Titanic Life Prediction

Suberlin Sinaga

A. Background and Problem Statement

The Titanic ship sank years ago. Some of the passengers are survived from the disaster while the rest can't make it. The data indicates that the Titanic's passenger survival condition follow some specific pattern. The objective is to predict the survivalability of a passenger by considering known factors into account.

B. Data Understanding

The data used comes from famous kaggle titanic problem set.

```r
# here::here()
titanic_training_set <- read.csv(here::here("data/titanic/train.csv"))
titanic_test_set <- read.csv(here::here("data/titanic/test.csv"))
```

Since in this early dataset the training and testing set was separated, I will combine them to ensure that I understand all the data as a whole.

```r
titanic_raw_all <- rbind(
    titanic_test_set %>% mutate(Survived = NA, Source = "test set"),
    titanic_training_set %>% mutate(Source = "train set")
)

y_var <- titanic_training_set$Survived
```

Naturally, the testing set has no target variable, hence I set the target variable values into `NA` when the data came from testing set.

```
'data.frame':   1309 obs. of  13 variables:
 $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
 $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
 $ Name       : chr  "Kelly, Mr. James" "Wilkes, Mrs. James (Ellen Needs)" "Myles, Mr. Thomas
 $ Sex        : chr  "male" "female" "male" "male" ...
 $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
 $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
 $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
 $ Ticket     : chr  "330911" "363272" "240276" "315154" ...
 $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
 $ Cabin      : chr  "" "" "" "" ...
 $ Embarked   : chr  "Q" "S" "Q" "S" ...
 $ Survived   : int  NA NA NA NA NA NA NA NA NA NA ...
 $ Source     : chr  "test set" "test set" "test set" "test set" ...
```

As shown, the dataset has 12 variables (exclude the Source variable). Between the 12 variables, 1 of them is dependent variable while the rest are independent. I also find out the variable like pclass, sex, survived cabin, and embarked are stored in a wrong data type. I think they will be better stored as factor instead of character in R. Lastly, I think, age will be better to be stored as integer with ceiling (0.1 will be counted as 1).

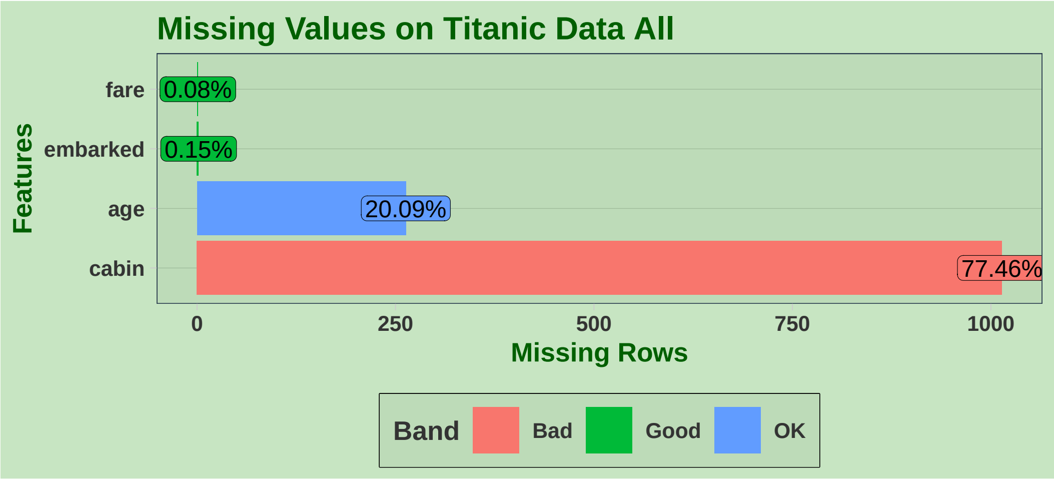
I think, there are some wrong assignment here on the data. For example, the passenger_id is stored as integer, it should be better to store them as character like ticket.

Another thing that I find out is that the variables name started by capital letter here. Just for the standardization, I will use snake_case format for the variable name.

Now, I will start to see overall summary of the data.

```
Total Rows : 1309
Total Columns : 12
Total Observations : 15708
Total Discrete Columns : 7
Total Continuous Columns : 5
Total Missing Values : 682
```

The next thing that I am curious of is about the missing values. Where did they come from.



From the data, I see that cabin is the variable with highest number of the missing values, it is about ~77.46%. For now, this information is just as reference. I will deal with missing values later on data preprocessing.

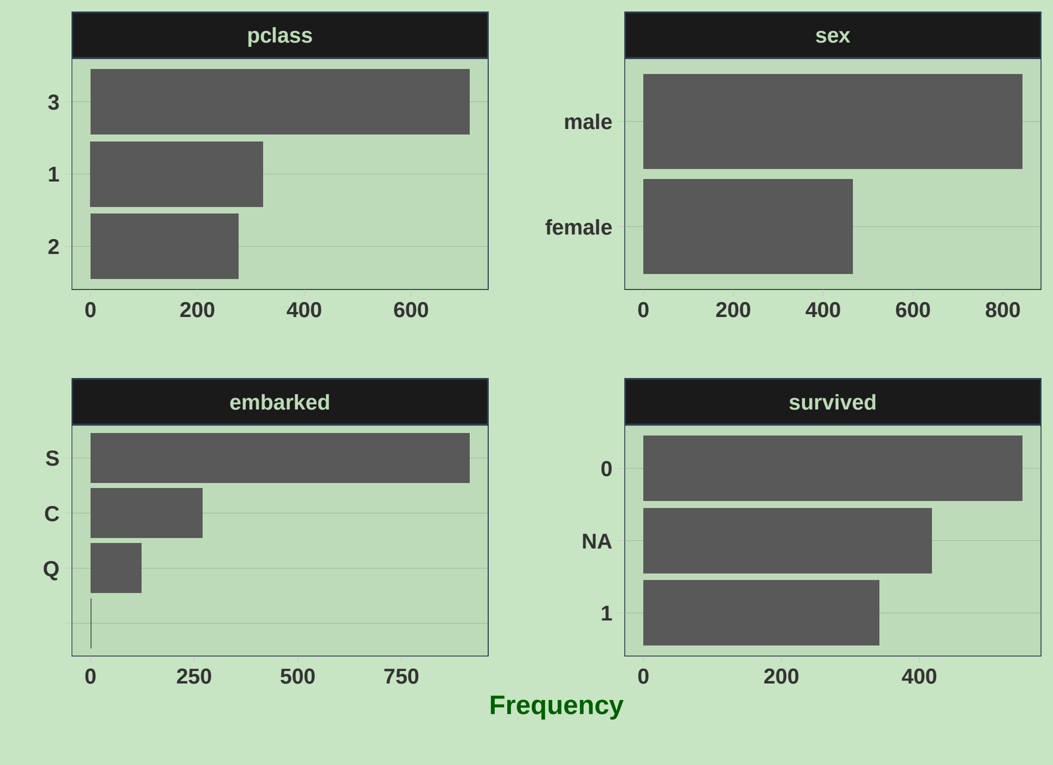
C. Exploratory Data Analysis

Univariate Data Analysis

In order to do univariate analysis, I will divide the data into two types, the discrete variable and continuous variable.

Discrete Variables

```
3 columns ignored with more than 50 categories.
name: 1307 categories
ticket: 929 categories
cabin: 187 categories
```

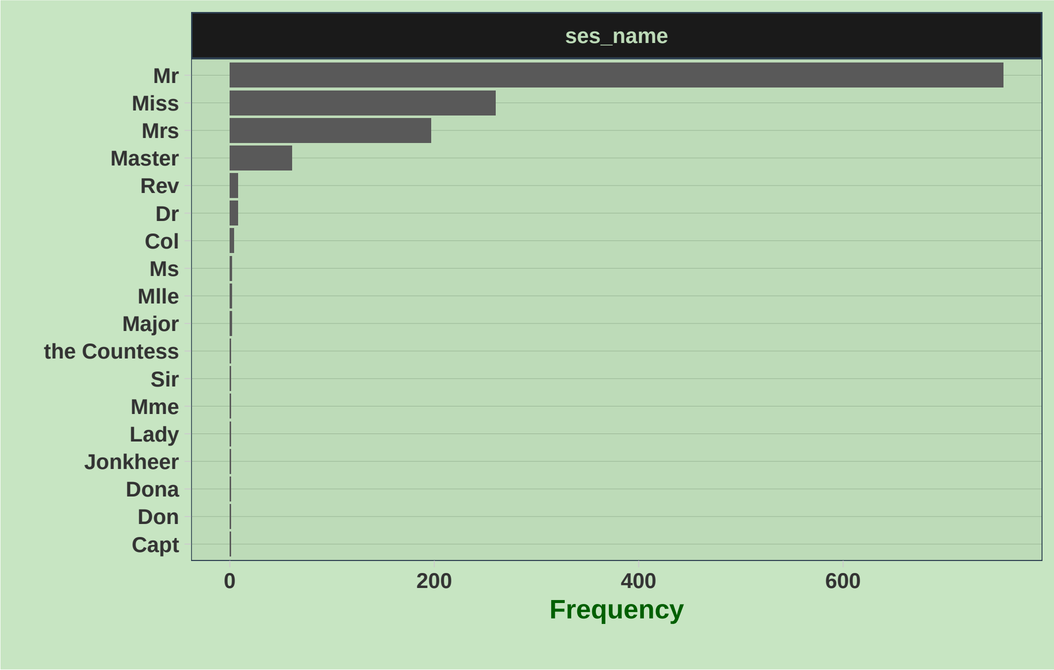


Due to category limitation to 50 as max, there are only 4 variables that can be plotted. From the plot, I can see that majority people are from class 3 (which is the cheapest class). People on-boarded are mostly men where the passengers mostly embarked from S port. The empty string on embarked plot is equal to missing values or unknown embarked location. The “NA” on survived variables indicates that the data comes from testing set with unknown survival condition (need to be predicted). From the known survival condition, we know that most of the passengers are death.

As for the rest variables, such as name, ticket and cabin I think, I will print a few of them to see what is in the data.

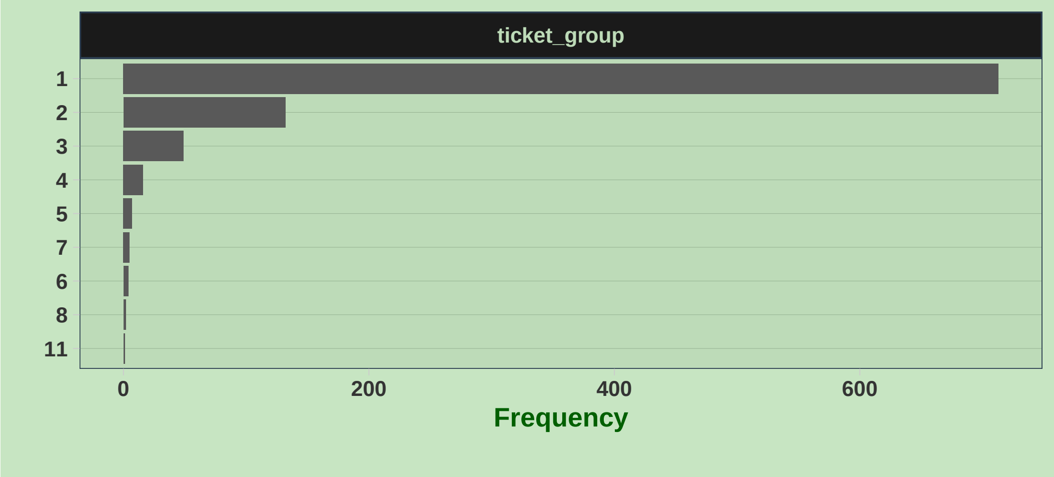
```
# A tibble: 10 x 3
   name                                                    ticket      cabin
   <chr>                                                   <chr>       <fct>
 1 Snyder, Mrs. John Pillsbury (Nelle Stevenson)           21228       B45
 2 Chaffee, Mrs. Herbert Fuller (Carrie Constance Toogood) W.E.P. 5734 E31
 3 Ryerson, Mrs. Arthur Larned (Emily Maria Borie)         PC 17608    B57 B59 ~
 4 Ostby, Miss. Helene Ragnhild                            113509      B36
 5 Brady, Mr. John Bertram                                 113054      A21
 6 Mock, Mr. Philipp Edmund                                13236       C78
 7 Franklin, Mr. Thomas Parham                             113778      D34
 8 Kimball, Mrs. Edwin Nelson Jr (Gertrude Parsons)       11753       D19
 9 Chevre, Mr. Paul Romaine                               PC 17594    A9
10 Bucknell, Mrs. William Robert (Emma Eliza Ward)        11813       D15
```

From the data, I can see that names contain unique identifier such as Mrs, Mr, and Miss. This might be a useful feature to predict the survive condition. Let's see the distribution of this variable.



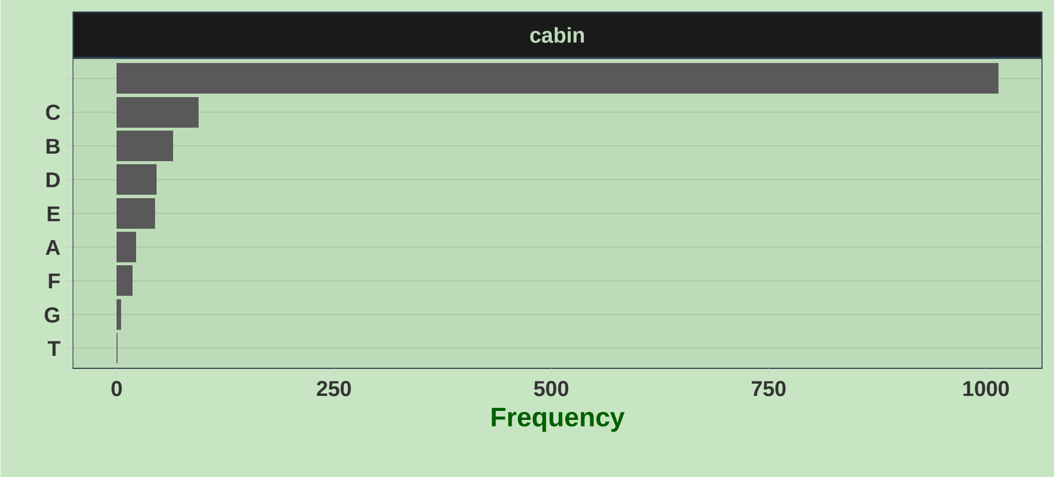
Interesting. It seems that there are various sosio economic status here that can be explored further.

As for ticket, it seems that there is a specific pattern found. For example, Ryerson, Mrs. Arthur Larned (Emily Maria Borie) and Chevre, Mr. Paul Romaine have ticket that together started with "PC". I think they are correlated somehow like departed from the same port. Now, I think I will check if the ticket can show some groups.



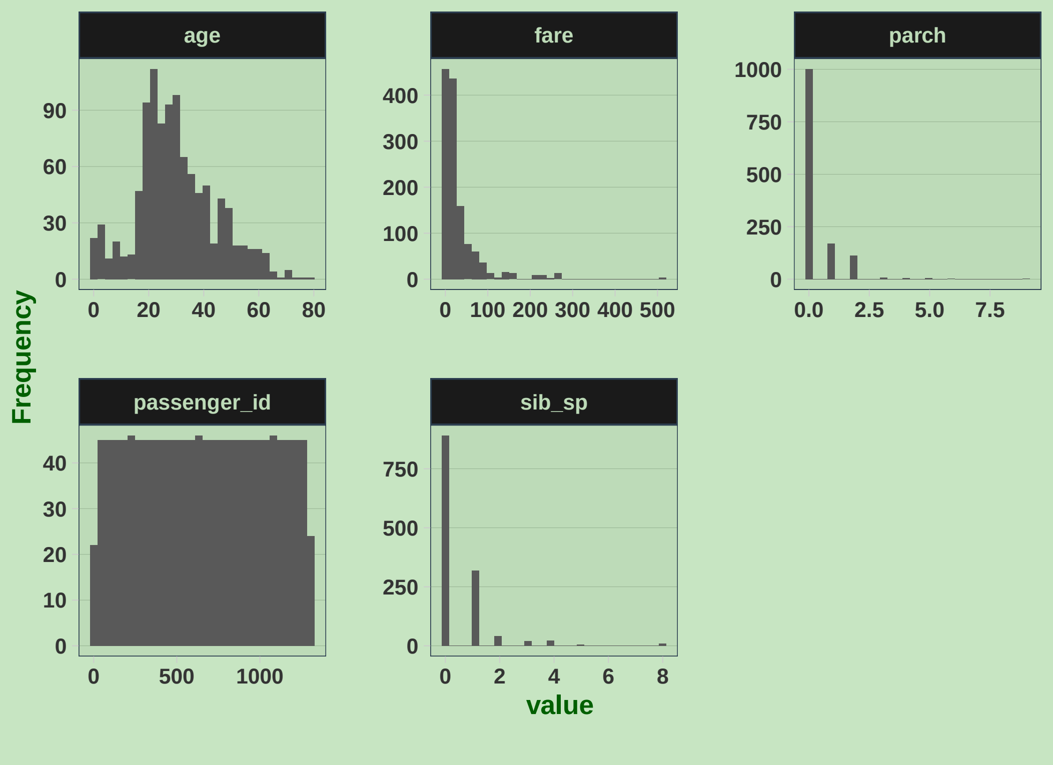
The plot indicates that most of the passengers are traveling alone where they mostly have ticket group equal to 1. I think, this information is also can be useful in the future.

Now, talking about cabin. One of the reason why so many people not survive is the failed to get life boat. The cabin's position defining the possibility to get into boat. Titanic cabin plan as explain here area consisted of boat deck, A deck, B deck, C deck, D deck, E deck, F deck, and G deck.



Many passengers have unknown cabin position. It is represented by the blank data of the cabin. considering that there are some known cabins to be explored, I think I can fill the missing with M to represent unknown data.

Numeric Variables



Based on its histogram, the age seems to be concentrated around 20 to 40 years old. I think there is outlier indeed found in the data, but I think it is a true values of the data and Will not do anything to it.

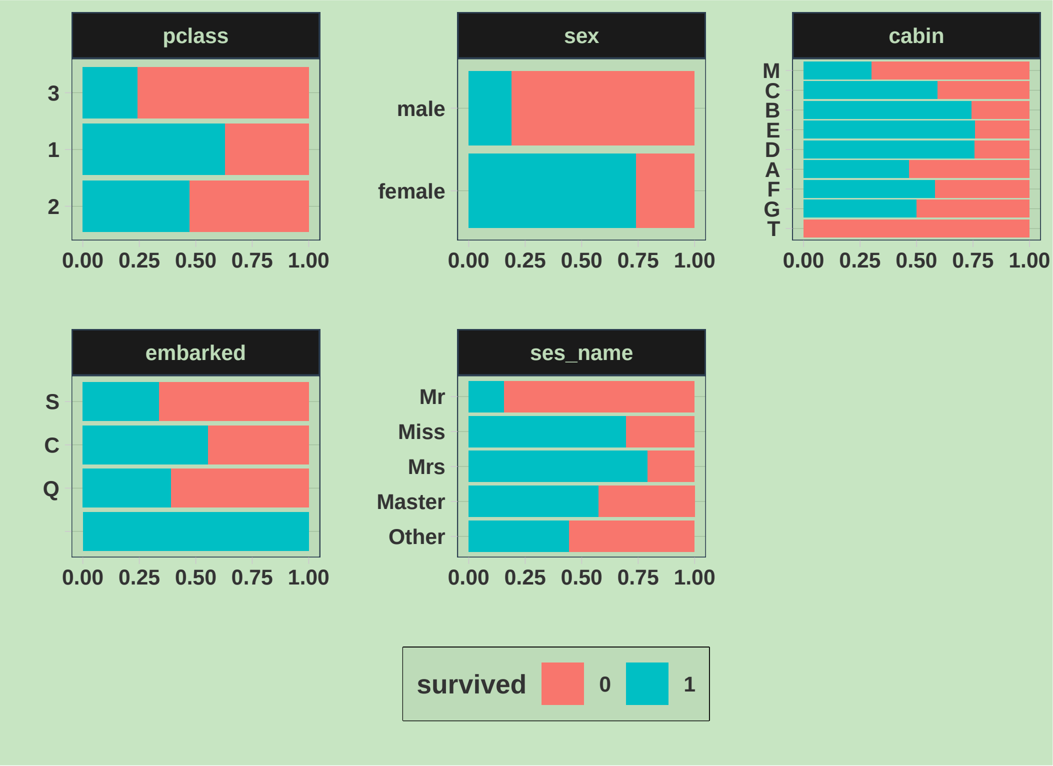
While fare variable seems to be very right skewed due to so many passengers buy the cheapest fare. The parch variable has poisson distribution where most of the values lies on 0. It means that most of the passengers have no parents and or children. Passenger_id variable looks uniform since passenger id is just an ID to uniquely identify each passenger. The sib_sp variable is like parch variable, it has poisson distribution where the most values are 0 that indicates people have no sibling and or spouse gets on-boarded on the ship.

Take a look at the passenger id variable. It looks like a uniform distributed. While, it is because the passenger id is just a random values used to uniquely identify each passenger. Since I can't see the use of that other than only as unique identifier, I will not analyze it further.

Bivariate Data Analysis

Nominal Vaiable vs Target

```
2 columns ignored with more than 50 categories.
name: 891 categories
ticket: 681 categories
```



I think, all those 5 variables have an effect on survivability. Let's take pclass for example, the higher the class, the higher the probability of the passengers to survive. This indicates that pclass variable is good enough to distinguish between the one who survived and the one who is not.

Then, come sex variable. It can also distinguish between the survive and not. Even we can say, if by any chance the algorithm said that all the woman is survive and all the men is not, it might have greater than 75% in term of accuracy.

As for cabin, it seems that there is no significant difference between cabin D, E, and B, and also between C and F. Even though they show no difference, I think I will use them as is without grouping them into 1 cabin group. This is because they might have different effect when another variables take place.

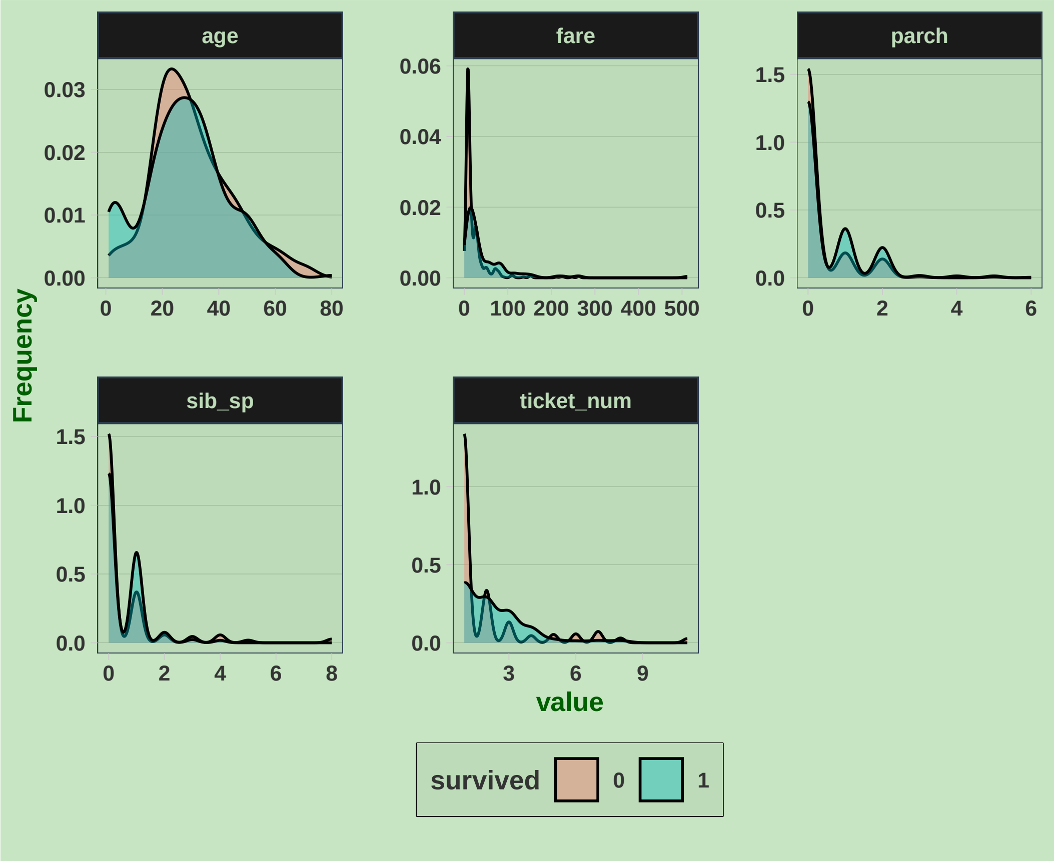
As for embarked variable, it also indicates that there are some difference of survivability between the one who were embarked from S, C, and Q port. There are missing values found in the data. Let me check them out.

```
  passenger_id pclass                                         name    sex age
1           62      1                          Icard, Miss. Amelie female  38
2          830      1 Stone, Mrs. George Nelson (Martha Evelyn) female  62
  sib_sp parch ticket fare cabin embarked survived    source
1      0     0 113572   80   B28                 1 train set
2      0     0 113572   80   B28                 1 train set
```

Based on this source both of those people were embarked from Southampthon.

As for ses_name variable, just like sex variable, I can see strong difference of survivability between the one entitled as Mr, Miss, Mrs, Master and other. Hence I think, this variable is also a good feature to be included.

Numerical Variable vs Target



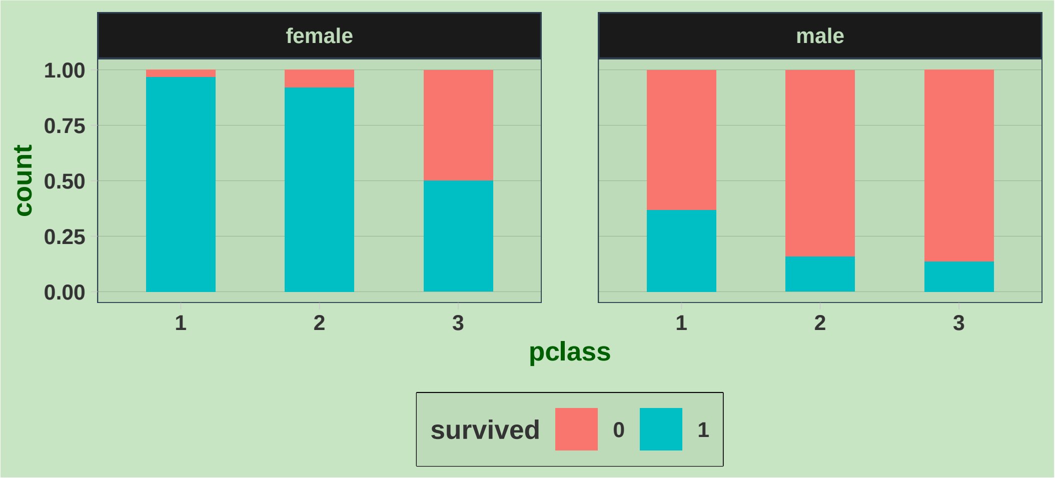
Based on the plot, I can see there seems difference between the distribution of the survive and the non survive passengers. Take fare and age for example. The peak of the distribution for survived passengers are around 25 while for non survived passengers are around 20 years old. While for fare, it seems that the higher the fare, the higher the survivability that the passengers have.

Multivariate Data Analysis

Multivariate analysis used to answer some question where more than two variables needed.

1. Passenger's Class Is Independent to Gender

Hypothetically, I would like to see that passenger's class is not related to their gender.

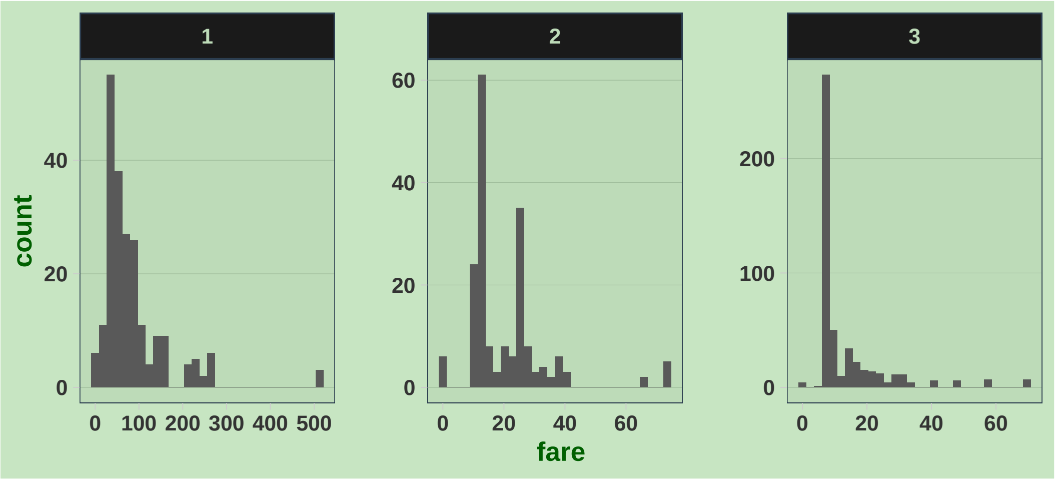


Considering the plot, where pclass 1 always has highest survival rate across gender, I think passenger's class has nothing to do with gender.

2. Fare Is Very Related to Passenger's Class

Previously it was found that fare has some missing values. In my opinion, the fare should be related to passenger's class. It is like, the higher the fare, the higher the class and the better facilities will be enjoyed by the passengers.

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



As seen, the range of fare for each class is slightly different.

```
# A tibble: 3 x 3
  pclass min_fare max_fare
  <fct>     <dbl>    <dbl>
1 1             0    512.
2 2             0     73.5
3 3             0     69.6
```

One thing that is the same from all the class type is the minimum values. All classes started from 0. There are so many reasons why those people have 0 fares. Let's take Leonard, Mt. Lionel for example. The fare is 0 because they were given the third class accommodation from LINE company to get back to America.

```
                                    name fare is_servant ticket survived
1   Chisholm, Mr. Roderick Robert Crispin    0         NA 112051     <NA>
2                 Ismay, Mr. Joseph Bruce    0         NA 112058     <NA>
3                     Leonard, Mr. Lionel    0         NA   LINE        0
4                   Harrison, Mr. William    0       TRUE 112059        0
5            Tornquist, Mr. William Henry    0         NA   LINE        1
6             Parkes, Mr. Francis "Frank"    0         NA 239853        0
7         Johnson, Mr. William Cahoone Jr    0         NA   LINE        0
8          Cunningham, Mr. Alfred Fleming    0         NA 239853        0
9                   Campbell, Mr. William    0         NA 239853        0
10       Frost, Mr. Anthony Wood "Archie"    0         NA 239854        0
11                    Johnson, Mr. Alfred    0         NA   LINE        0
12           Parr, Mr. William Henry Marsh    0         NA 112052        0
13              Watson, Mr. Ennis Hastings    0         NA 239856        0
```

```
14                  Knight, Mr. Robert J    0         NA 239855        0
15                Andrews, Mr. Thomas Jr    0         NA 112050        0
16                      Fry, Mr. Richard    0       TRUE 112058        0
17       Reuchlin, Jonkheer. John George    0         NA  19972        0
         cabin passenger_id
1                      1158
2  B52 B54 B56         1264
3                       180
4          B94          264
5                       272
6                       278
7                       303
8                       414
9                       467
10                      482
11                      598
12                      634
13                      675
14                      733
15         A36          807
16        B102          816
17                      823
```

In fact, those people which had 0 fare, has their own story. Take Reuchlin, Jonkheer. John George for example. Person who has passenger ID of 823 here got free ticket due to his position with Holland America Line which was part of the International Mercantile Marine.

Another think that I would like to check is about the missing value in fare.

```
  passenger_id pclass               name  sex age sib_sp parch ticket fare
1         1044      3 Storey, Mr. Thomas male  61      0     0   3701   NA
  cabin embarked survived   source
1              S     <NA> test set
```

Based on this source, the person who has missing value in their fare is Storey, Mr. Thomas. Based on this source, he is also part of postophoned Philadelpia westbound voyage along with Leonard, Mr. Lionel and the others. Based on this fact, then his fare value should be 0, since it is sponsored.

3. Age Ranges Based on Their SeS Title

Previously mentioned that age has some missing values on it. I believe the that the best way to fil the missing values is by grouping based on their SeS title and not their age or class. Let me show you why.

```
Age range based on SeS name.
```

```
# A tibble: 5 x 5
  ses_name total_data average min_age max_age
  <chr>         <int>   <dbl>   <int>   <int>
1 Master           61    5.55       1      15
2 Miss            260   21.8        1      63
3 Mr              757   32.3       11      80
4 Mrs             197   37.0       14      76
5 Other            34   42.7       23      70
```

```
Age range based on gender.
```

```
# A tibble: 2 x 5
  sex    total_data average min_age max_age
  <fct>       <int>   <dbl>   <int>   <int>
1 female        466    28.7       1      76
2 male          843    30.6       1      80
```

It is valid that the gender can not be used to range the age, SeS title instead.

D. Data Preprocessing

Data Splitting

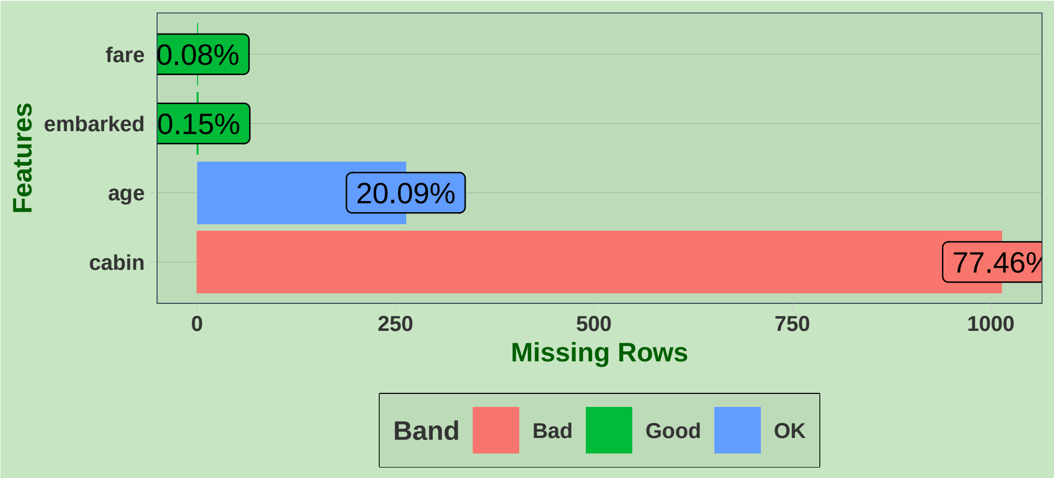
I will split the data into 3 parts, the training data, testing data, and validating data. I also want to build cross validation set from training data set.

```
train_test_source <- titanic_raw_all %>%
    filter(source == "train set")

train_idx <- initial_split(train_test_source, prop = 0.8)

train_set <- training(train_idx)
```

```
cross_val <- mc_cv(train_set)

test_set <- testing(train_idx)

val_set <- titanic_raw_all %>%
    filter(source == "test set") %>%
    select(-source)
```

Data Cleaning

Previously I know that the data contains missing values that need to be fixed. First let me display the missing data again.



Mostly, there are 4 variables with missing values found in the data. For cabin, I will fill the missing values with "M" that represents missing values for the cabin.

For age, I will fill the values using mean, since the data indicates that the distribution is a bit normal for the age. but, filling up age missing values have to consider their age range. For example, if the person is a "Master" then the average values of the age should be 7 years old. While if the person is a "Miss", then the average values can't be less than 17 years old assuming that people will get married at 17 years old at minimum.

For fare, I will input the values as 0 since the data that missing is coming from sponsored passenger which has 0 fare.

As for cabin, I will first simply put their initial cabin name, and put M when it is a missing values.

In order to clean up the data, I will first create a recipe to be used until final work. But, since tidyverse not supporting imputation based on group yet, I will impute the missing age data using the training set manually.

Feature Addition

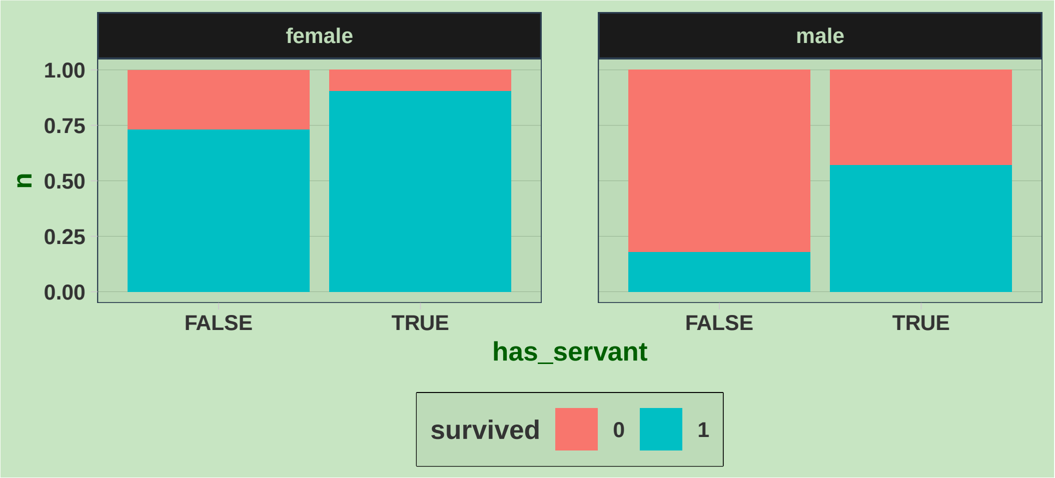
In this section I will add some features from external data. Take a look at the following example data.

```
  survived pclass is_servant    sex  n
1        0      1      FALSE female  3
2        1      1      FALSE female 74
3        1      1       TRUE female 17
4     <NA>      1      FALSE female 42
5     <NA>      1       TRUE female  8
```

This is a condition from servant data, where all servant females proven to have higher survivability. Another data is about sponsored travel where they have 0 fare due to sponsorship for their travel

```
   sex survived  n
1 male        0 14
2 male        1  1
```

Another information that I would like to add is the data that indicates whether a passengers have servant or not.



As seen in the plot, each gender category will have better survival percentage when they have servant compare to the one who doesn’t.

Feature Engineering

1. Feature Extraction

In this feature extraction, I will extract some features based on previous exploration.

2. Feature Transformation

3. Feature Selection

Variable Correlation

```
# A tibble: 9 x 2
  var         survived
  <chr>          <dbl>
1 survived      1
2 fare          0.241
3 has_servant   0.162
4 age          -0.124
5 is_servant    0.120
6 parch         0.0784
7 ticket_num    0.0673
8 sib_sp       -0.0327
9 family_size   0.0146
```

Based on the correlation plot, using 0.02 as a threshold, I think family_size is not so significant.

Information Value For Categorical Variable

```
      variable         iv             status
1     ses_name 1.42601487   Highly Predictive
2          sex 1.25035779   Highly Predictive
3        cabin 0.57589233   Highly Predictive
4       pclass 0.46952579   Highly Predictive
5     embarked 0.16660716   Highly Predictive
6 age_category 0.08145422 Somewhat Predictive
```

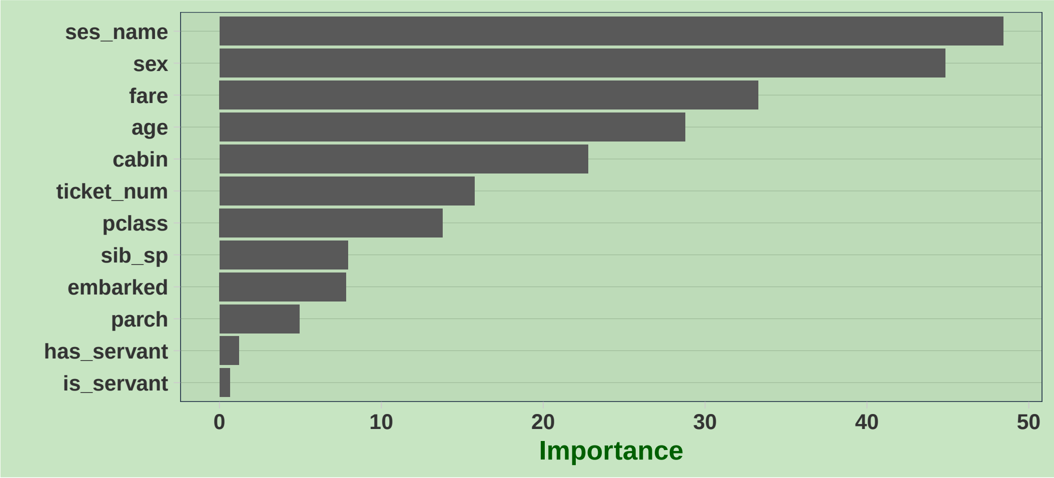
based on the data, age category is the only variable with somewhat predictive, but I think it is still okay.

Variable Importance Using Random Forest Algorithm

```
set.seed(123)
r_forest <- rand_forest(mode = "classification") %>%
    set_engine("ranger", importance = "impurity")

r_forest_wf <- workflow() %>%
  add_model(r_forest) %>%
  add_recipe(feat_select_iv)

r_forest_mdl <- r_forest_wf %>%
  fit_resamples(cross_val)

collect_metrics(r_forest_mdl)
```

```
# A tibble: 2 x 6
  .metric  .estimator  mean     n std_err .config
  <chr>    <chr>      <dbl> <int>   <dbl> <chr>
1 accuracy binary     0.824    25 0.00429 Preprocessor1_Model1
2 roc_auc  binary     0.879    25 0.00488 Preprocessor1_Model1
```

```
# A tibble: 1 x 3
  .metric  .estimator .estimate
  <chr>    <chr>          <dbl>
1 accuracy binary         0.844
```



Random forest indicates that all variables have some effect to the dependent variable so I will keep using them all.

E. DATA MODELING

To model the data, I will try to utilize the following model.

- Logistic regression
- Decision Tree
- Random Forest
- XGB (boosting tree)
- GBM (boosting tree)
- rpart (bagging tree)
- SVM

Model Training, Evaluation, and Selection

```r
# Logistic  Regression
set.seed(123)
logit_eng <- logistic_reg(penalty = "ROC")

logit_wf <- workflow() %>%
  add_model(logit_eng) %>%
  add_recipe(feat_select_iv)

logit_resamples <- logit_wf %>%
  fit_resamples(cross_val)

logit_mdl <- logit_wf %>%
    fit(data = train_set)
```

```r
# decision tree
d_tree <- C5_rules()
d_tree_wf <- workflow() %>%
  add_model(d_tree) %>%
  add_recipe(feat_select_iv)
set.seed(123)
```

```
d_tree_resamples <- d_tree_wf %>%
  fit_resamples(cross_val)

d_tree_mdl <- d_tree_wf %>%
  fit(data = train_set)
```

```
# random forest
r_forest <- rand_forest(mode = "classification")

r_forest_wf <- workflow() %>%
  add_model(r_forest) %>%
  add_recipe(feat_select_iv)

set.seed(123)
r_forest_resamples <- r_forest_wf %>%
  fit_resamples(cross_val)

r_forest_mdl <- r_forest_wf %>%
  fit(data = train_set)
```

```
# xtreme gradient boosting
my_recipe_xgb <- feat_select_iv %>%
    step_dummy(all_nominal_predictors())

xgb <- boost_tree(mode = "classification", engine = "xgboost")

xgb_wf <- workflow() %>%
  add_model(xgb) %>%
  add_recipe(my_recipe_xgb)

set.seed(123)
xgb_resamples <- xgb_wf %>%
  fit_resamples(cross_val)

xgb_mdl <- xgb_wf %>%
  fit(data = train_set)
```

```
# gradient boosting machine
gbm <- boost_tree(mode = "classification", engine = "lightgbm")

gbm_wf <- workflow() %>%
  add_model(gbm) %>%
  add_recipe(my_recipe_xgb)
set.seed(123)
gbm_resamples <- gbm_wf %>%
  fit_resamples(cross_val)
```

```
Warning: package 'lightgbm' was built under R version 4.2.3
```

```
gbm_mdl <- gbm_wf %>%
  fit(data = train_set)
```

```
# support vector machine
svm <- svm_linear(mode = "classification", engine = "kernlab")

svm_wf <- workflow() %>%
  add_model(svm) %>%
  add_recipe(feat_transform_recipe %>%
               step_dummy(all_nominal_predictors()))
set.seed(123)
svm_resamples <- svm_wf %>%
  fit_resamples(cross_val)

svm_mdl <- svm_wf %>%
  fit(data = train_set)
```

```
 Setting default kernel parameters
```

After training the model, I have the following summary of performance:

```
            algorithm accuracy_cv accuracy_test
1                 svm   0.8175281     0.8603352
2                 gbm   0.8222472     0.8379888
3                 xgb   0.8287640     0.8156425
4       random forest   0.8240449     0.8435754
5       decision tree   0.8130337     0.8324022
6 logistic regression   0.8226966     0.8435754
```

I think, the SVM model is the best here since it gives good predictin result to the test data.

Model Tuning

Here I will fine tune the SVM model

```
 Setting default kernel parameters
```

```
[1] 0.7751196
```

```
[1] 0.7799043
```

```
[1] 0.8286517
```

```
[1] 0.8659218
```