

## Supplementary Material

### Codes for model training

```
library(caret) # "train" function is used for ML models
#1. Traditional LR
LR1 <- glm(Target~.,data=trainset, family="binomial")

#2. LR2 optimized with Lasso (L1) penalty
grid <- expand.grid( alpha=1 , # Lasso L1 penalty
                    lambda = 10^seq(-4, 1, length = 50) )
control <- trainControl(
  method = "cv", # cross-validation
  number = 5, # number of folds in cross-validation
  classProbs = TRUE,
  summaryFunction = twoClassSummary,
  verboseIter = TRUE,
  savePredictions = "final",
  returnResamp = "final" )
set.seed(1)
model <- train(Target ~ ., data = trainset,
  method = "glmnet",
  metric = "ROC",
  tuneGrid = grid,
  trControl = control )

#3. Random Forest
grid <- expand.grid( mtry = c(1, 2, 3, 4, 5) )
control<- trainControl(method = "cv", # cross-validation
  number = 5, # number of folds in cross-validation
  classProbs=TRUE,
  summaryFunction = twoClassSummary,
  verboseIter = TRUE,
  savePredictions = "final",
  returnResamp = "final")
set.seed(1)
model <- train(Target ~ ., data = trainset,
  method = "rf",
  trControl = control,
  tuneGrid = grid,
  metric = "ROC"
  ntree = 150 ) # the default ntree value is 500

#4. Neural networks
grid <- expand.grid(size = c(2, 4, 6, 8, 10), # Number of neurons in the hidden layer
  decay = c(0.001, 0.01, 0.1, 0.5,1) )
control <- trainControl(method = "cv", # cross-validation
  number = 5, # number of folds in cross-validation
  classProbs = TRUE ,
  summaryFunction = twoClassSummary,
  verboseIter = TRUE,
  savePredictions = "final",
  returnResamp = "final" )
set.seed(1)
model <- train(Target ~ ., data = trainset,
  method = "nnet",
  trControl = control,
  tuneGrid = grid,
  maxit = 200, # how many times the network will iterate
  MaxNWts = 1000, #complexity of the model, size of the network
  metric = "ROC",
  trace = FALSE,
  linout = FALSE )

#5. GBM
grid = expand.grid(interaction.depth = seq(1,6,1),
  n.trees = c(50, 100, 150, 200, 250, 300),
  shrinkage = c(0.01, 0.1),
  n.minobsinnode =c(10, 20) )
control <- trainControl(
  method = "cv",
  number = 5,
  classProbs = TRUE,
  summaryFunction = twoClassSummary,
  verboseIter = TRUE ,
  savePredictions = "final",
  returnResamp = "final")
set.seed(1)
model <- train(Target~ ., data = trainset,
  method = "gbm",
  metric = "ROC" ,
  trControl = control,
  tuneGrid = grid ,
  verbose = FALSE )

#6. SVM
grid <- expand.grid(C = c(0.1, 1, 10))
control <- trainControl(
  method = "cv", # cross-validation
  number = 5, # number of folds in cross-validation
  classProbs = TRUE,
  summaryFunction = twoClassSummary,
  verboseIter = TRUE,
  savePredictions = "final",
  returnResamp = "final" )
set.seed(1)
model <- train(Target ~ ., data = trainset,
  method = "svmLinear",
  tuneGrid = grid,
  trControl = control,
  metric = "ROC",
  verbose = TRUE )
```