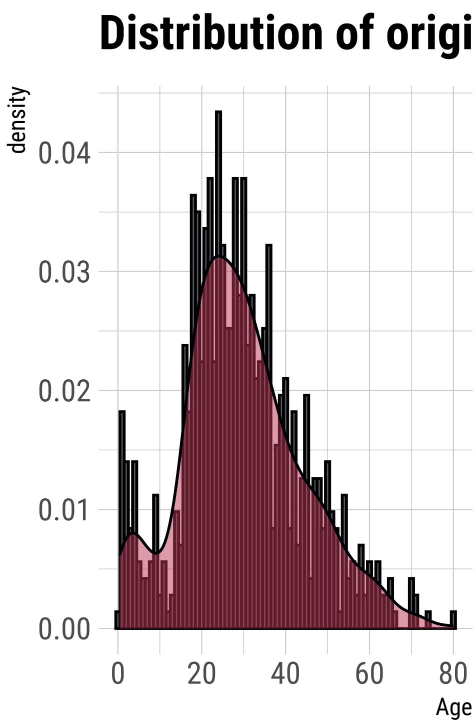
```
##   4   2  Age
##   4   3  Age
##   4   4  Age
##   4   5  Age
##   5   1  Age
##   5   2  Age
##   5   3  Age
##   5   4  Age
##   5   5  Age
```

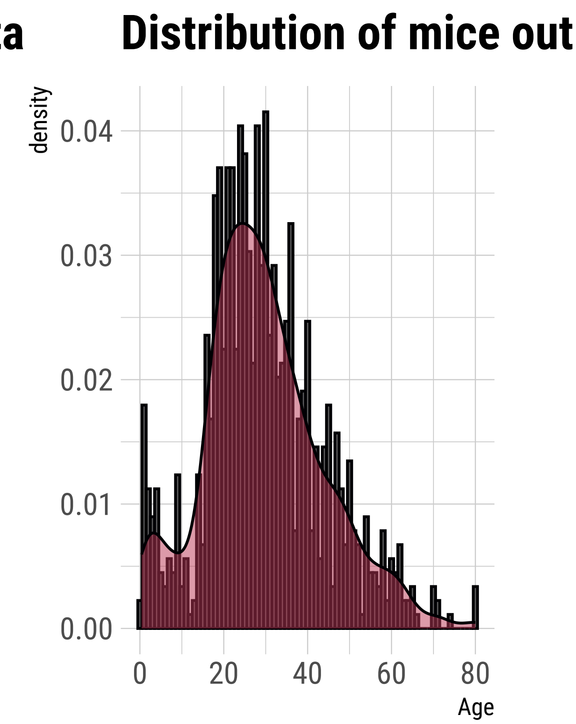
```
## Warning: Number of logged events: 2
```

```
mice_output <- complete(mice_mod)

gdistrOriginalData <- train %>%
  select(Age) %>%
  ggplot(aes(Age, y = ..density..)) +
  geom_histogram(bins = 25,binwidth = 1,color=inferno(1,alpha=1)) +
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = inferno(1,begin=0)) +
  ggtitle("Distribution of original data") +
  theme_ipsum_rc()
gdistrMICEData <- mice_output %>%
  select(Age) %>%
  ggplot(aes(Age, y = ..density..)) +
  geom_histogram(bins = 25,binwidth = 1,color=inferno(1,alpha=1)) +
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = inferno(1,begin=0)) +
  ggtitle("Distribution of mice output") +
  theme_ipsum_rc()

gridExtra::grid.arrange(gdistrOriginalData,gdistrMICEData,nrow = 1)
```

```
## Warning: Removed 177 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 177 rows containing non-finite values (`stat_density()`).
```

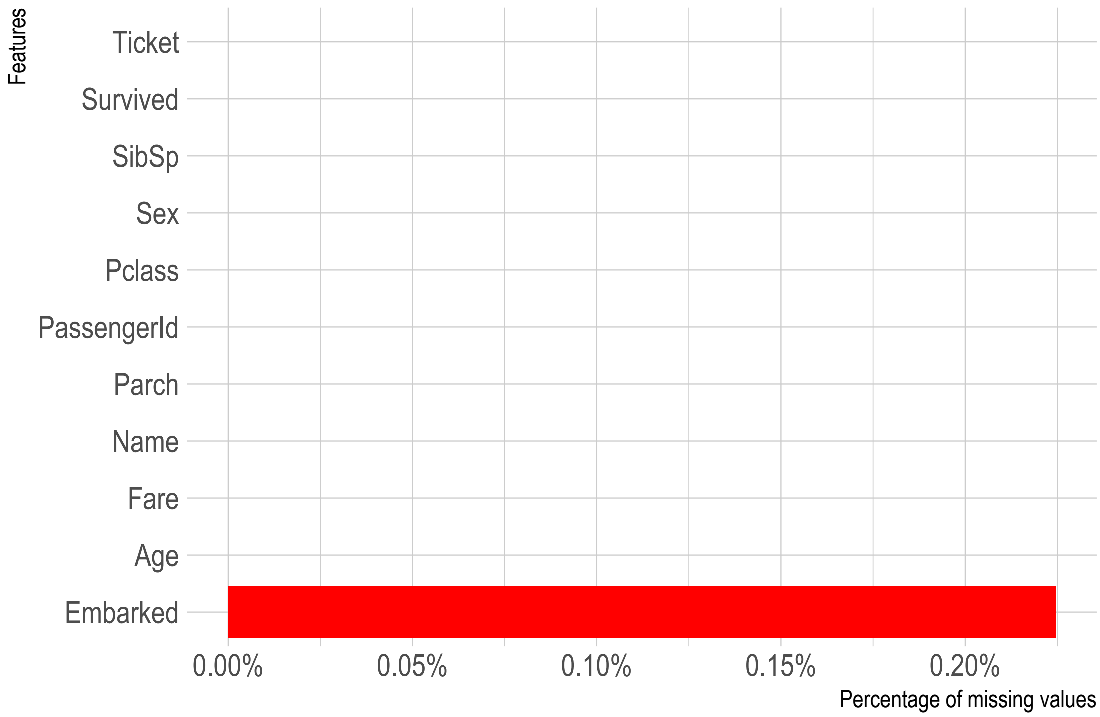





```
train$Age <- mice_output$Age

missing_values <- train %>% summarize_all(funs(sum(is.na(.))/n()))

missing_values <- gather(missing_values, key="feature", value="missing_pct")

missing_values
```

```
## # A tibble: 11 x 2
##    feature     missing_pct
##    <chr>             <dbl>
##  1 PassengerId     0
##  2 Survived        0
##  3 Pclass          0
##  4 Name            0
##  5 Sex             0
##  6 Age             0
##  7 SibSp           0
##  8 Parch           0
##  9 Ticket          0
## 10 Fare            0
## 11 Embarked        0.00224
```

```
missing_values %>%
  ggplot(aes(x=reorder(feature,-missing_pct),y=missing_pct)) +
  geom_bar(stat="identity",fill="red") +
  coord_flip() + # to flip the graph
  xlab("Features") +
  ylab("Percentage of missing values")+
  scale_y_continuous(labels=scales::percent) +
  theme_ipsum()
```




```
train %>%
  summarise(Age_mean = mean(Age,na.rm = T), Age_sd = sd(Age,na.rm = T))
```

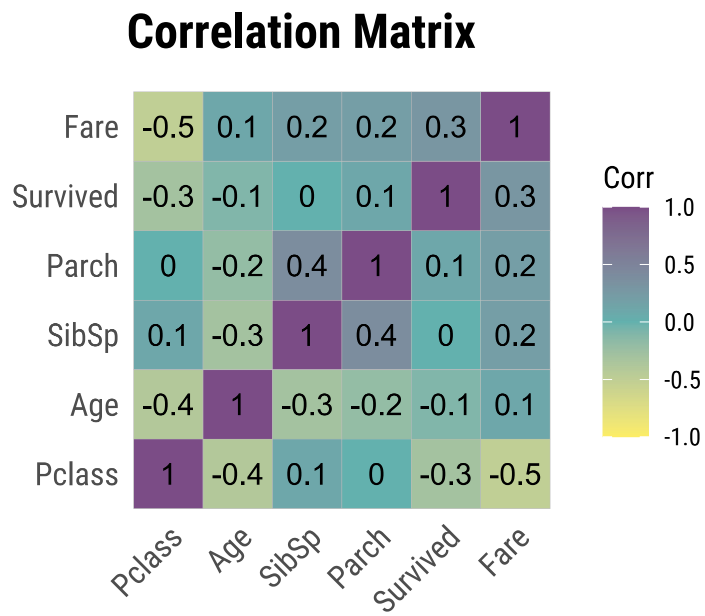
```
## # A tibble: 1 x 2
##   Age_mean Age_sd
##      <dbl>  <dbl>
## 1     29.6   14.3
```

2.8 Plotting The Data

2.8.1 Correlation Matrix

We are going to use correlation matrix of the numerical data to assess the correlation, which might gives a better idea of which feature might be important

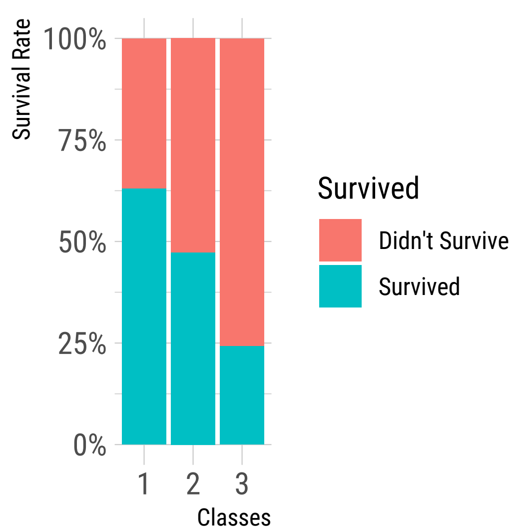
```
correlationMatrix <- train %>%
  filter(!is.na(Age)) %>%
  select(Survived, Pclass,Age,SibSp,Parch,Fare) %>%
  cor() %>%
  ggcorrplot(lab = T,
             ggtheme =theme_ipsum_rc(grid = F),
             title="Correlation Matrix",hc.order=T,
             colors =rev(viridis(3,alpha=0.7)),
             digits = 1)

correlationMatrix
```



The fare features seems to be the most correlated feature to survival of the passengers, but it doesn't negate the importance of the other features in the data. Which means that we will start by comparing the each that we consider to be important against survival feature

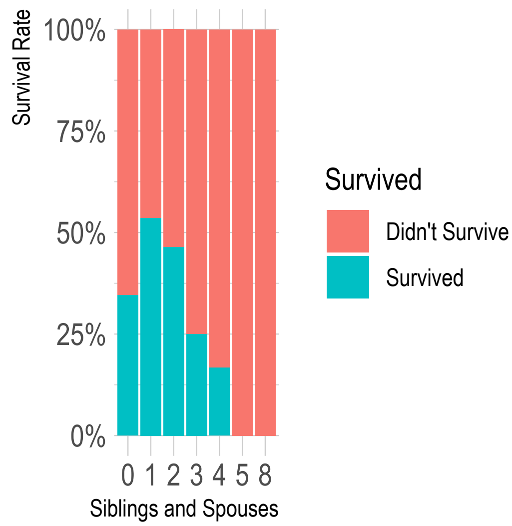
2.8.2 Class of Passenger Vs Survived

```
gPclassSurvived <- train %>%
  select(Pclass,Survived) %>%
  ggplot(aes(as_factor(Pclass),fill=as_factor(Survived))) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels=scales::percent) +
  theme_ipsum_rc() +
  labs(x = "Classes",y = "Survival Rate")+
  scale_fill_discrete(name = "Survived", labels = c("Didn't Survive","Survived"))

gPclassSurvived
```



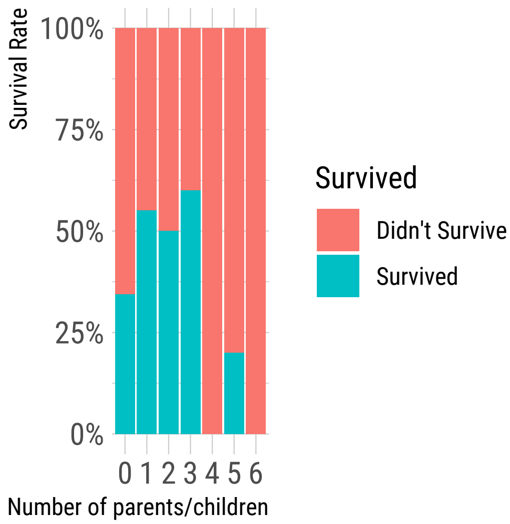
2.8.3 Siblings and Spouses Vs Survived

```
gSibSpSurvived <- train %>%
  select(SibSp,Survived) %>%
  ggplot(aes(as_factor(SibSp),fill=as_factor(Survived))) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Siblings and Spouses",y = "Survival Rate")+
  scale_fill_discrete(name = "Survived", labels = c("Didn't Survive","Survived")) +
  theme_ipsum()

gSibSpSurvived
```



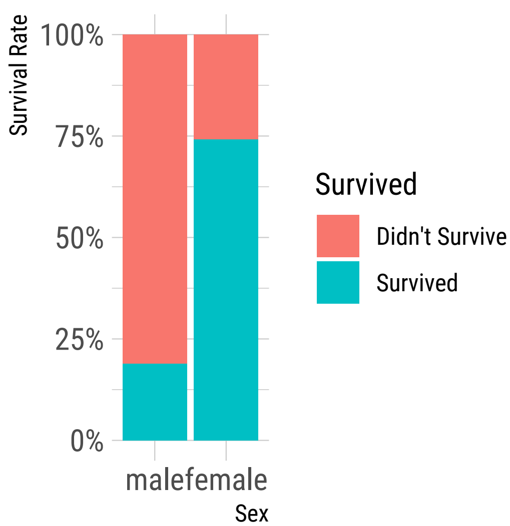
2.8.4 Number of children/parents Vs Survived

```
gParchSurvived <- train %>%
  select(Parch,Survived) %>%
  ggplot(aes(as_factor(Parch),fill=as_factor(Survived))) +
  geom_bar(position = "fill") +
  scale_y_continuous(label = scales::percent)+
  labs(x = "Number of parents/children",y = "Survival Rate")+
  scale_fill_discrete(name = "Survived", labels = c("Didn't Survive","Survived")) +
  theme_ipsum_rc()
gParchSurvived
```



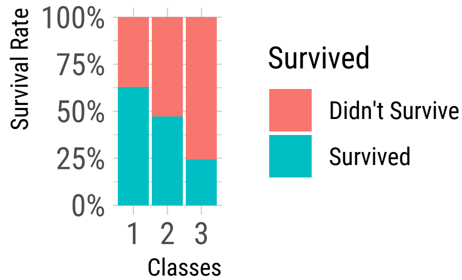
2.8.5 Gender VS Survived

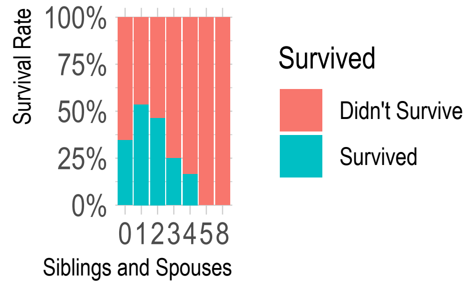
```
gSexSurvived <- train %>%
  select(Sex,Survived) %>%
  ggplot(aes(as_factor(Sex),fill = as_factor(Survived))) +
  geom_bar(position = "fill") +
  scale_y_continuous(label = scales::percent) +
  labs(x = "Sex",y = "Survival Rate")+
  scale_fill_discrete(name = "Survived", labels = c("Didn't Survive","Survived")) +
  theme_ipsum_rc()
gSexSurvived
```

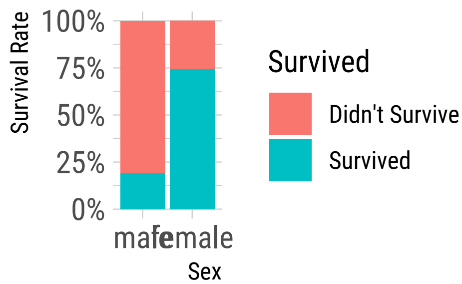


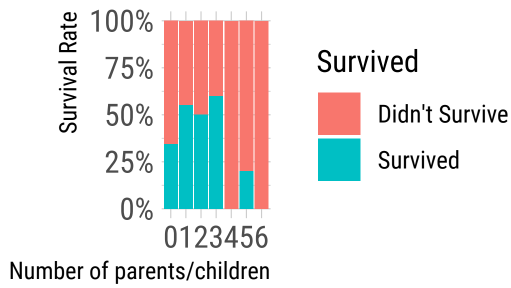
2.8.6 Dashboard of the previous graphs

```
gridExtra::grid.arrange(gPclassSurvived,
                        gSibSpSurvived,
                        gSexSurvived,
                        gParchSurvived,
                        nrow=2)
```










```
train %>%
  group_by(Sex) %>%
  summarise(Age_mean = mean(Age,na.rm=TRUE),
            age_sd = sd(Age,na.rm=T),
            surival_mean = mean(Survived,na.rm =T),
            surival_sd = sd(Survived,na.rm = T))
```

```
## # A tibble: 2 x 5
##   Sex    Age_mean age_sd surival_mean surival_sd
##   <chr>     <dbl>  <dbl>        <dbl>      <dbl>
## 1 female     27.5   14.0        0.742      0.438
## 2 male       30.7   14.4        0.189      0.392
```

3 kernel distribution

A kernel distribution is a nonparametric representation of the probability density function (pdf) of a random variable in any population

The kernel smoothing function defines the shape of the curve used to generate the pdf Kernel distribution is Quote from histogram in other word (smooth representation of a histogram) That the integral =1 There is a benefit of smooth representation of a histogram like Ignores irregularities and outliers , more efficient in approximation so it deals better with large data than small data

$$\hat{f}_h = \frac{1}{n} = \sum_{i=1}^{n} K(x - x_i) = \frac{1}{nh} K\left(\frac{x - x_i}{h}\right)$$

3.1 Rules

3.1.1 Non-weighted Data

$$\hat{f}_h = \frac{1}{n} \sum_{i=1}^{n} K(x - x_i) = \frac{1}{nh} K\left(\frac{x - x_i}{h}\right)$$

3.1.2 Weighted Data

$$\hat{f}_h = \frac{1}{h}\sum_{i=1}^{N} w_i K\left(\frac{x - x_i}{h}\right), \qquad \text{where} \sum_{i=1}^{N} w_i = 1$$

3.2 Kernel Function

1. Box
2. Triangle
3. Normal
4. Pantechnicon

Each density curve uses the same input data, but applies a different kernel smoothing function to generate the pdf. The density estimates are roughly comparable, but the shape of each curve varies slightly. For example, the box kernel produces a density curve that is less smooth than the others.

The choice of bandwidth value controls the smoothness of the resulting probability density curve (higher value of h more smoothing)

Specifying a smaller bandwidth produces a very rough curve, but reveals that there might be two major peaks in the data. Specifying a larger bandwidth produces a curve nearly identical to the kernel function Choosing the optimal (h) bandwidth methods :

1. Silverman's rule of thump that computes an optimal h by assuming that data is normally distributed
2. Improved Sheather Jones (ISJ) an algorithm is more robust with multimodality data or a lot of data (one disadvantage is it needs to large data)

Bounded domains data: have a constrains like data couldn't be negative (-ve lead to probability = 0)

3.3 Mirror method

1. Mirror the data
2. Sum the original and mirrored kernel density estimate
3. Chop it so that zero at the boundary side

3.4 2D

h : could be matrix (different h in different directions)

The choice of norm comes into $d \geq 2$

The p-norm is $||x||_p := (\sum_{i=1} |x|^p)^{\frac{1}{p}}$ - norm-p =1 Manhattan distance - norm-p =2 Euclidean norm - norm-p =inf maximum norm (it's not obvious in every case which norm is the correct one)

standard euclidean distance is good choice because it invariant under rotation as large data choice of k and p isn't important so

4 Answering our Questions

4.1 Acurrate summary of our data.

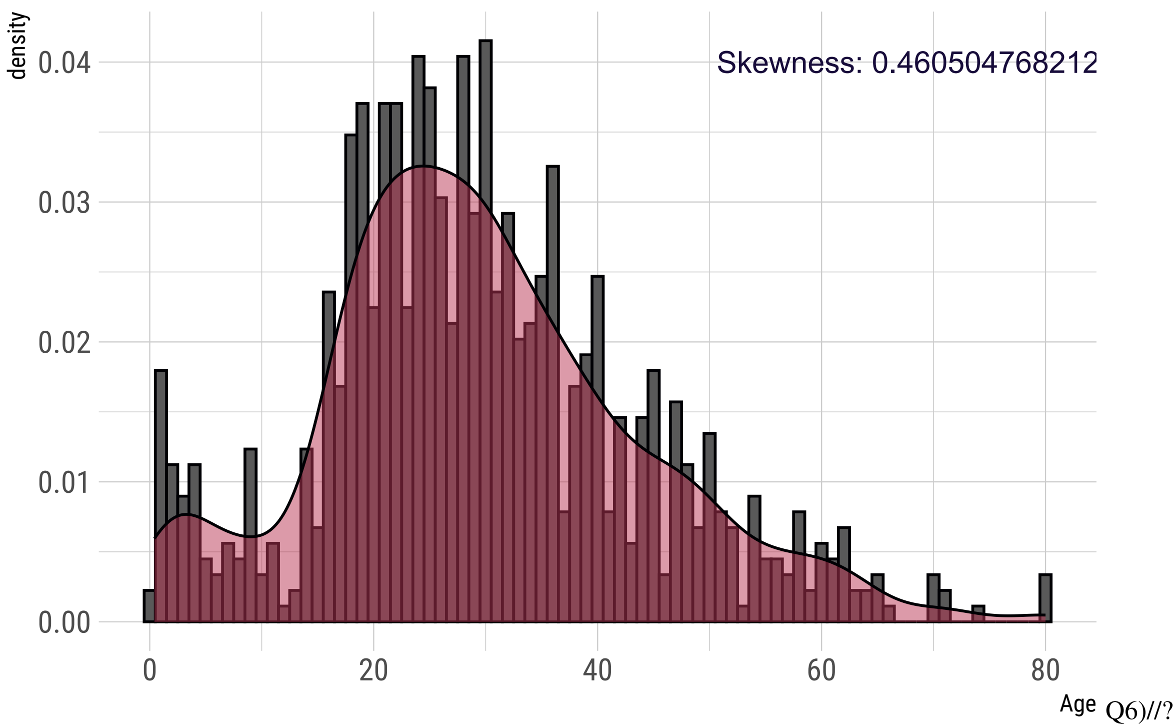
Q3) i)

ii)?? iii) Yes, by interval estimation iv)?? Q4) Q5)

```
gAgeDensity <- train %>%
  select(Age) %>%
  ggplot(aes(Age, y = ..density..)) +
  geom_histogram(bins = 20,binwidth = 1,color=inferno(1,alpha=1)) +
```

```
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = inferno(1,begin=0)) +
  annotate(
    "text",
    x = 70,
    y = 0.04,
    label = paste("Skewness:",skewness(train$Age,na.rm = T)),
    colour = inferno(1,begin = 0.1),
    size = 4
  ) +
  theme_ipsum_rc()
```

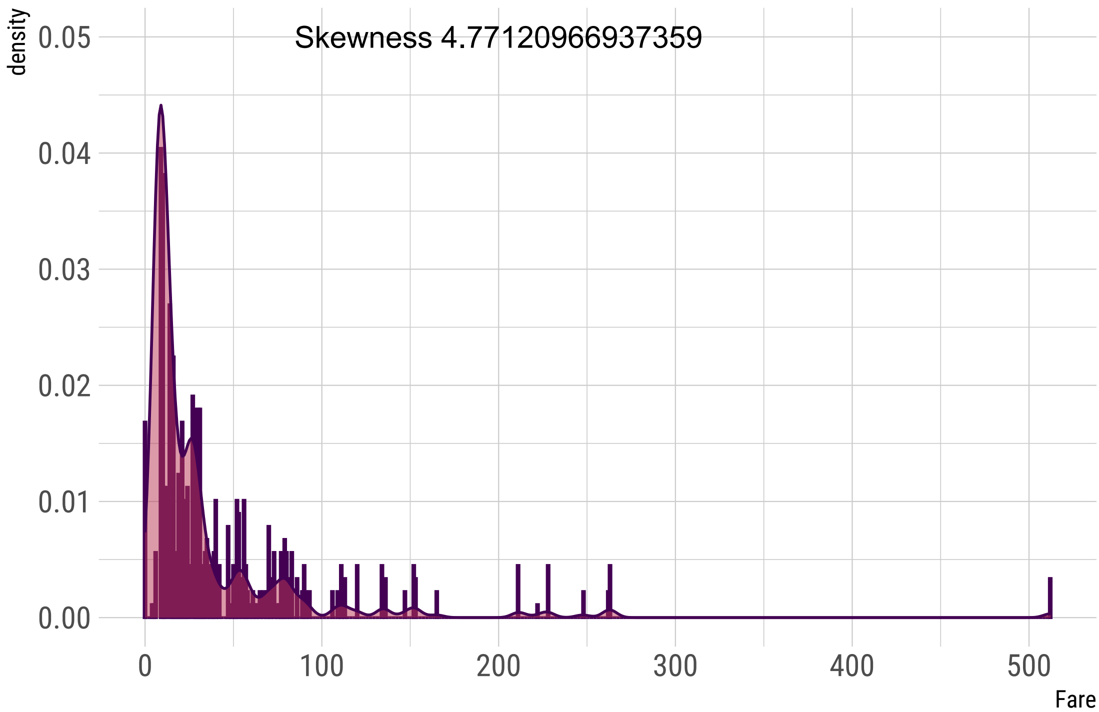
```
gAgeDensity
```




```
gFareDensity <- train %>%
  select(Fare) %>%
  ggplot(aes(Fare, y = ..density..)) +
  geom_histogram(bins = 20,binwidth = 1,color=viridis(1,alpha=1)) +
  geom_density(fill=inferno(1,begin = 0.5,alpha = 0.5),color = viridis(1,begin=0)) +
  scale_y_continuous(limits = c(0,0.05))+
  theme_ipsum_rc() +
  annotate(
    "text",
    x = 200,
    y = 0.05,
    label = paste("Skewness",skewness(train$Fare)),
    colour = "black",
    size = 4
  )
```

```
gFareDensity
```

```
## Warning: Removed 4 rows containing missing values (`geom_bar()`).
```



Q8) Q)9,10,11

Q12) While no. of sample size increase the variability of sampling distribution decrease and the mean increase. Q)13,14,15

Q16)While no. of sample size increase the variability of sampling distribution decrease,and The sample distribution mean will be normally distributed as long as the sample size is more than 30. Q17,18)