**Final Report**

**Brainstorm:**

After each group member obtaining idea about training data set, a brainstorming helped the group to decide which models could be utilized in this problem. First, the support vector machine (SVM) model seemed attractive as a result of the capability to deal with sample that could not be linear separated. The kernel functions in SVM embedded the low dimensional original space into a higher dimensional one. Therefore, the sample was not linearly separable in the original space would become separable in the higher dimension. Another reason made SVM a suitable candidate was SVM worked on two-class tasks, which exactly matched our desired outcome for this problem.

The second expected model was the random forest, since it was one of the most accurate learning algorithms right now. Based on the large data set in this problem, the random forest could be chosen to run efficiently on big databases. Also, it provided the importance of each feature in the classification, which could be very useful in our training sample that had many features included. The disadvantage for random forest would be overfitting and it will impact the accuracy of our test data.

The group also wanted to employ the logistic regression model because it was a binary classification. The process of manipulating the data, applying the models and a summary of our results would be elaborated in the following part of this report.

**Data preprocessing:**

As what we have seen from the training and testing data, we found that there are some features that are irrelevant to the possibility of survival. Also, some features has missing values which may affect our data processing. Besides, some new features that can be extracted from the old features, which may improve the accuracy of the solution.

To improve these: (1)We created a feature called “Tittle” which is extracted from the “Name” feature and make all the “Mme” into ”Mrs”, ”Mlle” into “Miss”, ”Don” into ”Sir”, ”Dona” & ”the Countess” & ”Jonkheer” into ”Lady”. (2) We also created a feature called “Family size” which has the formula: Family size= Sibsp+Parch+1. (3) We extracted the “surname” from the “Name” feature. (4)We put surname and family size together as “FamilyID” feature. For example, if someone has family size of 4 whose surname is Anderson, then his FamilyID is Aderson4. However, when the family size is less or equal to 3, then the FamilyID is “small”.

To deal with the missing values, we used separated methods. For the missing Age value, we used decision tree to reconstruct the missing values; for the missing Embarked value, we set the 6th and 830th value into “S”; for the missing Fare values, we set the 1044th value into the median value of all third class fares since the person purchased a third class ticket.

For the random forest model only, we build a new feature called”FamilyID2” which is the same as “FamilyID” except when Family size is less or equal to 4,then ”FamilyID2” is “small“.

**Feature selection:**

According to historical event, we started our feature selection progress with a hypothesis that people with following characteristics should have a higher chance to survive: female, higher pclass, higher social class (represented by name title). By checking the feature importance with correlogram, we verified our hypothesis. In addition, we found that age, family size and fare are also correlated with the chance to survive. However, we did not select fare as one of our features since it is positive correlated with pclass, in that case adding fare may cause overfitting problem. In the end, the features we selected are: sex, pclass, title, family size and age.

**Results of Classification Models:**

1. Random Forest:

First classification model we tested is the Random Forest. The number of trees to grow is set to 2000, since this number should not be set to too small to ensure that every input row gets predicted at least a few times. By fitting this model, we got a score of 0.8507 on train set data. The prediction on test set data which is submitted to Kaggle shows the score is 0.7751.

1. SVM:

Since we do not know the relationship between our features and label, SVM is a good method to fit higher dimension situation. The parameters we choose are: C=0, gamma=1. By fitting this model, we get a score of 0.8350 on train set data, and the prediction on test set data get an accurate rate at 78.95%.

1. Other models:

We also used models like Logistic Regression, Logistic Regression with xgboost (which is similar with adaboost), K-nearest neighbor with k=3,5,10,15. However, the scores we get from these models are all around 0.70 which are much lower than what we get from SVM.

**Conclusion:**

After submitting all our result to Kaggle, the SVM classification method get the highest score which is 0.7895. In result, we think that SVM would predict the survive status of Titanic passengers better, based on our experiment. And here is a link to our submissions, https://www.kaggle.com/c/titanic/submissions