Background

Traindata.csv contains 3220 records with 57 continuous features while testdata.csv contains 1380 records. Features 1 to 54 are non-negative number and most of them are 0. Features 55 to 57 are positive numbers ranging from 1 to 9000. In this classification problem, trainlabel.csv contains two classes either 0 or 1. Around 40% labels are 1.

Pre-processing

As most of the features are 0 and some of them are extreme large value, normalization is needed on both traindata and testdata. Standard score is used on all 57 features. $x_{new} = \frac{x - \text{mean}(x)}{\text{standard deviation}(x)}$.

To compare performance of different model, 85% of data is training set which is used on training/tuning model while 15% of data is validation set which is used comparing the accuracy on models.

Model

Two types of model are used in this problem: support vector machine and gradient boosted tree. For support vector machine, library e1071 is used as a library of svm in R while library xgboost is used as a library of gradient boosted tree. For svm, 4 types of kernel are used with hyper-parameter tuned by grid search:

Radial: gamma=0.03

$$e^{-\gamma|u-v|^2}$$

Polynomial: gamma=0.1, degree=2, coef0=100

$$(\gamma * u'v + coef0)^d$$

Linear

$$u'v$$

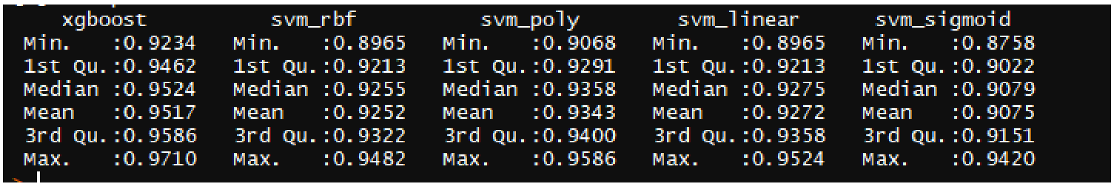
Sigmoid: gamma=0.005, coef0=0

$$\tanh(\gamma * u'v + coef0)$$

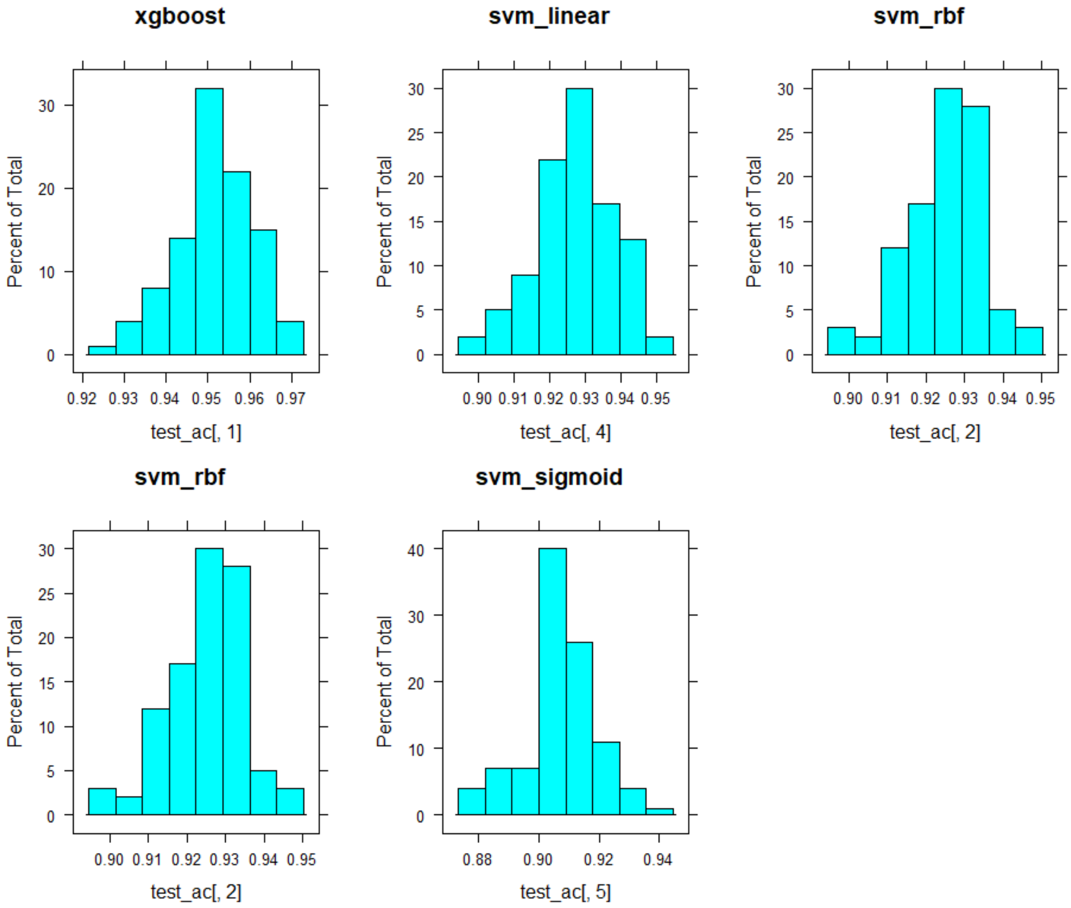
For gradient boosted tree model, log loss is used for evaluating metric in training model.$Logloss = -\frac{1}{n}\sum_{i=1}^{n}(y_i \log_e \hat{y}_i + (1 - y_i)\log_e(1 - \hat{y}_i))$, where y_i is the true label, $\hat{y}_i$ is the predicted probability. Log loss preforms better in model training because it considers the difference between true label and prediction. After tuning the hyper parameter of xgboost, following parameters give the best performance: maximum depth of a tree=5, step size shrinkage=0.03, number of rounds=500, L2 regularization =2

Result

As training data is not big enough, the random split between training and validation set may vary the model. Therefore 100 random splits are used to create 5x100 models and compare the mean accuracy.



```
    xgboost           svm_rbf           svm_poly          svm_linear        svm_sigmoid
 Min.   :0.9234   Min.   :0.8965   Min.   :0.9068   Min.   :0.8965   Min.   :0.8758
 1st Qu.:0.9462   1st Qu.:0.9213   1st Qu.:0.9291   1st Qu.:0.9213   1st Qu.:0.9022
 Median :0.9524   Median :0.9255   Median :0.9358   Median :0.9275   Median :0.9079
 Mean   :0.9517   Mean   :0.9252   Mean   :0.9343   Mean   :0.9272   Mean   :0.9075
 3rd Qu.:0.9586   3rd Qu.:0.9322   3rd Qu.:0.9400   3rd Qu.:0.9358   3rd Qu.:0.9151
 Max.   :0.9710   Max.   :0.9482   Max.   :0.9586   Max.   :0.9524   Max.   :0.9420
```



Xgboost gives the best performance, so the xgboost model is re-trained with all data to try to improve the accuracy for the final prediction.