TITANIC

Meet

2023-03-28

Working on data

Loading required library to work upon data in R

```
library(tidyverse)
library(ggplot2)
library(randomForest)
```

Importing data from the train model of Titanic dataset

```
train_df <- read_csv("train.csv", show_col_types = FALSE)
test_df <- read_csv("test.csv", show_col_types = FALSE)
```

Now that we have imported the datasets, lets review them and observe some insights.

Review the dataset

```
head(train_df)
```

```
## # A tibble: 6 × 12
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1           1        0      3 Braund… male     22     1     0 A/5 2…  7.25
<NA>
## 2           2        1      1 Cuming… fema…    38     1     0 PC 17… 71.3
C85
## 3           3        1      3 Heikki… fema…    26     0     0 STON/…  7.92
<NA>
## 4           4        1      1 Futrel… fema…    35     1     0 113803 53.1
C123
## 5           5        0      3 Allen,… male     35     0     0 373450  8.05
<NA>
## 6           6        0      3 Moran,… male     NA     0     0 330877  8.46
<NA>
## # … with 1 more variable: Embarked <chr>
```

```
glimpse(train_df)
```

```
## Rows: 891
## Columns: 12
## $ PassengerId <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17,…
```

```
## $ Survived    <dbl> 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1,
0, 1…
## $ Pclass      <dbl> 3, 1, 3, 1, 3, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3, 2,
3, 3…
## $ Name        <chr> "Braund, Mr. Owen Harris", "Cumings, Mrs. John Bradley
(Fl…
## $ Sex         <chr> "male", "female", "female", "female", "male", "male",
"mal…
## $ Age         <dbl> 22, 38, 26, 35, 35, NA, 54, 2, 27, 14, 4, 58, 20, 39,
14, …
## $ SibSp       <dbl> 1, 1, 0, 1, 0, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4, 0,
1, 0…
## $ Parch       <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1, 0,
0, 0…
## $ Ticket      <chr> "A/5 21171", "PC 17599", "STON/O2. 3101282", "113803",
"37…
## $ Fare        <dbl> 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 8.4583,
51.8625,…
## $ Cabin       <chr> NA, "C85", NA, "C123", NA, NA, "E46", NA, NA, NA,
"G6", "C…
## $ Embarked    <chr> "S", "C", "S", "S", "S", "Q", "S", "S", "S", "C", "S",
"S"…
```

```
str(train_df)
```

```
## spc_tbl_ [891 × 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ PassengerId: num [1:891] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : num [1:891] 0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : num [1:891] 3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr [1:891] "Braund, Mr. Owen Harris" "Cumings, Mrs. John
Bradley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs.
Jacques Heath (Lily May Peel)" ...
##  $ Sex        : chr [1:891] "male" "female" "female" "female" ...
##  $ Age        : num [1:891] 22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : num [1:891] 1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : num [1:891] 0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr [1:891] "A/5 21171" "PC 17599" "STON/O2. 3101282"
"113803" ...
##  $ Fare       : num [1:891] 7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr [1:891] NA "C85" NA "C123" ...
##  $ Embarked   : chr [1:891] "S" "C" "S" "S" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   PassengerId = col_double(),
##   ..   Survived = col_double(),
##   ..   Pclass = col_double(),
##   ..   Name = col_character(),
##   ..   Sex = col_character(),
##   ..   Age = col_double(),
##   ..   SibSp = col_double(),
```

```
##   ..   Parch = col_double(),
##   ..   Ticket = col_character(),
##   ..   Fare = col_double(),
##   ..   Cabin = col_character(),
##   ..   Embarked = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>

summary(train_df)

##   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
##

summary(test_df)

##   PassengerId         Pclass          Name               Sex
##  Min.   : 892.0   Min.   :1.000   Length:418         Length:418
##  1st Qu.: 996.2   1st Qu.:1.000   Class :character   Class :character
##  Median :1100.5   Median :3.000   Mode  :character   Mode  :character
##  Mean   :1100.5   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :3.000
##
##       Age            SibSp           Parch           Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   Length:418
##  1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :27.00   Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :30.27   Mean   :0.4474   Mean   :0.3923
##  3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.0000
```

```
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000
##  NA's   :86
##       Fare            Cabin             Embarked
##  Min.   :  0.000   Length:418         Length:418
##  1st Qu.:  7.896   Class :character   Class :character
##  Median : 14.454   Mode  :character   Mode  :character
##  Mean   : 35.627
##  3rd Qu.: 31.500
##  Max.   :512.329
##  NA's   :1
```

We can see in the summary that **177** missing values of Age is there in the *train_df* dataset but Age is an important factor in the survival on *Titanic*.

Cleaning Data

Missing Values

Now lets make the **missing values** and the **space values** if any into **NA** to avoid any future confusions.
We will create a copy of *train_df* and name it *train_df2*

```
train_df2 <- train_df
train_df2[train_df2=="" | train_df2==" "] <- NA
```

Lets do the same for *test_df* dataset.

```
test_df2 <- test_df
test_df2[test_df2=="" | test_df2==" "] <- NA
```

For future convenience, let's combine the dataset but the main problem is that the *test* dataset don't contain the column **Survived**.
So we are going to bind the column with values **NA**.

```
test_df2 <- cbind(test_df2, Survived = NA)
```

Now that we have added the column to the dataset, both the datasets contains same columns so now we will bind the rows to combine both data.

```
alldata <- rbind(train_df2, test_df2)
```

Take a look at the summary of the combined data

```
summary(alldata)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3838   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
```

```
##                  NA's   :418
##      Sex                 Age             SibSp            Parch
##  Length:1309        Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  Class :character   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##  Mode  :character   Median :28.00   Median :0.0000   Median :0.000
##                     Mean   :29.88   Mean   :0.4989   Mean   :0.385
##                     3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.000
##                     Max.   :80.00   Max.   :8.0000   Max.   :9.000
##                     NA's   :263
##     Ticket               Fare            Cabin             Embarked
##  Length:1309        Min.   :  0.000   Length:1309        Length:1309
##  Class :character   1st Qu.:  7.896   Class :character   Class :character
##  Mode  :character   Median : 14.454   Mode  :character   Mode  :character
##                     Mean   : 33.295
##                     3rd Qu.: 31.275
##                     Max.   :512.329
##                     NA's   :1
```

Observing Data

Now we observed that the there is one missing value in *Fare* column. So let's observe the profile of the row to fill the data.

```
alldata %>%
  filter(is.na(Fare))
```

```
## # A tibble: 1 × 12
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1        1044       NA      3 Storey… male   60.5     0     0 3701      NA
<NA>
## # … with 1 more variable: Embarked <chr>
```

Assigning suitable value in Fare

```
fare_df <- alldata %>%
  filter(Embarked=="S", Sex=="male", Pclass==3, Age>=55)
head(fare_df)
```

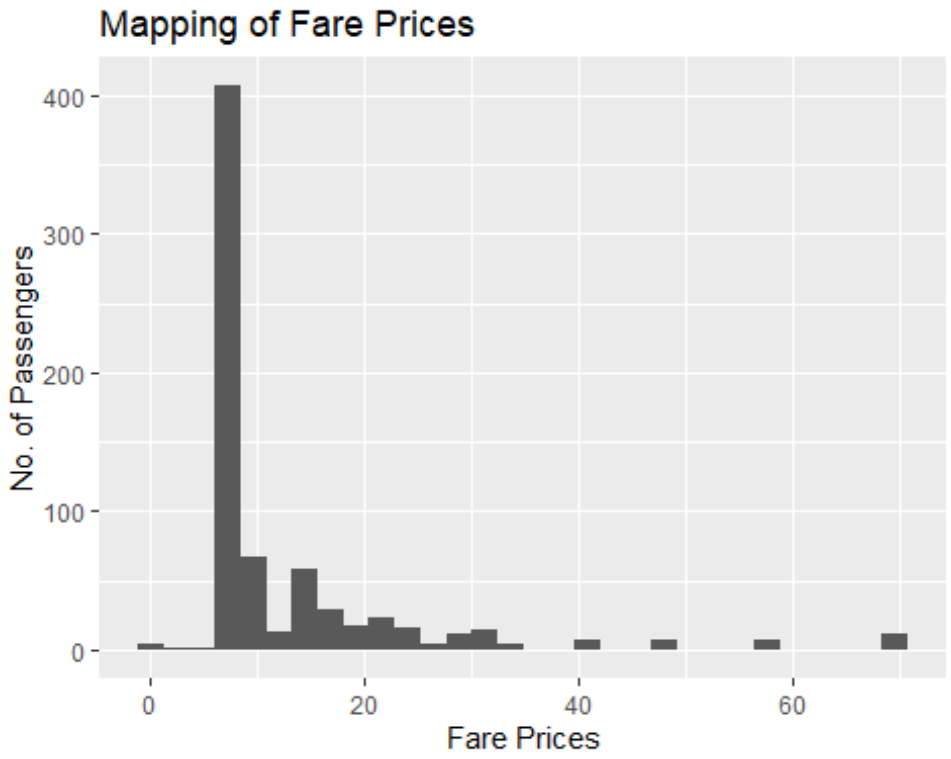
```
## # A tibble: 5 × 12
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1          95        0      3 Coxon,… male   59       0     0 364500  7.25
<NA>
## 2         153        0      3 Meo, M… male   55.5     0     0 A.5. …  8.05
<NA>
## 3         327        0      3 Nysvee… male   61       0     0 345364  6.24
<NA>
```

```
## 4         852        0      3 Svenss… male    74       0     0 347060  7.78
<NA>
## 5        1044       NA      3 Storey… male    60.5     0     0 3701    NA
<NA>
## # … with 1 more variable: Embarked <chr>
```

From the data we can get idea about the **Median** of the data.
Now let's check and verify the data that is there any variation in Fare prices in *Pclass=3*.

```
ggplot(data=alldata %>% filter(Pclass==3))+
  geom_histogram(mapping=aes(x=Fare))+
  labs(x="Fare Prices", y="No. of Passengers", title="Mapping of Fare
Prices")

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



It's around the **median** which is obtained by the filtered data.
Now lets calculate the median and assign the value to **NA**.

```
alldata$Fare[is.na(alldata$Fare)] <- median(fare_df$Fare, na.rm=T)

alldata %>%
  filter(Age==60.5)

## # A tibble: 1 × 12
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
```

```
<chr>
## 1        1044       NA      3 Storey… male   60.5     0     0 3701    7.51
<NA>
## # … with 1 more variable: Embarked <chr>
```

NOTE: *na.rm* is used to remove the missing values from the input vector.

Converting Sex column into numericals

Let's check the **NA** values in the *Sex* column if any.

```
table(is.na(alldata$Sex))

##
## FALSE
##  1309
```

There is no **NA** values.
Now, assign the value **1** for *male* and **0** for *female*.

```
alldata$Sex[alldata$Sex == "male"] <- 1
alldata$Sex[alldata$Sex == "female"] <- 0
head(alldata)

## # A tibble: 6 × 12
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1           1        0      3 Braund… 1        22     1     0 A/5 2…  7.25
<NA>
## 2           2        1      1 Cuming… 0        38     1     0 PC 17… 71.3
C85
## 3           3        1      3 Heikki… 0        26     0     0 STON/…  7.92
<NA>
## 4           4        1      1 Futrel… 0        35     1     0 113803 53.1
C123
## 5           5        0      3 Allen,… 1        35     0     0 373450  8.05
<NA>
## 6           6        0      3 Moran,… 1        NA     0     0 330877  8.46
<NA>
## # … with 1 more variable: Embarked <chr>
```

Name

Separating Title

Lets take a look at the sample of the 30 *Name* column to draw the conclusions about professional title they have.

```
sample(alldata$Name, 30)
```

```
##  [1] "Barkworth, Mr. Algernon Henry Wilson"
##  [2] "Warren, Mr. Frank Manley"
##  [3] "Walcroft, Miss. Nellie"
##  [4] "Cameron, Miss. Clear Annie"
##  [5] "Kirkland, Rev. Charles Leonard"
##  [6] "Davies, Mr. Alfred J"
##  [7] "Hansen, Mr. Claus Peter"
##  [8] "Dean, Mr. Bertram Frank"
##  [9] "Barbara, Mrs. (Catherine David)"
## [10] "Dorking, Mr. Edward Arthur"
## [11] "Colley, Mr. Edward Pomeroy"
## [12] "Larsson-Rondberg, Mr. Edvard A"
## [13] "Moubarek, Mrs. George (Omine Amenia\" Alexander)\""
## [14] "Augustsson, Mr. Albert"
## [15] "Flynn, Mr. John Irwin (\"Irving\")"
## [16] "Shine, Miss. Ellen Natalia"
## [17] "Geiger, Miss. Amalie"
## [18] "Lang, Mr. Fang"
## [19] "Hays, Mrs. Charles Melville (Clara Jennings Gregg)"
## [20] "Holverson, Mrs. Alexander Oskar (Mary Aline Towner)"
## [21] "Morley, Mr. William"
## [22] "Bishop, Mr. Dickinson H"
## [23] "Asplund, Master. Edvin Rojj Felix"
## [24] "Hodges, Mr. Henry Price"
## [25] "Jefferys, Mr. Clifford Thomas"
## [26] "Coelho, Mr. Domingos Fernandeo"
## [27] "Moor, Mrs. (Beila)"
## [28] "Dennis, Mr. William"
## [29] "Davies, Mr. Charles Henry"
## [30] "Nourney, Mr. Alfred (Baron von Drachstedt\")\""
```

Let's separate the *Professional_title* data from *Name* column.

```
alldata <- alldata %>%
  separate(Name, into=c('name2', 'name3'), sep=', ')

alldata <- alldata %>%
  separate(name3, into=c('Professional_title', 'name4'), sep='. ')

## Warning: Expected 2 pieces. Additional pieces discarded in 845 rows [1, 2,
4, 5, 7, 8,
## 9, 10, 11, 13, 14, 15, 16, 18, 19, 21, 23, 24, 25, 26, ...].

alldata <- alldata %>%
  select(-name2,-name4)
```

Mapping

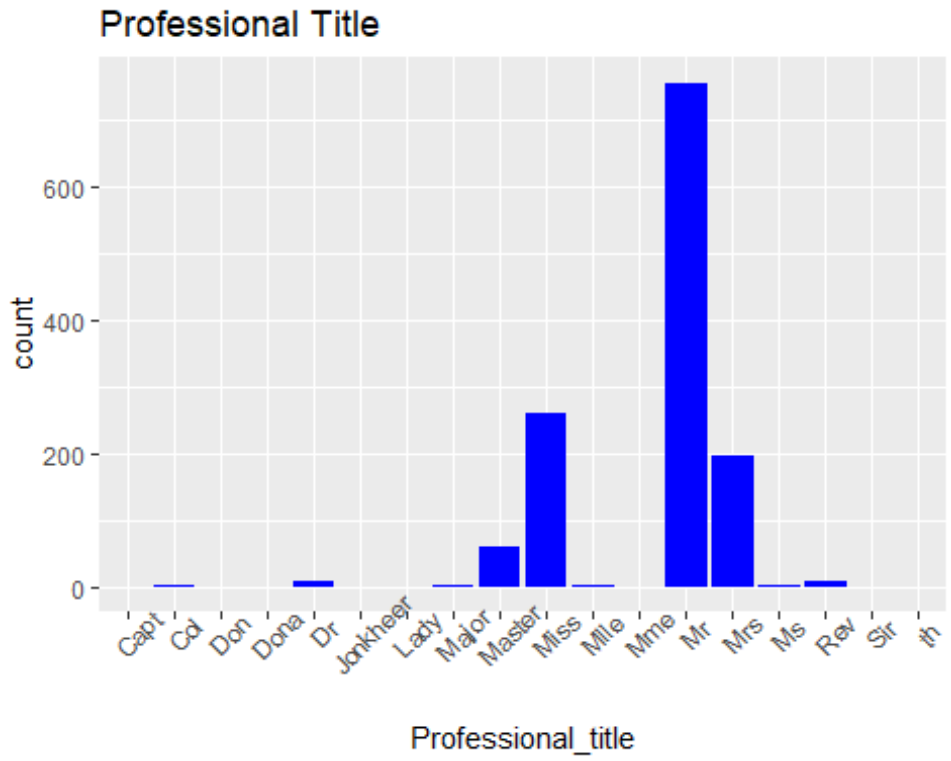
Check for the **NA** if any,

```
table(is.na(alldata$Professional_title))
```

```
##
## FALSE
##  1309
```

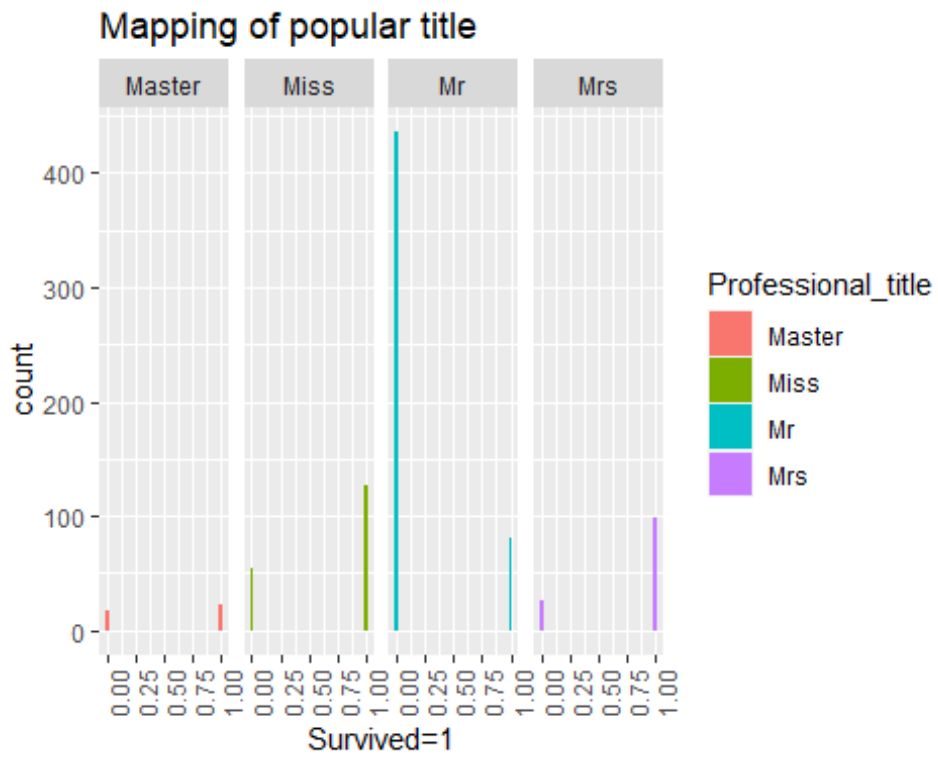
As there is no **NA** so lets make a plot of total counts of all titles.

```
ggplot(data=alldata) +
  geom_bar(mapping=aes(x=Professional_title), fill='blue') +
  labs(title="Professional Title") +
  theme(axis.text.x = element_text(angle=45))
```



Now, we observe the popular title to have the idea about Survival for different title holders.

```
ggplot(data = alldata %>% filter(Professional_title %in% c("Mr", "Miss",
"Mrs", "Master"))) +
  geom_histogram(mapping=aes(x=Survived, fill=Professional_title))+
  facet_grid(~Professional_title)+
  labs(title="Mapping of popular title", x="Survived=1")+
  theme(axis.text.x=element_text(angle=90))
```



Converting the rare titles into popular ones

Let's observe the data of *male title* and draw some insights about them.

```
alldata %>%
  filter(Professional_title %in% c("Capt", "Col", "Don", "Dr", "Jonkheer",
"Major", "Rev", "Sir"))
```

```
## # A tibble: 26 × 12
##    PassengerId Survi…¹ Pclass Profe…² Sex     Age SibSp Parch Ticket  Fare
Cabin
##          <dbl>   <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1          31       0      1 Don     1        40     0     0 PC 17…  27.7
<NA>
##  2         150       0      2 Rev     1        42     0     0 244310  13
<NA>
##  3         151       0      2 Rev     1        51     0     0 S.O.P…  12.5
<NA>
##  4         246       0      1 Dr      1        44     2     0 19928   90
C78
##  5         250       0      2 Rev     1        54     1     0 244252  26
<NA>
##  6         318       0      2 Dr      1        54     0     0 29011   14
<NA>
##  7         399       0      2 Dr      1        23     0     0 244278  10.5
<NA>
```

```
##  8         450       1      1 Major    1        52     0     0 113786  30.5
C104
##  9         537       0      1 Major    1        45     0     0 113050  26.6
B38
## 10         600       1      1 Sir      1        49     1     0 PC 17…  56.9
A20
## # … with 16 more rows, 1 more variable: Embarked <chr>, and abbreviated
## #   variable names ¹Survived, ²Professional_title
```

Let's rename the titles like *Capt*, *Col*, *Don*, *Dr*, *Jonkheer*, *Major*, *Rev*, *Sir* into *Mr* for making our life easy as there is no important insight and No. of Survived are equivalent to *Mr*….

```
alldata$Professional_title[alldata$Professional_title %in% c("Capt", "Col",
"Don", "Dr", "Jonkheer", "Major", "Rev", "Sir")] <- "Mr"
```

Do the same process with *female title*.

```
alldata %>%
  filter(Professional_title %in% c("Dona", "Lady", "Mlle", "Mme", "th"))
```

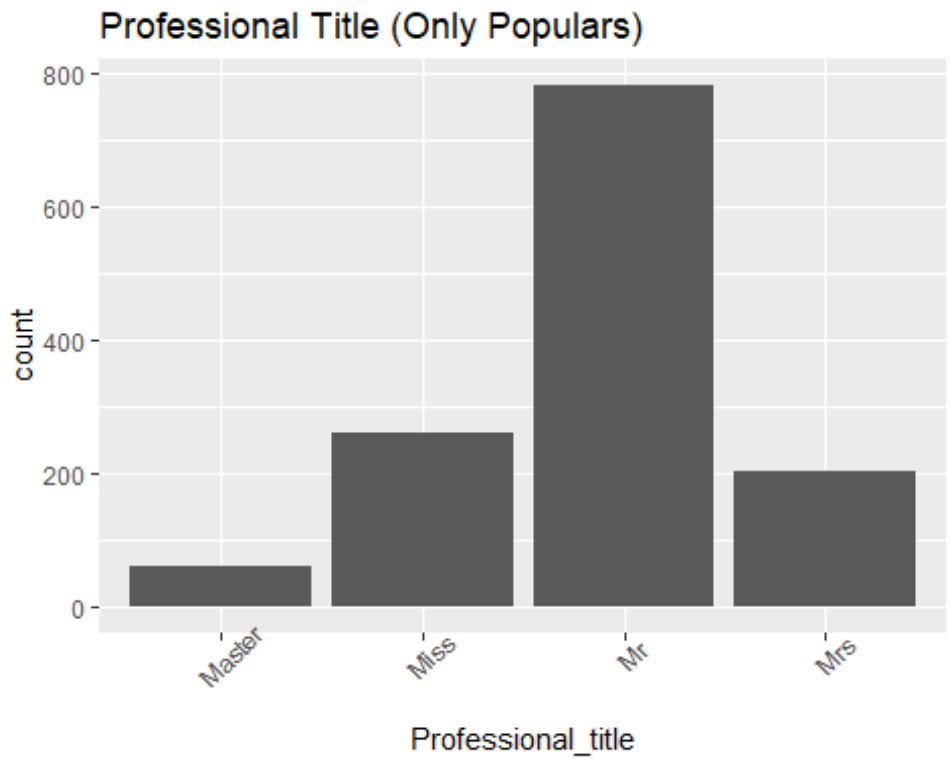
```
## # A tibble: 6 × 12
##   PassengerId Survived Pclass Profe…¹ Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1         370        1      1 Mme     0        24     0     0 PC 17…  69.3
B35
## 2         557        1      1 Lady    0        48     1     0 11755   39.6
A16
## 3         642        1      1 Mlle    0        24     0     0 PC 17…  69.3
B35
## 4         711        1      1 Mlle    0        24     0     0 PC 17…  49.5
C90
## 5         760        1      1 th      0        33     0     0 110152  86.5
B77
## 6        1306       NA      1 Dona    0        39     0     0 PC 17… 109.
C105
## # … with 1 more variable: Embarked <chr>, and abbreviated variable name
## #   ¹Professional_title
```

Let's rename the titles like *Dona*, *Lady*, *Mlle*, *Mme*, *th* into *Mrs* and *Ms* into *Miss* as Ms is basically the short of Miss.

```
alldata$Professional_title[alldata$Professional_title %in% c("Dona", "Lady",
"Mlle", "Mme", "th")] <- "Mrs"

alldata$Professional_title[alldata$Professional_title == "Ms"] <- "Miss"

ggplot(data=alldata) +
  geom_bar(mapping=aes(x=Professional_title)) +
  labs(title="Professional Title (Only Populars)") +
  theme(axis.text.x = element_text(angle=45))
```



Embarked

First of all see the **NA** value if any in the *Embarked* column

```
table(is.na(alldata$Embarked))
```

```
##
## FALSE  TRUE
##  1307     2
```

Observe the data

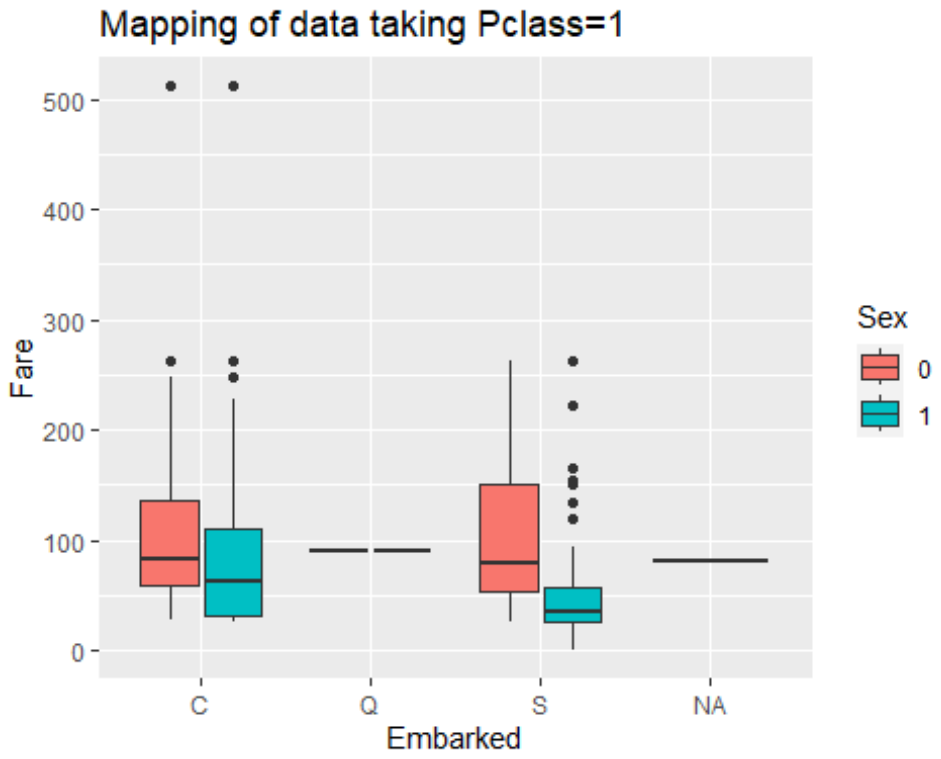
Review the both rows and draw the conclusion to fill the missing values.

```
alldata %>%
  filter(is.na(Embarked))
```

```
## # A tibble: 2 × 12
##   PassengerId Survived Pclass Profe…¹ Sex     Age SibSp Parch Ticket  Fare
Cabin
##         <dbl>    <dbl>  <dbl> <chr>   <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1          62        1      1 Miss    0        38     0     0 113572    80
B28
## 2         830        1      1 Mrs     0        62     0     0 113572    80
B28
## # … with 1 more variable: Embarked <chr>, and abbreviated variable name
## #   ¹Professional_title
```

We can see that both have *Pclass*=1, same *Ticket* number, *Fare*=80, same *Cabin* and both *female*.

```
ggplot(alldata %>% filter(Pclass==1)) +
  geom_boxplot(mapping=aes(x=Embarked, y=Fare, fill=Sex)) +
  labs(title="Mapping of data taking Pclass=1")
```



With the help of plot we can conclude that missing value is the **C**.

Filling the missing value

```
alldata$Embarked[is.na(alldata$Embarked)] <- "C"
```

Age

Let's see the **NA** value if any in the *Age* column.

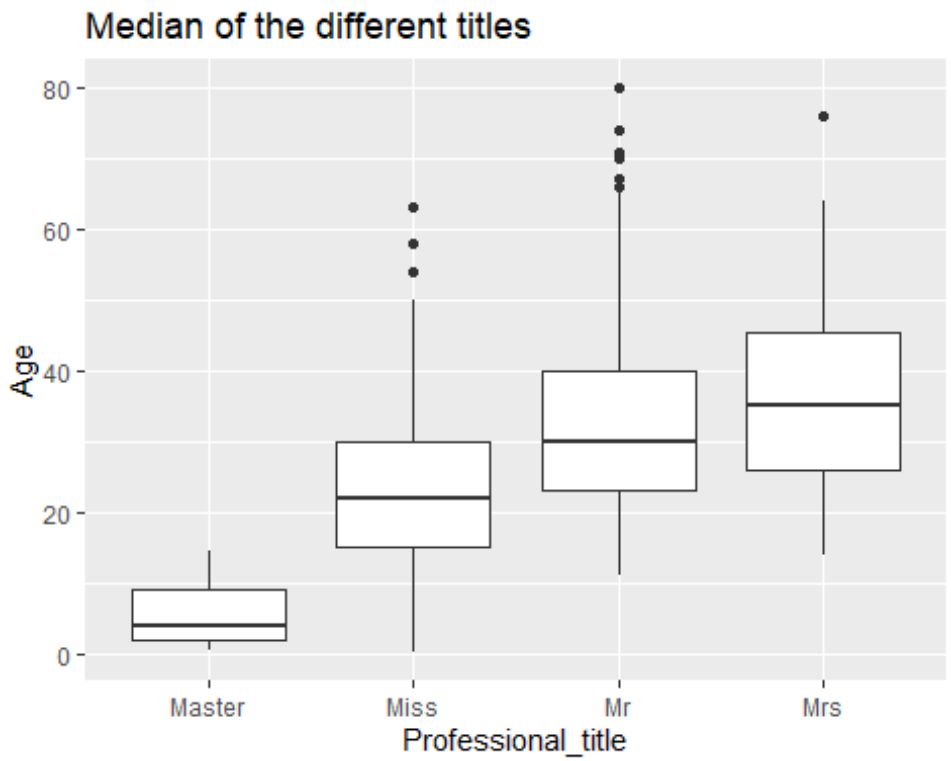
```
table(is.na(alldata$Age))

##
## FALSE  TRUE
##  1046   263
```

Let' draw a plot to have a better idea about *age* and *title* as they can have some relationship to fill our missing.

```
ggplot(alldata) +
  geom_boxplot(mapping=aes(x=Professional_title, y=Age)) +
  labs(title = "Median of the different titles")
```

```
## Warning: Removed 263 rows containing non-finite values (`stat_boxplot()`).
```



Let's take the **median** of the respective *Professional_title* and fill the missing value respectively.

Filling Master title

```
master_df <- alldata %>%
  filter(Professional_title=="Master")
master_df$Age[is.na(master_df$Age)] <- median(master_df$Age, na.rm=T)
```

Filling Miss title

```
miss_df <- alldata %>%
  filter(Professional_title=="Miss")
miss_df$Age[is.na(miss_df$Age)] <- median(miss_df$Age, na.rm=T)
```

Filling Mr title

```
mr_df <- alldata %>%
  filter(Professional_title=="Mr")
mr_df$Age[is.na(mr_df$Age)] <- median(mr_df$Age, na.rm=T)
```

Filling Mrs title

```
mrs_df <- alldata %>%
  filter(Professional_title=="Mrs")
mrs_df$Age[is.na(mrs_df$Age)] <- median(mrs_df$Age, na.rm=T)
```

Binding

Now, merge the data into again *alldata* dataset.

```
alldata <- rbind(master_df, miss_df, mr_df, mrs_df)
alldata <- alldata %>%
  arrange(PassengerId)
```

Ticket

Check the **NA** if any,

```
table(is.na(alldata$Ticket))
```

```
##
## FALSE
##  1309
```

Take the sample and observe the data of *ticket* column.

```
sample(alldata$Ticket, 30)
```

```
##  [1] "F.C.C. 13529"     "243847"           "347080"           "SOTON/O2
3101287"
##  [5] "28220"            "248738"           "29108"            "C.A. 29566"
##  [9] "11765"            "315095"           "349251"           "315090"
## [13] "F.C.C. 13540"     "C.A. 34644"       "C.A. 17248"       "237671"
## [17] "1601"             "237789"           "382649"           "C 4001"
## [21] "PP 4348"          "4133"             "349238"           "315089"
## [25] "PC 17558"         "248727"           "239059"           "345777"
## [29] "9234"             "CA 31352"
```

Can't have any relevance to the survival of the passengers, so decided to remove the column

```
alldata <- alldata %>%
  select(-Ticket)
```

Pclass

Check the **NA** if any,

```
table(is.na(alldata$Pclass))
```

```
##
## FALSE
##  1309
```

SibSp

Check the **NA** if any,

```
table(is.na(alldata$SibSp))
```

```
##
## FALSE
##  1309
```

Parch

Check the **NA** if any,

```
table(is.na(alldata$Parch))
```

```
##
## FALSE
##  1309
```

Cabin

Check the **NA** if any,

```
table(is.na(alldata$Cabin))
```

```
##
## FALSE  TRUE
##   295  1014
```

As there are many empty cells in the *Cabin* column so best possible solution is to drop the column only.

```
alldata <- alldata %>%
  select(-Cabin)
```

Appling Model into our cleaned data

Let's apply a randomForest

Honestly didn't know the shit about this models right now :(
but let's apply to our data.

```
i <- is.na(alldata$Survived)
myforest <- randomForest(data=alldata[!i,], Survived ~ Pclass + Age + SibSp +
Parch + Fare + Embarked + Sex + Professional_title,
                              ntree=10000, sampsize = 400, mtry=4)

## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values.  Are you sure you want to do regression?
```

Now let's look the **model** works decent or not...

```
table(round(predict(myforest, newdata=alldata[!i,])) ==
alldata[!i,]$Survived)
```

```
##
## FALSE  TRUE
##    89   802
```

Add the predictions into our data for submissions

```
alldata$forestpred <- round(predict(myforest, newdata=alldata))
rm(myforest)
```

Export the **.csv** file

```
write.csv(alldata %>%
  filter(PassengerId %in% c(892:1310)) %>%
  select(PassengerId, forestpred) %>%
  rename(Survived = forestpred), "submission.csv")
```

Now deleted the No. column in the Excel manually and submitted the data……..**0.77272**
Great for my first Project.

Thank you