**Summary and results for 3 models tested**

Classification is a popular data mining task, where the value of a discrete (dependent) variable is predicted, based on the values of several independent variables. In this research, we investigate how predictive classification models can be inferred from the available data. The classification models are required to make good predictions, and be comprehensible and intuitive.

A total of 1000 observations from training data and 1111 observations from validation data were used. 126 variables were selected using near zero variance

**KNN Model results**

> modKnn

k-Nearest Neighbors

1000 samples

 119 predictor

   4 classes: '0-0', '2-1', '2-2', '3-2'

Pre-processing: centered (119), scaled (119)

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 900, 900, 900, 900, 900, 899, ...

Resampling results across tuning parameters:

  k   Accuracy   Kappa

   5  0.9924874  0.9847971

   7  0.9887374  0.9771811

   9  0.9862374  0.9721204

  11  0.9824747  0.9645812

  13  0.9787121  0.9570045

  15  0.9774621  0.9544988

  17  0.9762121  0.9520047

  19  0.9674495  0.9344403

  21  0.9624495  0.9244412

  23  0.9599495  0.9194416

Accuracy was used to select the optimal model using  the largest value.

The final value used for the model was k = 5.

> knnPredict <- predict(modKnn,data=testdata)

> table(knnPredict)

knnPredict

0-0 2-1 2-2 3-2

  0 446   0 554

>confusionMatrix(modKnn, testdata$Y\_FB)

Cross-Validated (10 fold) Confusion Matrix

(entries are un-normalized aggregated counts)

          Reference

Prediction 0-0 2-1 2-2 3-2

       0-0   0   0   0   0

       2-1   0 351   0   5

       2-2   0   0   0   0

       3-2   0   1   0 442

 Accuracy (average) : 0.9925

**SVM Model Results**

> modSvm

L2 Regularized Support Vector Machine (dual) with Linear Kernel

1000 samples

 119 predictor

   4 classes: '0-0', '2-1', '2-2', '3-2'

Pre-processing: centered (119), scaled (119)

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 899, 900, 899, 900, 900, 901, ...

Resampling results across tuning parameters:

  cost    Loss  Accuracy   Kappa

    0.25  L1    0.9899897  0.9798148

    0.25  L2    0.9849893  0.9697784

    0.50  L1    0.9899897  0.9798049

    0.50  L2    0.9869996  0.9737991

    1.00  L1    0.9899897  0.9798344

    1.00  L2    0.9909796  0.9818127

    2.00  L1    0.9929998  0.9858851

    2.00  L2    0.9919998  0.9838609

    4.00  L1    0.9929998  0.9858851

    4.00  L2    0.9920099  0.9838953

    8.00  L1    0.9919998  0.9838510

    8.00  L2    0.9859895  0.9718074

   16.00  L1    0.9899998  0.9798123

   16.00  L2    0.9839895  0.9677459

   32.00  L1    0.9909998  0.9818267

   32.00  L2    0.9869895  0.9737796

   64.00  L1    0.9909998  0.9818267

   64.00  L2    0.9859796  0.9717306

  128.00  L1    0.9899998  0.9798123

  128.00  L2    0.9839895  0.9677166

Accuracy was used to select the optimal model using  the largest value.

The final values used for the model were cost = 2 and Loss = L1.

> svmPredict <- predict(modSvm,data=testdata)

> table(svmPredict)

svmPredict

0-0 2-1 2-2 3-2

  0 444   0 556

> confusionMatrix(modSvm, testdata$Y\_FB)

Cross-Validated (10 fold) Confusion Matrix

(entries are un-normalized aggregated counts)

          Reference

Prediction 0-0 2-1 2-2 3-2

       0-0   0   0   0   0

       2-1   1 440   1   3

       2-2   0   0   0   0

       3-2   0   1   0 554

 Accuracy (average) : 0.994

**RF model results**

> modRF

Random Forest

1000 samples

 119 predictor

   4 classes: '0-0', '2-1', '2-2', '3-2'

Pre-processing: centered (119), scaled (119)

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 900, 900, 900, 900, 901, 900, ...

Resampling results across tuning parameters:

  mtry  Accuracy   Kappa

    2   1.0000000  1.0000000

   18   1.0000000  1.0000000

   35   1.0000000  1.0000000

   52   1.0000000  1.0000000

   68   1.0000000  1.0000000

   85   1.0000000  1.0000000

  102   1.0000000  1.0000000

  119   0.9987624  0.9975226

Accuracy was used to select the optimal model using  the largest value.

The final value used for the model was mtry = 2.

> plot(modRF, print.thres = 0.5, type="S")

> rfPredict <- predict(modRF, newdata = testdata)

> table(rfPredict)

rfPredict

0-0 2-1 2-2 3-2

  0 986   0 125

> confusionMatrix(modRF, testdata$Y\_FB)

Cross-Validated (10 fold) Confusion Matrix

(entries are un-normalized aggregated counts)

          Reference

Prediction 0-0 2-1 2-2 3-2

       0-0   0   0   0   0

       2-1   0 352   0   0

       2-2   0   0   0   0

       3-2   0   0   0 445

 Accuracy (average) : 1