IE 583 HW – 4

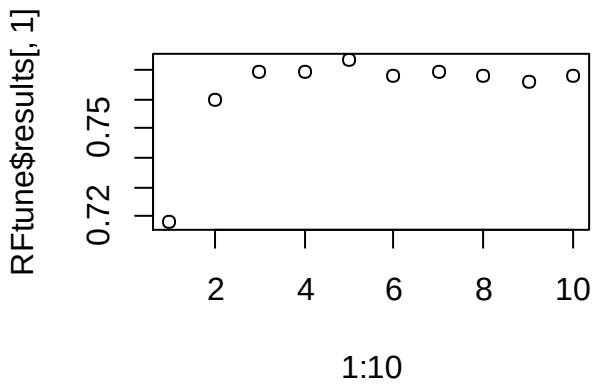
Deepak-George Thomas

1. Random Forest

1.1 Tuning Random Forest (using caret package) taking Out of Bag error values into account –

mtry values 1 to 10

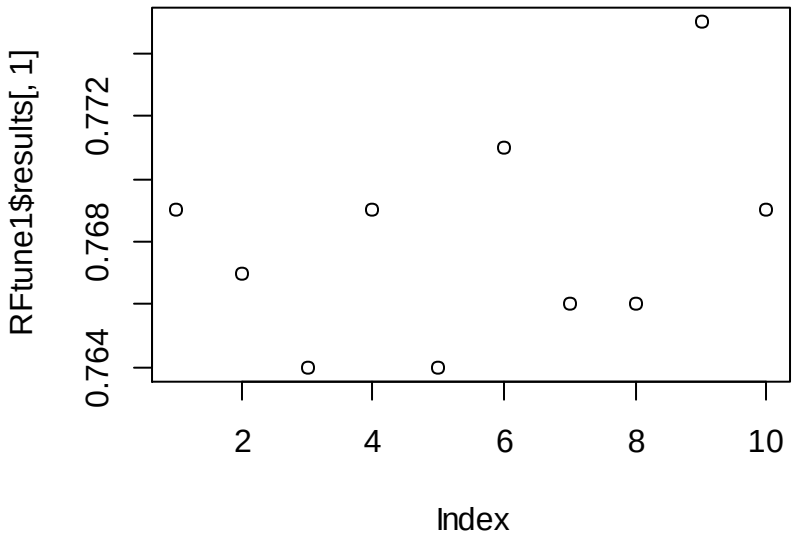
```
  Accuracy      Kappa mtry
1     0.718 0.08678756    1
2     0.760 0.32885906    2
3     0.769 0.37159956    3
4     0.769 0.38367129    4
5     0.773 0.39691817    5
6     0.768 0.38947368    6
7     0.769 0.39274448    7
8     0.768 0.38818565    8
9     0.766 0.38679245    9
10    0.768 0.39330544   10
```



mtry values 11 to 20

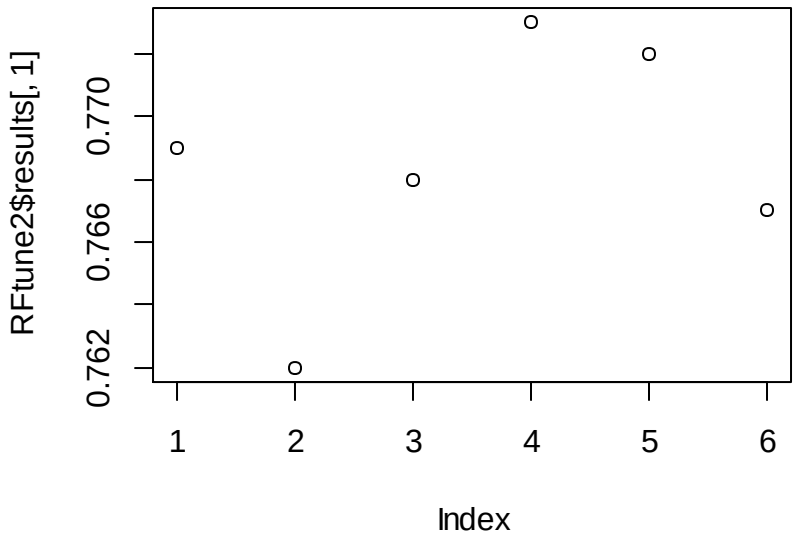
```
Accuracy     Kappa mtry
1     0.769 0.3978102   11
2     0.770 0.4010417   12
3     0.772 0.4111570   13
4     0.767 0.3976215   14
5     0.771 0.4104016   15
6     0.767 0.4013361   16
7     0.758 0.3762887   17
8     0.768 0.4057377   18
9     0.774 0.4175258   19
```

```
10     0.766 0.3969072   20
```



mtry values from 21 to 25

```
Accuracy     Kappa mtry
1    0.769 0.4076923   20
2    0.762 0.3865979   21
3    0.768 0.4032922   22
4    0.773 0.4191402   23
5    0.772 0.4123711   24
6    0.767 0.4001030   25
```



The best accuracy of 0.774 is obtained from a model with mtry = 19.

1.2 ***Evaluation of Random Forest Results –***

Once tuning is completed, the next step is to evaluate the test error estimate. This was done using cross validation as well as independent test data set.

```
mtry  Accuracy  Kappa
   2    0.716     0.08483305
  25    0.735     0.30102149
  48    0.731     0.30462854
```

The CV results show that a mtry value of 25 leads to highest accuracy of 0.735.

Evaluation of RF results using independent test data

```
Prediction bad good
      bad   46   53
      good  21  210

               Accuracy : 0.7758
                 95% CI : (0.7269, 0.8196)
```

1.3 ***Input Engineering for Random Forest***

Input Engineering was performed to account for data imbalance as the ratio of good to bad classes were 700 to 300.

1.3.1 ***Random Forest with Upsampled Data***

OOB estimate of error rate: 9.43% (Out of bag error rate)

Confusion matrix:

	bad	good	class.error
bad	*659*	*41*	*0.05857143*
good	*91*	*609*	*0.13000000*

The confusion matrix is based on entire data (therefore no test set), however the oob error rate gives an estimate of true error.
Accuracy obtained using CV = 0.736. While performing cross validatation for upsampled data, the folds were manually made using for loop. This was done to ensure the test folds were independent of the training folds.

1.3.2 ***Random Forest with Downsampled Data***

OOB estimate of error rate: 25.5% (Out of bag error rate)

Confusion matrix:

	bad	good	class.error
bad	*225*	*75*	*0.25*
good	*78*	*222*	*0.26*

The confusion matrix is based on entire data, however the oob error rate gives an estimate of true error.
Accuracy obtained using CV = 0.6625. While performing cross validatation for downsampled data, the folds were manually made using for loop. This was done to ensure the test folds were independent of the training folds.

2. Support Vector Machines

2.1 Fitting the model and predicting the training error

Confusion Matrix

```
Reference
Prediction bad good
      bad  116   30
      good 184  670
```

```
      Training  Accuracy : 0.786
```

```
  2.2    Estimation of test error
```

```
2.2.1  Independent Test Data Set
  Reference
Prediction bad good
      bad   20    6
      good  79  225
```

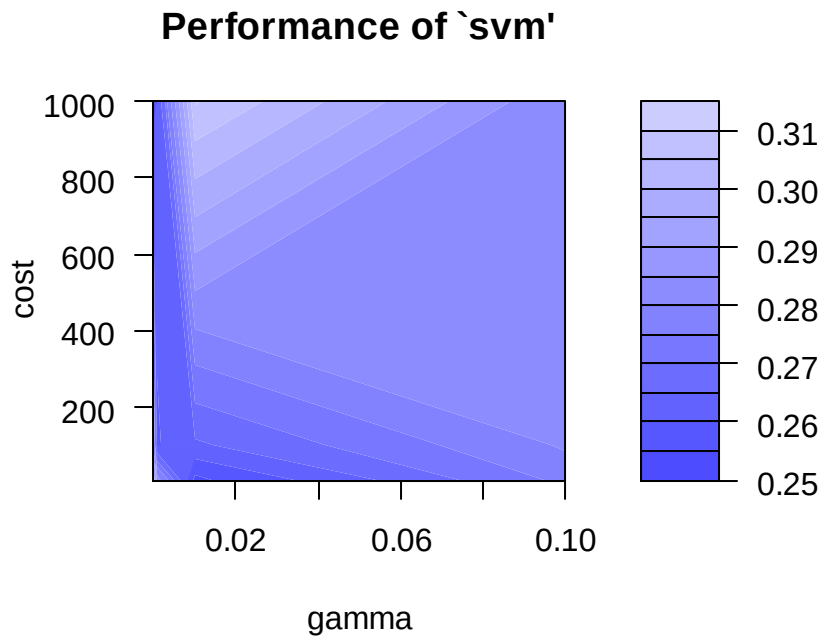
```
         Accuracy : 0.7424
           95% CI : (0.6917, 0.7888)
```

2.2.2 Tuning Parameters for Cross Validation using radial kernel

The parameters that were tuned were gamma and cost –

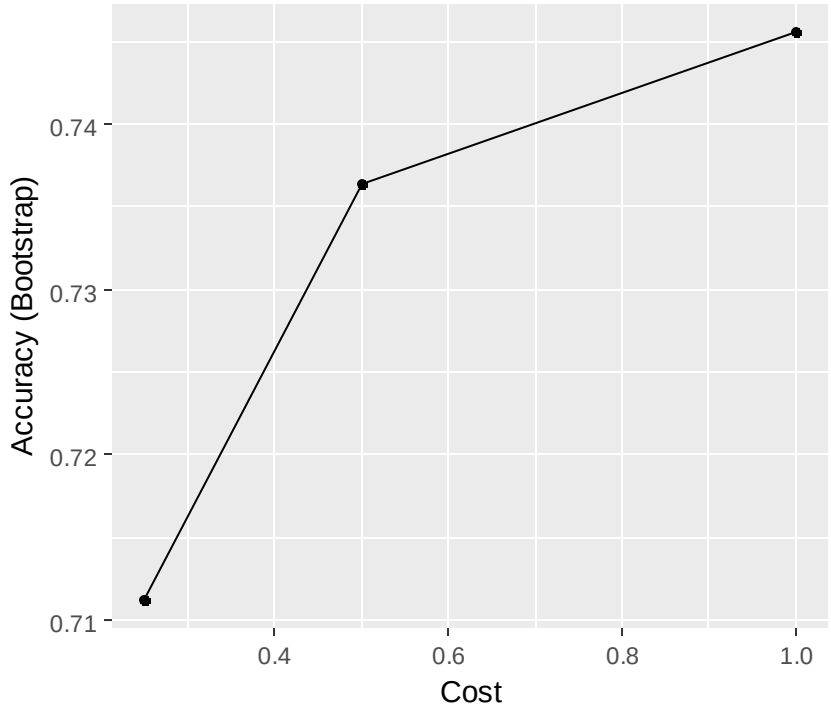
Tuning Parameters on a test data set

```
Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 gamma cost
  0.01   10

- best performance: 0.2537313

- Detailed performance results:
   gamma cost     error dispersion
1  1e-08   10 0.3000000 0.05286289
2  1e-07   10 0.3000000 0.05286289
3  1e-06   10 0.3000000 0.05286289
4  1e-05   10 0.3000000 0.05286289
5  1e-04   10 0.3000000 0.05286289
6  1e-03   10 0.2835821 0.05844448
7  1e-02   10 0.2537313 0.04505167
8  1e-01   10 0.2761194 0.03995617
9  1e-08  100 0.3000000 0.05286289
10 1e-07  100 0.3000000 0.05286289
11 1e-06  100 0.3000000 0.05286289
12 1e-05  100 0.3000000 0.05286289
13 1e-04  100 0.2850746 0.05821108
14 1e-03  100 0.2597015 0.04624458
15 1e-02  100 0.2641791 0.04452667
16 1e-01  100 0.2805970 0.04382630
17 1e-08 1000 0.3000000 0.05286289
18 1e-07 1000 0.3000000 0.05286289
19 1e-06 1000 0.3000000 0.05286289
20 1e-05 1000 0.2865672 0.06085128
21 1e-04 1000 0.2582090 0.03726365
22 1e-03 1000 0.2611940 0.04294198
23 1e-02 1000 0.3104478 0.04438748
24 1e-01 1000 0.2805970 0.04382630
```



On the basis of the optimum parameters obtained above, the search grid was chosen to be - gamma=10^(-8:-1),cost=10^(-1:3))

```
  C     Accuracy   Kappa
  0.25  0.7111930  0.08809652
  0.50  0.7364011  0.25054373
  1.00  0.7455922  0.33121592
The final values used for the model were sigma = 0.01248137 and C = 1.
The cost function giving highest accuracy is one order of magnitude lower tha
n what was expected.
```



2.3 Input Engineering for Support Vector Machines

2.3.1 SVM using Upsampled data

SVM results after Upsampling

```
Reference
Prediction bad good
      bad  579  175
      good 121  525
Training Accuracy : 0.7886
                 95% CI : (0.7662, 0.8097)
```

Estimate of True Error –
Independent Test Data Set

Independent Test Data Set

```
Prediction bad good
      bad  179   70
      good  52  161

               Accuracy : 0.7359
                 95% CI : (0.6932, 0.7756)
```

Accuracy obtained using CV – 0.7198214

2.3.2 SVM using Downsampled data

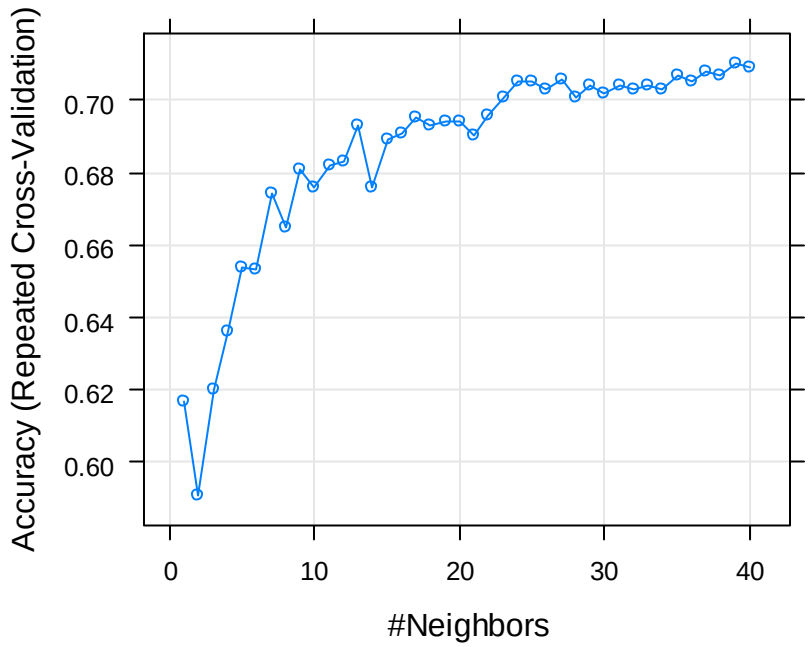
```
   Reference
Prediction bad good
      bad  243   69
      good  57  231

               Training Accuracy : 0.79
                 95% CI : (0.7552, 0.8219)

Estimate of True Error –
Independent Test Data Set
Reference
Prediction bad good
      bad   80   29
      good  19   70

               Accuracy : 0.7576
                 95% CI : (0.6917, 0.8155)
```

Accuracy obtained using CV – 0.5754167

Both input engineering methods, produce (test) accuracy lower than than obtained without any feature engineering.

3. Comparision of SVM, Random Forest with KNN, Naïve Bayes & Decision Trees

3.1 KNN Results –

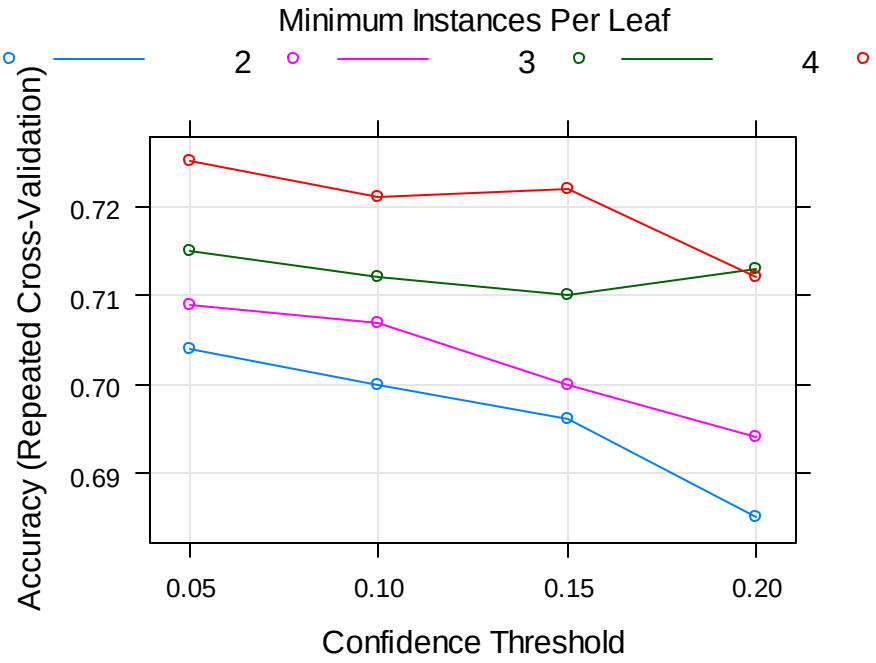


The highest accuracy of 71% was found to be at k=39. The number of neighbours was limited to k = 40 due to computational constraints. However, from class notes we know that all k values above 60 overfit to the data and the best model accuracy values are between 20 and 60.

3.2 Naïve Bayes

usekernel	Accuracy	Kappa
FALSE	0.6944444	0.344231192
TRUE	0.7010000	0.004605263

3.3 Decision Trees



The best accuracy of 0.725 was found with C = 0.05 and M = 4.

Random Forest and SVM without the assistance of any input engineering produced estimates of true error of 0.774 and 0.745 respectively, higher than that of decision trees, naïve bayes and knn.