STAT 3302 Final Project Report

Zhengqi Dong

Shuhan Shen

Yunxiao Wang

Jaehyun Han

Introduction

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. On April 15, 2912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew. This sensational tragedy shocked the international community and led to better safety regulations for ships. The shortage of lifeboats led to such loss of life and some groups of people were more likely to survive than others[1].

Our group worked on the dataset of size 1309 and 12 variables that contains the information of Titanic passengers. By looking at different characteristics of the passengers and whether they survived or died from the incident, the scientific question to be answered through our analysis is what kinds of passengers were more related to survival. The following are the variables of interest in the dataset:

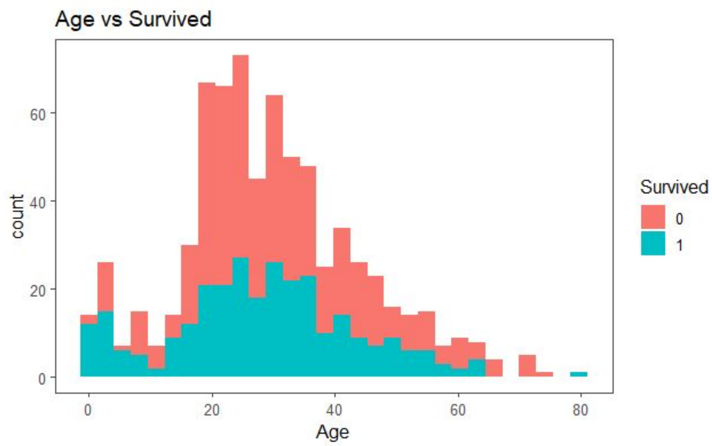
> <u>survival</u>: whether a passenger survived or died (0 = No; 1 = Yes)
> <u>pclass</u>: passenger class (1 = 1st, 2 = 2nd, 3= 3rd)
> <u>sex</u>: male or female
> <u>age</u>: age of the passenger in years (fractional if less than 1)
> <u>embarked</u>: port of embarkation (C = Cherbourg, Q = Queenstown, S = Southampon)

We first set *survival* as our response variable and excluded variables that are uniquely designated to each case, such as *name* and *ticket*. Among the remaining variables, some of them are particularly more interesting and worth investigating. For instance, *pclass* is a proxy for socio-economy status as the 1st may represent upper; 2nd , middle; and 3rd, lower. By investigating the relationship between survival and other potential variables, we can study what characteristics are more related to survival of a passenger from the incident. The training dataset we had was of size 891 (with some missing survival values) and the test dataset of size 418.

Exploratory Data Analysis

<u>(1) Age</u>

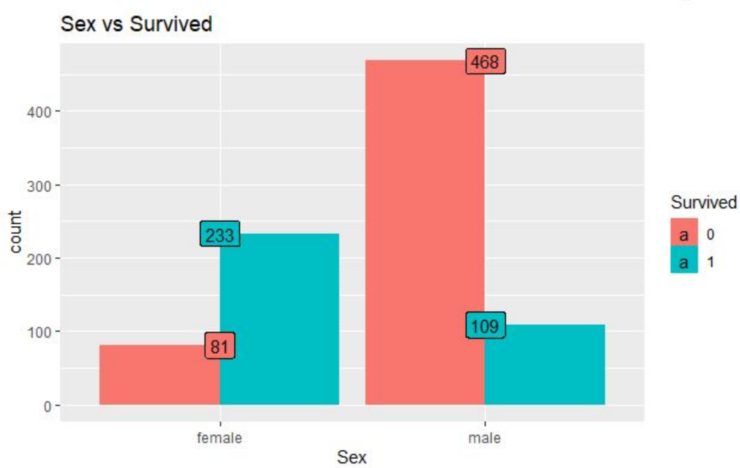
[1] Titanic: Machine Learning from Disaster, n.d.



The age range between 20 and 50 are slightly more likely to survive. The distribution looks right-skewed. Comparing the distribution of survived and dead passengers, they seem fairly similar.

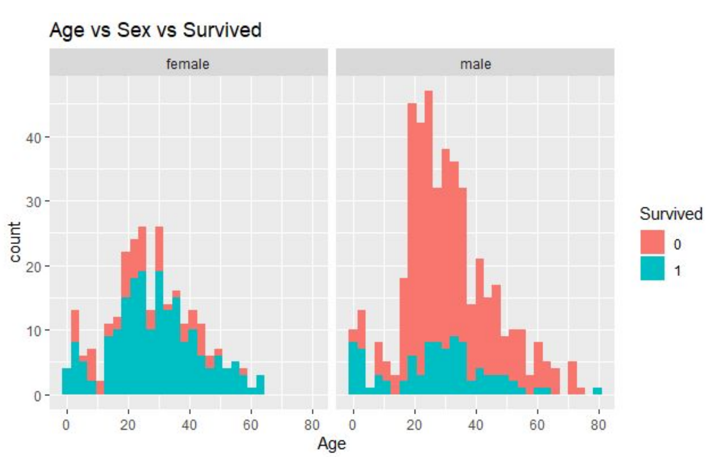
There are 177 missing values. There are many ways to solve the missing value, such as removing the data, or filling with zero or NaN. A decent way to do this is to create a model that predicts the average ages based on other variables. We used the mice library to deal with age and every variable below.

<u>(2) Sex</u>



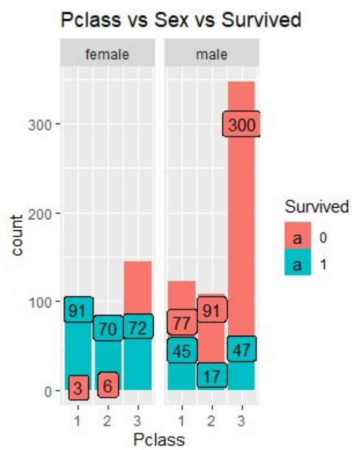
This graph shows that the percentage of female survivals is as high as roughly 75% while the male survivals rate is around 16.7%. Sex may be a meaningful factor to be studied.

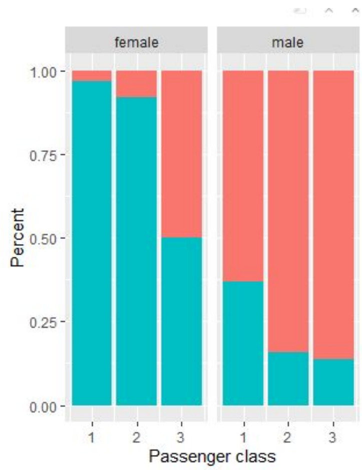
<u>(3) Age and Sex</u>



Similar trends showed when considered both age and sex. The difference between the number of survivals and the number of deaths is largest at age 20 to 30 for both females and male.

(4) Pclass vs Sex





These graphs show that higher classes were more likely to survive. Especially when class and sex are considered simultaneously, first and second class female passengers show an extremely high rate of survival while the second and third class male passengers had lower than 25% of survival rate.

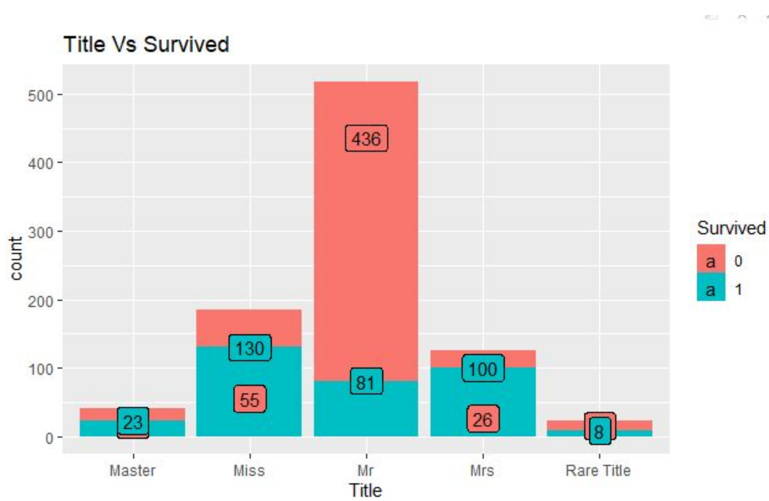
(5) Survived vs Embarked

Levels: C Q S (C = Cherbourg; Q = Queenstown, S = Southampton)

	0	1
C	0.441	0.559
Q	0.610	0.390
S	0.663	0.337

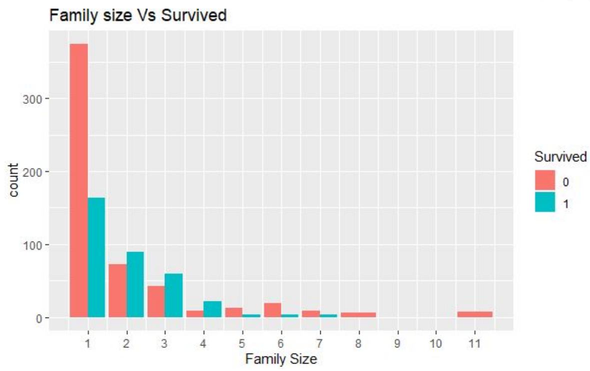
The probability of survival decreases in the order as Embarked port from C > Q > S.

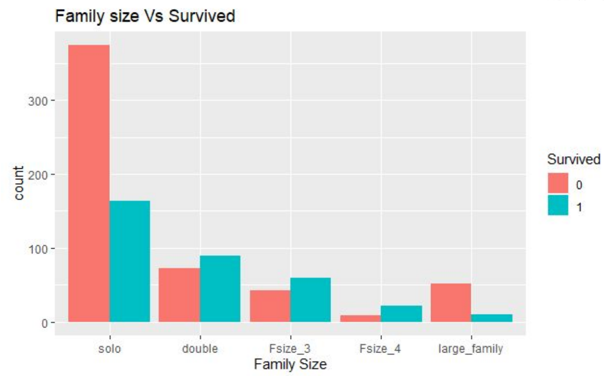
(6) Title Vs Survived



This seems correlated to sex variable because some of the most common titles represent sex. The intuition is reflected through the above graph that shows a higher survival rate for female-related title holders and lower rate for male-related title holders.

(7) Family Size (raw count) and aggregated family size (factor)





Family size is an aggregation of *sibsp(# of siblings / spouses)*and *parch(# of children/parents)*. Passengers with family size 1 and large family had a higher rate of death than survival. The graphs show that comparing the family sizes, the survival rate decreases as the number of family members increases.

Model Building

From EDA, we find five covariates, *title(T), Fsize(F), age(A), sex(S) and pclass(P)* can affect *survival.* We decide to not use *title* in model building because it is high related to *sex*. Simple linear logistic regression models were created with the *survival* response variables and above potential predictors. Because the purpose of this project is to build a logistic regression model(binary), in order to make the variable "Fsize" to be more useful and easier to deal with, we convert it to factor. We tried SLLR with all possible two-way interaction terms to find which interaction is useful. We decide to drop some meaningless models when coding, like three-way and higher interaction models.

Below are meaningful models we built:

Model	AIC	Resid. Dev	Df	ANOVA Chi Square test (use <0.05=T)
F+A+S+P	789.85	771.85	882	T,F,T,T
F+S+P	805.51	789.51	883	T,T,T
A+P+S+F+A:P+A:S+A:F +P:S+P:F+S:F	774.14	714.14	861	F,T,T,T,F,T,F,T,F,F
P+S+F+A:S+P:S	765.84	741.84	880	T,T,T,T,T
P+S+F+P:S	782.74	762.74	881	T,T,T,T

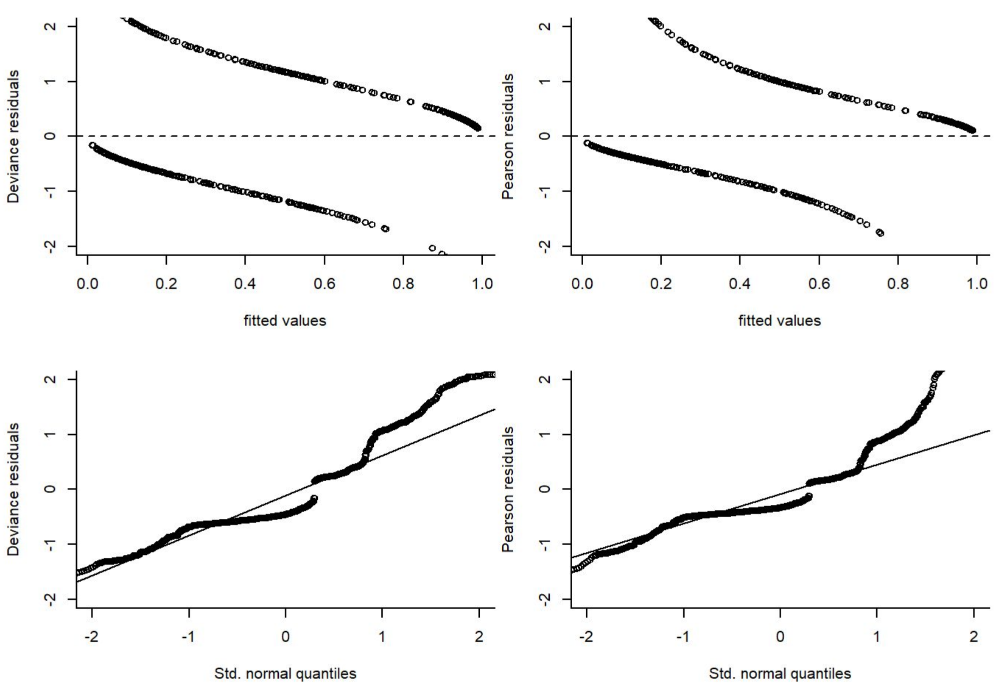
In the building process we find that *age* is not significantly useful to predict, but it is still worth putting in interaction terms.

When we add too many interaction terms, it doesn't show obvious improvement. I think the reason is because there might exist a high correlation between those variables, so not too much variance can be explained by adding new terms.

Model selection

The model *survival~age:sex+pcalss*sex+Fsize* has been chosen as our final model because it has the lowest AIC value and it is not too complex. The Chi-squared test shows that each predictor we added is useful.

Model diagnostics



It seems that the residual doesn't work very well in this case, and that's because the number of successes(survival), mi = 1, for each person(i) is one, which is like Bernoulli distribution, so that's kind of misleading.

Conclusion

Based on a sequence of plot drawed at EDA section, several conclusion we can make for answer our proposed scientific questions:

1) The people who are in range between 20 to 50 are less likely being survived than the elder and juvenile.
2) Female is more like being survived than male on average.
3) We see a negative trend between the *Pclass* variable and number of people being survived, and this trend is more obvious in the female group than male.
4) The plot for *Title* Vs *Survived* shows that the probability of being survived also correlated with social status. Specifically, the title with "Mr" has the lowest survival rate, and the title with "Master" and "Mrs" appears to have a higher survival rate.
5) Last, by comparing the number of people being survived in various family sizes, we observed a family with size between 2 and 4 has a particularly high chance of being survived.

With the comparision of univariate logistic regression model for each variable, we identified Age, Sex, and Title has reasonably low AIC and most important features to include. In order to build a more appropriate logistic regression model, we considered to add interaction terms into the model, and we noticed the interation between Age and Sex and Pclass and Sex are more interesting to discover. At the end, the model Survived~P+S+F+A:S+P:S with AIC 765.84was chosen for our final result, which is the lowest AIC so far.

For future analysis, we consider using PCA to reduce the amount of features. The model has its limitations, so applying the PCA technique to pick the most important variables may help us get a more accurate predictive model. What's more, we already splitted the dataset and have a test set this time. In the future, we are interested in using some methods like cross validation to do a better prediction.

Appendix

(1) SLLR on Survived~Age + Sex + Pclass + Fsize + Title

```
Survived_model1 <- glm(train$Survived ~ train$Age + train$Pclass + train$Sex + train$Fsize +
train$Title, family=binomial)
summary(Survived_model1)
anova(Survived_model1, test="Chisq")
```

```
Call:
glm(formula = train$Survived ~ train$Age + train$Pclass + train$Sex +
    train$Fsize + train$Title, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6607  -0.5302  -0.3939   0.5434   2.4461

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             19.01600  506.93895   0.038   0.9701
train$Age               -0.01922    0.00831  -2.314   0.0207 *
train$PclassClass_2     -1.41522    0.29277  -4.834 1.34e-06 ***
train$PclassClass_3     -2.33562    0.26938  -8.670  < 2e-16 ***
train$Sexmale          -15.23888  506.93863  -0.030   0.9760
train$FsizeFsize_3       0.14699    0.35603   0.413   0.6797
train$FsizeFsize_4       0.28286    0.59088   0.479   0.6322
train$Fsizelarge_family -2.59556    0.47902  -5.418 6.01e-08 ***
train$Fsizesolo          0.30909    0.27081   1.141   0.2537
train$TitleMiss        -15.81685  506.93890  -0.031   0.9751
train$TitleMr           -3.63345    0.55676  -6.526 6.75e-11 ***
train$TitleMrs         -15.30806  506.93896  -0.030   0.9759
train$TitleRare Title   -3.80412    0.78759  -4.830 1.36e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  719.85  on 878  degrees of freedom
AIC: 745.85

Number of Fisher Scoring iterations: 13

Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)


             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                           890    1186.66
train$Age     1    1.973       889    1184.68    0.1601
train$Pclass  2  134.213       887    1050.47 < 2.2e-16 ***
train$Sex     1  240.985       886     809.48 < 2.2e-16 ***
train$Fsize   4   37.632       882     771.85 1.335e-07 ***
train$Title   4   52.006       878     719.85 1.375e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(2) SLLR on Survived~Age + Sex + Pclass + Fsize

```
Call:
glm(formula = train$Survived ~ train$Age + train$Pclass + train$Sex +
    train$Fsize, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8597  -0.6133  -0.4209   0.5701   2.6814

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
(Intercept)              3.593036   0.403535   8.904  < 2e-16 ***
train$Age               -0.030438   0.007472  -4.074 4.63e-05 ***
train$PclassClass_2     -1.249814   0.269250  -4.642 3.45e-06 ***
train$PclassClass_3     -2.224786   0.252655  -8.806  < 2e-16 ***
train$Sexmale           -2.781246   0.202934 -13.705  < 2e-16 ***
train$FsizeFsize_3       0.570215   0.330337   1.726   0.0843 .
train$FsizeFsize_4       0.539983   0.550145   0.982   0.3263
train$Fsizelarge_family -2.062837   0.459631  -4.488 7.19e-06 ***
train$Fsizesolo          0.007927   0.242339   0.033   0.9739
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  771.85  on 882  degrees of freedom
AIC: 789.85

Number of Fisher Scoring iterations: 5

Analysis of Deviance Table

Model: binomial, link: logit


Response: train$Survived

Terms added sequentially (first to last)


             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                           890    1186.66
train$Age     1    1.973       889    1184.68    0.1601
train$Pclass  2  134.213       887    1050.47 < 2.2e-16 ***
train$Sex     1  240.985       886     809.48 < 2.2e-16 ***
train$Fsize   4   37.632       882     771.85 1.335e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

（3）SLLR on Survived ~ Sex + Pclass + Fsize

```
Call:
glm(formula = train$Survived ~ train$Pclass + train$Sex + train$Fsize,
    family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5756  -0.6599  -0.4494   0.6468   2.8450

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)               2.4223     0.2658   9.113  < 2e-16 ***
train$PclassClass_2      -0.9642     0.2539  -3.797 0.000147 ***
train$PclassClass_3      -1.7924     0.2216  -8.089 6.00e-16 ***
train$Sexmale            -2.7794     0.2003 -13.878  < 2e-16 ***
train$FsizeFsize_3        0.6668     0.3262   2.044 0.040936 *
train$FsizeFsize_4        0.8578     0.5449   1.574 0.115442
train$Fsizelarge_family  -1.8798     0.4451  -4.223 2.41e-05 ***
train$Fsizesolo          -0.0925     0.2380  -0.389 0.697482
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  789.51  on 883  degrees of freedom
AIC: 805.51

Number of Fisher Scoring iterations: 5

Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)


             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                           890    1186.66
train$Pclass  2  103.547       888    1083.11 < 2.2e-16 ***
train$Sex     1  256.220       887     826.89 < 2.2e-16 ***
train$Fsize   4   37.377       883     789.51 1.506e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(4) SLLP with all two-way interaction terms (Survive ~ Age * Sex + Age * Pclass + Age * Fsize + Sex* Pclass + Sex* Fsize + Pclass * Fsize)

```
Call:
glm(formula = train$Survived ~ train$Age + train$Pclass + train$Sex +
    train$Fsize + train$Age:train$Pclass + train$Age:train$Sex +
    train$Age:train$Fsize + train$Pclass:train$Sex + train$Pclass:train$Fsize +
    train$Sex:train$Fsize, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7492  -0.5419  -0.4524   0.3625   2.7276

Coefficients:
                                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                                   4.978606   1.316716   3.781 0.000156 ***
train$Age                                    -0.033665   0.026719  -1.260 0.207668
train$PclassClass_2                          -0.706102   1.426927  -0.495 0.620712
train$PclassClass_3                          -3.918495   1.090764  -3.592 0.000328 ***
train$Sexmale                                -3.462717   1.070374  -3.235 0.001216 **
train$FsizeFsize_3                            1.037522   1.725100   0.601 0.547556
train$FsizeFsize_4                           -2.419446   1.759260  -1.375 0.169050
train$Fsizelarge_family                      -1.935242   2.146986  -0.901 0.367388
train$Fsizesolo                              -1.552118   1.191814  -1.302 0.192808
train$Age:train$PclassClass_2                -0.034001   0.028617  -1.188 0.234781
train$Age:train$PclassClass_3                -0.002901   0.020012  -0.145 0.884751
train$Age:train$Sexmale                      -0.016062   0.020122  -0.798 0.424735
train$Age:train$FsizeFsize_3                 -0.028801   0.030756  -0.936 0.349049
train$Age:train$FsizeFsize_4                  0.011524   0.040857   0.282 0.777899
train$Age:train$Fsizelarge_family             0.011288   0.043308   0.261 0.794359
train$Age:train$Fsizesolo                     0.040254   0.023758   1.694 0.090193 .
train$PclassClass_2:train$Sexmale            -0.821525   0.911811  -0.901 0.367598
train$PclassClass_3:train$Sexmale             1.807888   0.774316   2.335 0.019553 *
train$PclassClass_2:train$FsizeFsize_3       -0.102168   1.309473  -0.078 0.937811
train$PclassClass_3:train$FsizeFsize_3       -0.367467   1.163729  -0.316 0.752179
train$PclassClass_2:train$FsizeFsize_4        2.168045   1.565036   1.385 0.165961
train$PclassClass_3:train$FsizeFsize_4        2.570120   1.440219   1.785 0.074337 .
train$PclassClass_2:train$Fsizelarge_family  14.653081 594.913749   0.025 0.980350

train$PclassClass_3:train$Fsizelarge_family  -0.609512   1.768470  -0.345 0.730354
train$PclassClass_2:train$Fsizesolo           0.823437   0.929303   0.886 0.375574
train$PclassClass_3:train$Fsizesolo           0.809678   0.753351   1.075 0.282478
train$Sexmale:train$FsizeFsize_3              0.840142   0.784009   1.072 0.283901
train$Sexmale:train$FsizeFsize_4              1.413841   1.189573   1.189 0.234625
train$Sexmale:train$Fsizelarge_family        -0.030308   1.370882  -0.022 0.982362
train$Sexmale:train$Fsizesolo                -0.259587   0.616409  -0.421 0.673662
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  714.14  on 861  degrees of freedom
AIC: 774.14

Number of Fisher Scoring iterations: 13

Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)
```

```
                           Df Deviance Resid. Df Resid. Dev  Pr(>Chi)    
NULL                                         890    1186.66              
train$Age                   1    1.973       889    1184.68  0.160083    
train$Pclass                2  134.213       887    1050.47 < 2.2e-16 ***
train$Sex                   1  240.985       886     809.48 < 2.2e-16 ***
train$Fsize                 4   37.632       882     771.85 1.335e-07 ***
train$Age:train$Pclass      2    1.614       880     770.24  0.446146    
train$Age:train$Sex         1    8.878       879     761.36  0.002887 ** 
train$Age:train$Fsize       4    8.154       875     753.21  0.086093 .  
train$Pclass:train$Sex      2   25.315       873     727.89 3.184e-06 ***
train$Pclass:train$Fsize    8    9.555       865     718.34  0.297685    
train$Sex:train$Fsize       4    4.197       861     714.14  0.380044    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(5) SLLP on Survived ~ Age + Pclass + Sex + Fsize + Age : Sex + Pclass : Sex

```
Call:
glm(formula = train$Survived ~ train$Age + train$Pclass + train$Sex + 
    train$Fsize + train$Age:train$Sex + train$Pclass:train$Sex, 
    family = binomial)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.0774  -0.6048  -0.4540   0.3800   2.5521  

Coefficients:
                                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)                        4.22875    0.78134   5.412 6.23e-08 ***
train$Age                         -0.02004    0.01318  -1.520 0.128464    
train$PclassClass_2               -1.17766    0.74038  -1.591 0.111698    
train$PclassClass_3               -3.56980    0.64412  -5.542 2.99e-08 ***
train$Sexmale                     -3.21621    0.87395  -3.680 0.000233 ***
train$FsizeFsize_3                 0.64916    0.33873   1.916 0.055305 .  
train$FsizeFsize_4                 0.53755    0.57475   0.935 0.349641    
train$Fsizelarge_family           -1.96184    0.48538  -4.042 5.30e-05 ***
train$Fsizesolo                    0.01194    0.25450   0.047 0.962592    
train$Age:train$Sexmale           -0.02031    0.01617  -1.256 0.209261    
train$PclassClass_2:train$Sexmale -0.56239    0.82379  -0.683 0.494803    
train$PclassClass_3:train$Sexmale  1.75633    0.70433   2.494 0.012645 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  741.84  on 879  degrees of freedom
AIC: 765.84

Number of Fisher Scoring iterations: 6
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)


                     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)    
NULL                                   890    1186.66              
train$Age             1    1.973       889    1184.68   0.16008    
train$Pclass          2  134.213       887    1050.47 < 2.2e-16 ***
train$Sex             1  240.985       886     809.48 < 2.2e-16 ***
train$Fsize           4   37.632       882     771.85 1.335e-07 ***
train$Age:train$Sex   1    7.969       881     763.88   0.00476 ** 
train$Pclass:train$Sex  2   22.043       879     741.84 1.634e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(6) SLLP on Survived ~ Pclass + Sex + Fsize + Pclass : Sex

```
Call:
glm(formula = train$Survived ~ train$Pclass + train$Sex + train$Fsize + 
    train$Pclass:train$Sex, family = binomial)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.9883  -0.5558  -0.5314   0.4104   2.6932  

Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)    
(Intercept)                        3.5450     0.6153   5.761 8.35e-09 ***
train$PclassClass_2               -1.1130     0.7369  -1.510 0.130925    
train$PclassClass_3               -3.3409     0.6218  -5.373 7.74e-08 ***
train$Sexmale                     -4.0836     0.6294  -6.488 8.70e-11 ***
train$FsizeFsize_3                 0.7308     0.3331   2.194 0.028232 *  
train$FsizeFsize_4                 0.9084     0.5678   1.600 0.109655    
train$Fsizelarge_family           -1.8321     0.4843  -3.783 0.000155 ***
train$Fsizesolo                   -0.1186     0.2487  -0.477 0.633609    
train$PclassClass_2:train$Sexmale -0.1379     0.8061  -0.171 0.864174    
train$PclassClass_3:train$Sexmale  2.1118     0.6685   3.159 0.001583 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  762.74  on 881  degrees of freedom
AIC: 782.74
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)


                        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)    
NULL                                      890    1186.66              
train$Pclass             2  103.547       888    1083.11 < 2.2e-16 ***
train$Sex                1  256.220       887     826.89 < 2.2e-16 ***
train$Fsize              4   37.377       883     789.51 1.506e-07 ***
train$Pclass:train$Sex   2   26.769       881     762.74 1.539e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(7) SLLP on Survived ~ Pclass + Sex + Fsize + Age: Sex +Pclass : Sex

```
Call:
glm(formula = train$Survived ~ train$Pclass + train$Sex + train$Fsize + 
    train$Age:train$Sex + train$Pclass:train$Sex, family = binomial)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.0774  -0.6048  -0.4540   0.3800   2.5521  

Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)    
(Intercept)                      4.228749   0.781338   5.412 6.23e-08 ***
train$PclassClass_2             -1.177659   0.740383  -1.591 0.111698    
train$PclassClass_3             -3.569802   0.644120  -5.542 2.99e-08 ***
train$Sexmale                   -3.216213   0.873950  -3.680 0.000233 ***
train$FsizeFsize_3               0.649161   0.338726   1.916 0.055305 .  
train$FsizeFsize_4               0.537553   0.574747   0.935 0.349641    
train$Fsizelarge_family         -1.961841   0.485376  -4.042 5.30e-05 ***
train$Fsizesolo                  0.011936   0.254497   0.047 0.962592    
train$Sexfemale:train$Age       -0.020043   0.013184  -1.520 0.128464    
train$Sexmale:train$Age         -0.040351   0.009788  -4.123 3.75e-05 ***
train$PclassClass_2:train$Sexmale -0.562392   0.823788  -0.683 0.494803    
train$PclassClass_3:train$Sexmale  1.756333   0.704332   2.494 0.012645 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 1186.66  on 890  degrees of freedom
Residual deviance:  741.84  on 879  degrees of freedom
AIC: 765.84

Number of Fisher Scoring iterations: 6

Analysis of Deviance Table

Model: binomial, link: logit

Response: train$Survived

Terms added sequentially (first to last)


                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)    
NULL                                     890    1186.66              
train$Pclass            2  103.547       888    1083.11 < 2.2e-16 ***
train$Sex               1  256.220       887     826.89 < 2.2e-16 ***
train$Fsize             4   37.377       883     789.51 1.506e-07 ***
train$Sex:train$Age     2   25.628       881     763.88 2.722e-06 ***
train$Pclass:train$Sex  2   22.043       879     741.84 1.634e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Code：

```
## 1. Importing library, defining function, and checking data

```{r setup, include=FALSE}

# Basic package

library(tidyverse)

library(broom)

library(readr)

library(ggplot2)

# Advanced(Fancy) package
```
```

library(ggthemes)

library(gridExtra)

This allows you to set default behavior for R chunks

knitr::opts_chunk$set(echo = TRUE)

```

### 1.1 Import dataset:

```{r include=F, echo=F}

set.seed(1)

train <- read.csv("train.csv", stringsAsFactors = F)

test <- read.csv('test.csv', stringsAsFactors = F)

full  <- bind_rows(train, test) # bind training & test data

```

1.2 Checking dataset

```{r}

str(train)

str(test)

str(full)

```
```

1.3 Background description:

`![Titanic_sinking.jpg](Titanic_sinking.jpg)`

[Reference](https://www.kaggle.com/c/titanic)

2. EDA

2.1 Age vs Survived

```{r}
# Age vs Survived
ggplot(full[1:891,], aes(Age, fill = factor(Survived))) +
  theme_few()+
  geom_histogram(bins=30) +
  ggtitle("Age vs Survived") +
  scale_fill_discrete(name = "Survived") # For the label in the right
```

Note: 1)People who are in range between 20 to 50 are less likely being survived, and the people who below age 10 are more likely being survived. 2)The distribution of age looks like a right skewed distribution.

The warning tell us, there are 177 missing value. So, let's imputing those missing age values. There are many way to take care the missing value, such as remove the data, or filling with zero or NaN. But, we can do better than that. A decent way to do this is to create a model that predicts the average ages based on other variables. There are many package can do this interpolation, such as rpart(recursive partitioning for regression), and mice(Multivariate Imputation by Chained Equations). [reference for mice](http://www.jstatsoft.org/article/view/v045i03/v45i03.pdf). Let's try the mice library:

````
```{r}

library('mice') # imputation

# Show number of missing Age values in training set

sum(is.na(full[1:891,]$Age)) # => 177


# Make variables factors into factors

# factor_vars <- c('PassengerId','Pclass','Sex','Embarked', 'Surname','Family')

#

# full[factor_vars] <- lapply(full[factor_vars], function(x) as.factor(x))


mice_mod <- mice(full[, !names(full) %in%
c('PassengerId','Name','Ticket','Cabin','Family','Surname','Survived')], method='rf')


mice_output <- complete(mice_mod)
````
````

```
# Plot age distributions

par(mfrow=c(1,2))

hist(full$Age, freq=F, main='Age: Original Data',

  col='darkgreen', ylim=c(0,0.04))

hist(mice_output$Age, freq=F, main='Age: MICE Output',

  col='lightgreen', ylim=c(0,0.04))
```

The result look pretty good, so let’s replace our age vector in the original data with the output from the mice model.

```{r}
# Replace Age variable from the mice model.

full$Age <- mice_output$Age


# Show new number of missing Age values

sum(is.na(full$Age))
```

Now, the missing value is gone!

2.2 Sex Vs Survive

```{r}
# Sex vs Survived

ggplot(full[1:891,], aes(Sex, fill = factor(Survived))) +
```

```
  geom_bar(stat = "count", position = 'dodge')+
  xlab("Sex") +
  ggtitle("Sex vs Survived") +
  geom_label(stat='count',aes(label=..count..))+
  scale_fill_discrete(name = "Survived") # For the label in the right
```

Note: 1)female is more likely being survived than male, femail survived rate roughly 75%, and male is roughly 16.7%, so almost 5 time greater!

2.3 Age Vs Sex Vs Survived

```{r}
#Sex vs Survived vs Age
ggplot(full[1:891,], aes(Age, fill = factor(Survived))) +
  geom_histogram(bins=30) + # bins: controls the width of bar, so larger the thinner. You can use geom_bar() if you don't want to specify it!
  xlab("Age") +
  facet_grid(.~Sex)+
  ggtitle("Age vs Sex vs Survived") +
  # geom_label(stat='count',aes(label=..count..)) +
  scale_fill_discrete(name = "Survived") # For the label in the right

```

Note: 1) Again, female is more likely being survived than male. 2) The differences between the number of peoples survived and not survived is largest at roughly age 20to30, and this is true for both female and male.

2.4. Pclass vs Sex

```{r}
# geom_bar vs geom_hist:
#   - Bar charts provide a visual presentation of categorical data
#   - Histograms are used to plot the distribution of data

# Pclass vs Sex Vs Survived
p1 <- ggplot(full[1:891,], aes(Pclass, fill = factor(Survived))) +
  geom_bar(stat='count') +
  xlab("Pclass") +
  facet_grid(.~Sex)+
  ggtitle("Pclass vs Sex vs Survived") +
  geom_label(stat='count',aes(label=..count..)) +
  scale_fill_discrete(name = "Survived") # For the label in the right

p2 <- ggplot(full[1:891,], aes(x = Pclass, fill = factor(Survived))) +
  geom_bar(stat='count', position='fill') +
  labs(x = 'Passenger class', y= "Percent") +
```

  facet_grid(.~Sex) +
  theme(legend.position="none")
grid.arrange(p1, p2, ncol=2)
```

Note: 1) female is more likely of being survived than male in average. 2)In female group, majority passengers in class 1 and class 2 are survived, and more people in classed died. However, in male group, the survived rate in class 2 (~18.7%) just as bad as class 3(~15.67%).

### 2.5 Pclass Vs Embarked

Let's removed the missing value at first

```{r}
full[c(62, 830), 'Embarked']

# Get rid of our missing passenger IDs
embark_fare <- full %>%
  filter(PassengerId != 62 & PassengerId != 830)

# Use ggplot2 to visualize embarkment, passenger class, & median fare
ggplot(embark_fare, aes(x = Embarked, y = Fare, fill = factor(Pclass))) +
  geom_boxplot() +
  geom_hline(aes(yintercept=80),
  colour='red', linetype='dashed', lwd=2) +

  theme_few()

```

Notice that missing value in the marning message:

```{r}

# Since their fare was $80 for 1st class, they most likely embarked from 'C'

full$Embarked[c(62, 830)] <- 'C'

# Replace missing fare value with median fare for class/embarkment

full$Fare[1044] <- median(full[full$Pclass == '3' & full$Embarked == 'S', ]$Fare, na.rm = TRUE)

```

Replace their embarkment with "C"

```{r}

train[c(62, 830), 'Embarked']   # => [1] "" ""

# Let's delete them

# train <- train %>%  filter(PassengerId != 62 & PassengerId != 830)

# Instead of delete them it's better to replace their embarkment with "C", since there fare was $80 for 1st class.

train$Embarked[c(62, 830)] <- c("C", "C")

# show the table of counts

count_table <- table(train$Embarked, train$Survived)

count_table

```

```{r}

round(count_table / apply(count_table, 1, sum), 3)

```

3. Processing data and Further EDA

Notices, there are some useful infromation in passenger name, what is it? For example: the passenger title!(e.g. Ms, Miss, Mrs..) So, we can use this information to ask some question like, is there any relationship between the passenger title and probability of survived? Also, The surname can be useful as well. It allow us to use "surname" to represent a families.

Now, let's create a new variables, called title.

3.1 Feature Engineer work:

```{r}

# Grab title from passenger names

full$Title <- gsub('(.*, )|(\\..*)', '', full$Name)


cat("Show title counts by sex:")

table(full$Sex, full$Title)

# kable(table(full$Sex, full$Title)) # A fance table for html presentation


# Titles with very low cell counts to be combined to "rare" level

rare_title <- c('Dona', 'Lady', 'the Countess','Capt', 'Col', 'Don',

                'Dr', 'Major', 'Rev', 'Sir', 'Jonkheer')

# Also reassign mlle, ms, and mme accordingly

full$Title[full$Title == 'Mlle']        <- 'Miss'

full$Title[full$Title == 'Ms']          <- 'Miss'

full$Title[full$Title == 'Mme']         <- 'Mrs'

full$Title[full$Title %in% rare_title]  <- 'Rare Title'

cat("\nShow title counts by sex after merged title has very few count in the data: ")

table(full$Sex, full$Title)

# Finally, grab surname from passenger name

full$Surname <- sapply(full$Name,

                      function(x) strsplit(x, split = '[,.]')[[1]][1])

cat("\nHead of full$Surname:\n")

head(full$Surname)

```

3.2 Title Vs Survived

```{r}

ggplot(full[1:891,], aes(x = Title, fill = factor(Survived))) +

```
  geom_bar(stat='count', position='stack') +

  ggtitle("Title Vs Survived") +

  geom_label(stat='count',aes(label=..count..)) +

  scale_fill_discrete(name = "Survived") # For the label in the right
```

Note: we see the Mr. "Mr" are died pretty badly, which proved our previous observation that male are less likely being survived than female.

3.3 Family size Vs Survived

Family size might be a interesting predictor for evaluating the probabilty of being survived. So, let's use the sum of "sibsp" and "parch" to create another new variable, and then we can analysis there relationship!

Create variable Fsize, which is sum of the number of siblings/spouses and number of chldren/parens and one(The person himself)

```{r}
# Create a family size variable including the passenger themselves

full$Fsize <- full$SibSp + full$Parch + 1


# Create a family variable

full$Family <- paste(full$Surname, full$Fsize, sep='_')

head(full$Family)
```

```{r}

# Use ggplot2 to visualize the relationship between family size & survival

ggplot(full[1:891,], aes(x = Fsize, fill = factor(Survived))) +

  geom_bar(stat='count', position='dodge') +

  scale_x_continuous(breaks=c(1:11)) +

  ggtitle("Family size Vs Survived") +

  labs(x = 'Family Size') +

  # geom_label(stat='count',aes(label=..count..)) +

  scale_fill_discrete(name = "Survived") # For the label in the right


```

Note: By comparing the "family size" and "Survived", we noticed the singleton, familes sizes 1, and large families (size > 5) are less likely being survived than the family with size between 2 and 4. Keep this in mind, that might be something we want to use in building our regression model.

4. Building SLLR model:

Produce a table including the pclass factor variable, number of passenger survived(survival=1) in each class, and the total number of passenger in each class(survival=1 or 0)

Response variable:

- survived (0 == died, 1 == survived)

Explanatory variables/covaraite of interest:

- Pclass

- Sex

- Age

- Fsize

Define the function to be used:

```{r}

## Define the logit function.

logit <- function (p)

{

  log(p / (1 - p))

}


## Define the inverse logit function.

sigmoid <- function (etas)

{

  exp(etas) / (1 + exp(etas))

}
```

```

### 4.1 Redefined the variable to factor:

Because the purpose of this project is to build a logistic regression model(binary), in order to make the varaible "Fsize" to be more useful and easier to deal with, we need to convert it to factor! (as well as other categorical variables. Factor just a nice data type in R, that is design for categorical variable.)

```{r}

set.seed(1)

train <- full[1:891,]

## define 'Survived' to be 1 if any passenger survived; 0 if died

train$Survived <- as.numeric(train$Survived == 1)

## define the variable 'Sex'

## is 0 if Sex is light medium or medium.

## is 1 if color is dark medium or dark.

# Sex <- factor(ifelse(crabs$color <= 2, "not dark", "dark"))

train$Sex <- factor(train$Sex)

```
Redefine the Pclass factor variable ordered as
1: Class_1, 2: Class_2, 3: Class_3, Otherwise: Error.
(factors by default are ordered alphabetically)
train$Pclass <-
  factor(ifelse(train$Pclass==1, "Class_1",
               ifelse(train$Pclass==2, "Class_2",
                      ifelse(train$Pclass==3, "Class_3", "Error"))),
         levels=c("Class_1", "Class_2", "Class_3"))

Redefine the Fsize factor variable ordered as
1: solo, 2: double, 3: Fsize_3, 4: Fsize_4, 5: large_family
(factors by default are ordered alphabetically)
train$Fsize <-
factor(ifelse(train$Fsize==1, "solo",
ifelse(train$Fsize==2, "double",
ifelse(train$Fsize==3, "Fsize_3",
ifelse(train$Fsize==4, "Fsize_4", "large_family")))),
levels=c("solo", "double", "Fsize_3", "Fsize_4", "large_family"))

Alternatively:
train$Fsize[train$Fsize>=5] <- 'large_family' # THis must go first, otherwise it won't work
```

````
train$Fsize[train$Fsize==1] <- 'solo'

train$Fsize[train$Fsize==2] <- 'double'

train$Fsize[train$Fsize==3] <- 'Fsize_3'

train$Fsize[train$Fsize==4] <- 'Fsize_4'

train$Fsize <- as.factor(train$Fsize)

# levels(train$Fsize) <- c("solo","double","Fsize_3","Fsize_4","large_family")


factor_vars <- c('Pclass','Sex','Embarked', 'Title','Surname','Family','Fsize')


train[factor_vars] <- lapply(train[factor_vars], function(x) as.factor(x))

```




```{r}

# Fsize Vs Survived

ggplot(train[!is.na(full$Survived),], aes(x = Fsize, fill = factor(Survived))) +

  geom_bar(stat='count', position='dodge') +

  ggtitle("Family size Vs Survived") +

  labs(x = 'Family Size') +

  # geom_label(stat='count',aes(label=..count..)) +

  scale_x_discrete (limits = c('solo', 'double', 'Fsize_3', 'Fsize_4', 'large_family')) +

  scale_fill_discrete(name = "Survived") # For the label in the right
````

````
```

```{r eval=FALSE, include=FALSE}
rounded_age <- round(train$Age*2)/2
count_table <- table(rounded_age, train$Survived)
prob_survived <- round(count_table / apply(count_table, 1, sum), 3)[,2]
```

```{r eval=FALSE, include=FALSE}
plot(sort(unique(rounded_age)), logit(prob_survived),
    xlab="weight (to nearest 0.5kg)", ylab="proportion")
```

```{r}
# show the table of counts
count_table <- table(train$Sex, train$Survived)
count_table
```

```{r}
# show the table of proportions: p_ij = r_ij/(r_i + r_j):
round(count_table / apply(count_table, 1, sum), 3)
````

```

En, looks like female has a high probability of being survived!

```{r}

# fit the glm with Sex:

Survived_sex_model <- glm(train$Survived ~ train$Sex, family=binomial)

summary(Survived_sex_model)

anova(Survived_sex_model, test="Chisq")

```

p-value(>|Z|) tells us, the coefficient for age is sig different from zero, and the expected probability of being survived for male is $e^{-2.5137} * 100% = 8.097% $ less than the female in average. The pr(>Chi) tells us it's useful to include sex into our model, which reduced the AIC from 1186.7 to 917.8.

4.4 SLLR on Survived~Pclass

```{r}

# show the table of counts

count_table <- table(train$Pclass, train$Survived)

count_table

```
```

```{r}

# show the table of proportions: p_ij = r_ij/(r_i + r_j):

round(count_table / apply(count_table, 1, sum), 3)

```

En, looks like the passenger classes does related to the probability of survived, the higher the classes and low the probability of being survived.

```{r}

# show the table of counts

count_table <- table(train$Fsize, train$Survived)

count_table

```

```{r}

# show the table of proportions: p_ij = r_ij/(r_i + r_j):

round(count_table / apply(count_table, 1, sum), 3)

```

En, seems like the probability of survived is increased and then decreased for the familiy size over 4, so it might not be a linear relationship.

```{r}
```

show the table of counts

count_table <- table(train$Title, train$Survived)

count_table

```

```{r}

# show the table of proportions: p_ij = r_ij/(r_i + r_j):

round(count_table / apply(count_table, 1, sum), 3)

```

En, seems like the probability of survived is increased and then decreased for the familiy size over 4, so it might not be a linear relationship.

Selecting Models

```{r echo=TRUE}

Survived_model1 <- glm(train$Survived ~ train$Age + train$Pclass + train$Sex + train$Fsize, family=binomial)

summary(Survived_model1)

anova(Survived_model1, test="Chisq")

```

```{r}

Survived_model2 <- glm(train$Survived ~ train$Pclass + train$Sex + train$Fsize, family=binomial)

summary(Survived_model2)

anova(Survived_model2, test="Chisq")

```

```{r}

Survived_model3 <- glm(train$Survived ~ train$Age + train$Pclass + train$Sex + train$Fsize + train$Age : train$Pclass + train$Age : train$Sex + train$Age : train$Fsize + train$Pclass : train$Sex + train$Pclass : train$Fsize + train$Sex : train$Fsize, family=binomial)

summary(Survived_model3)

anova(Survived_model3, test="Chisq")

```

```{r}

Survived_model4 <- glm(train$Survived ~ train$Age + train$Pclass + train$Sex + train$Fsize + train$Age : train$Sex + train$Pclass : train$Sex, family=binomial)

summary(Survived_model4)

anova(Survived_model4, test="Chisq")

```
```

```{r}

Survived_model5 <- glm(train$Survived ~ train$Pclass + train$Sex + train$Fsize + train$Pclass : train$Sex, family=binomial)

summary(Survived_model5)

anova(Survived_model5, test="Chisq")

```

```{r}

Survived_model6 <- glm(train$Survived ~ train$Pclass + train$Sex + train$Fsize + train$Age : train$Sex + train$Pclass : train$Sex, family=binomial)

summary(Survived_model6)

anova(Survived_model6, test="Chisq")

```

With some interaction terms, we notice that the Deviance Residual had dicreased, but not substantially.

I think the reason is because that there might exist a very high correlation between those variables, so not too much variance can be explained by adding new terms. Also we can see that the degree of freedom is pretty big here(over 800), so if we applied the PCA technique to reduced the amount of features/variables and then picked several most important component as our representative variables, it's possible that we could get a better predictive model!

5. Model diagnostic

````
```{r}

## produce the default diagnostic plots

par(mfrow=c(2,2))

plot(Survived_model4)



## calculate the fitted values.

fits  <- fitted(Survived_model4)


## calculate the deviance residuals

dev.resids  <- resid(Survived_model4)

pear.resids <- as.numeric(resid(Survived_model4, type="pearson"))



par(mfrow=c(2,2), cex=0.65, mar=c(4, 4, 2.3, 0.2), bty="L")


plot(fits, dev.resids,

    xlab="fitted values", ylab="Deviance residuals", ylim=c(-2,2))

abline(h=0, lty=2)


plot(fits, pear.resids,
````
````

```
     xlab="fitted values", ylab="Pearson residuals", ylim=c(-2,2))
abline(h=0, lty=2)


qqnorm(dev.resids,
       xlab="Std. normal quantiles", ylab="Deviance residuals", main="",
       xlim=c(-2,2), ylim=c(-2,2))
qqline(dev.resids)

qqnorm(pear.resids,
       xlab="Std. normal quantiles", ylab="Pearson residuals", main="",
       xlim=c(-2,2), ylim=c(-2,2))
qqline(pear.resids)
```

Haha, so the residual doesn't work very well in this case, and that's because the number of success(survived), m_i = 1, for each person(i) is one, which is like Bernoulli distribution, so that's kinda of missleading.

Now, there are only three things we should care about, 1)) Devidance residual, 2)df(degree of freedom), and 3)p-value (With Chi-square dist).

Let's use deviance table to help us figure out what is the best model here!

(Opt)8. Making Prediction

8.1 Modeling with Random Forest

```{r eval=FALSE, include=FALSE}

train <- full[1:891,]

test <- full[892:1309,]

# random forest

library('randomForest')

# Set a random seed

set.seed(754)

set.seed(123)
```

```
rf_model <- randomForest(factor(Survived) ~ Pclass + Sex + Fare + Embarked + Title + Fsize, data = train)
```