Assignment 5

Arvind Pawar

June 28, 2019

ALY6020-Predictive Analytics

Instructor: Dr. Marco Montes de Oca

K-nearest neighbor

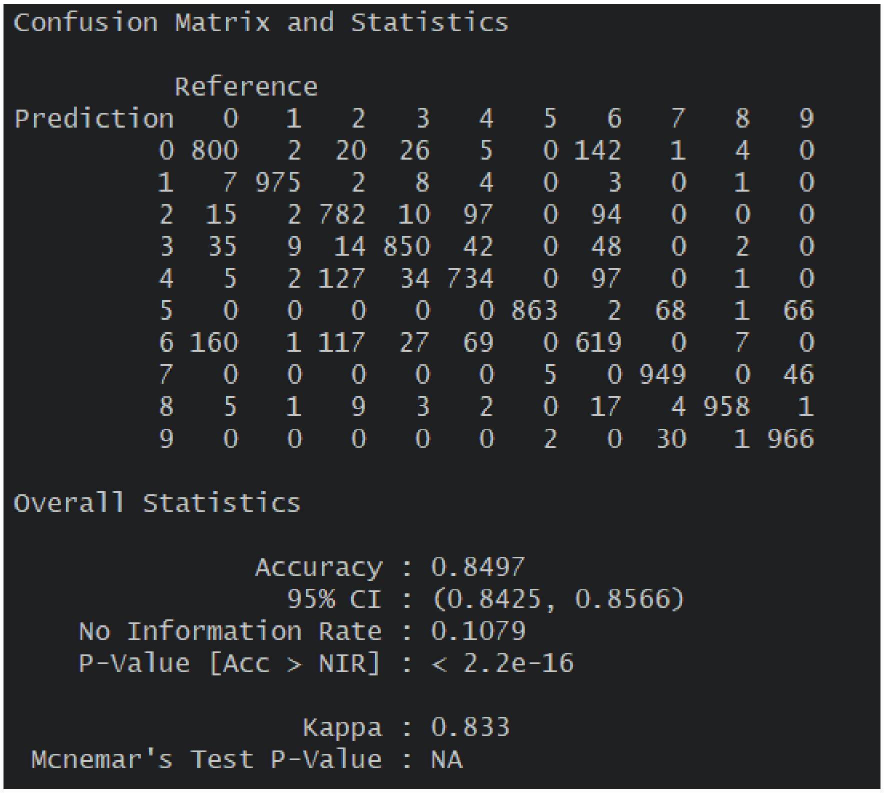
Introduction

- KNN is a widely used classification technique but can also be used for predictive regression problems.
- It is easy to interpret the output
- It has more predictive power as compared to other models.
- It is non-parametric, that means it does not make an underlying assumption about the distribution of data.
- When we train the data, it classifies coordinates into groups which are identified by an attribute.
- The new test instances coordinates are identified on the same plane, and a circular plane is drawn around these coordinates such that the number of elements lying within the circular plane is equal to the given k value.
- Then the Euclidean distance between the test value and all the neighbors is calculated.
- The train instance is predicted to belong in the class of majority nearest neighbors.
- K value is user defined quantity, which significantly affects the accuracy of the model. In this assignment, our goal is to use 'K-nearest neighbors' classifier to the clothing item dataset.
- We have already applied logistic regression on the same dataset. Let us apply the KNN algorithm using different K's. K= 1, 11, and 21. We used the class library in R.

Analysis

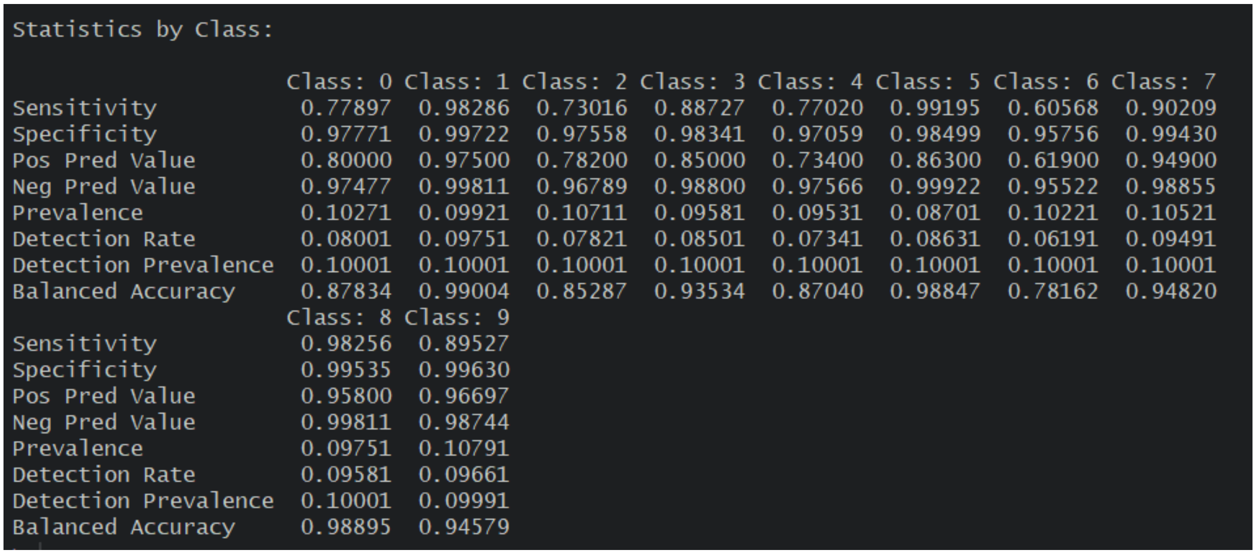
KNN with K=1

- KNN with k=1 refers that only one nearest neighbor is considered to decide the new data point.
- In this model, when the observed label is 0, there are 800 observations predicted correctly by this model.
- 2, 20, 26, 5, 0, 142, 1, 4, 0 are incorrectly predicted as clothing item 1-9 respectively.
- Similarly, the confusion matrix clearly shows the number of observations predicted from 0-9 and their corresponding observed values.
- For this model, the accuracy is 84.97% with a 95% confidence interval.
- The kappa value measures the performance of the classifier. If the value of kappa is less, then there is a small difference between accuracy and null error rate.



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1   2   3   4   5   6   7   8   9
         0 800   2  20  26   5   0 142   1   4   0
         1   7 975   2   8   4   0   3   0   1   0
         2  15   2 782  10  97   0  94   0   0   0
         3  35   9  14 850  42   0  48   0   2   0
         4   5   2 127  34 734   0  97   0   1   0
         5   0   0   0   0   0 863   2  68   1  66
         6 160   1 117  27  69   0 619   0   7   0
         7   0   0   0   0   0   5   0 949   0  46
         8   5   1   9   3   2   0  17   4 958   1
         9   0   0   0   0   0   2   0  30   1 966

Overall Statistics

               Accuracy : 0.8497
                 95% CI : (0.8425, 0.8566)
    No Information Rate : 0.1079
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.833
 Mcnemar's Test P-Value : NA
```

- Sensitivity is a measurement that determines the probability of actual positives; that is, the proportion of test observation predicted correctly.
- Specificity is the measurement of that determines the probabilities of actual negatives, that is the proportion of test observation not predicted as (say 0) when there were not labelled as 0.
- Class 5 has a maximum sensitivity of 99.19%, which shows that most of the test instances correctly predicted as 5. Class 6 has a minimum sensitivity of 60.56% as compared to other classes.
- Class 9 has maximum specificity 99.63%, and class 6 has less specificity as compared to other classes.



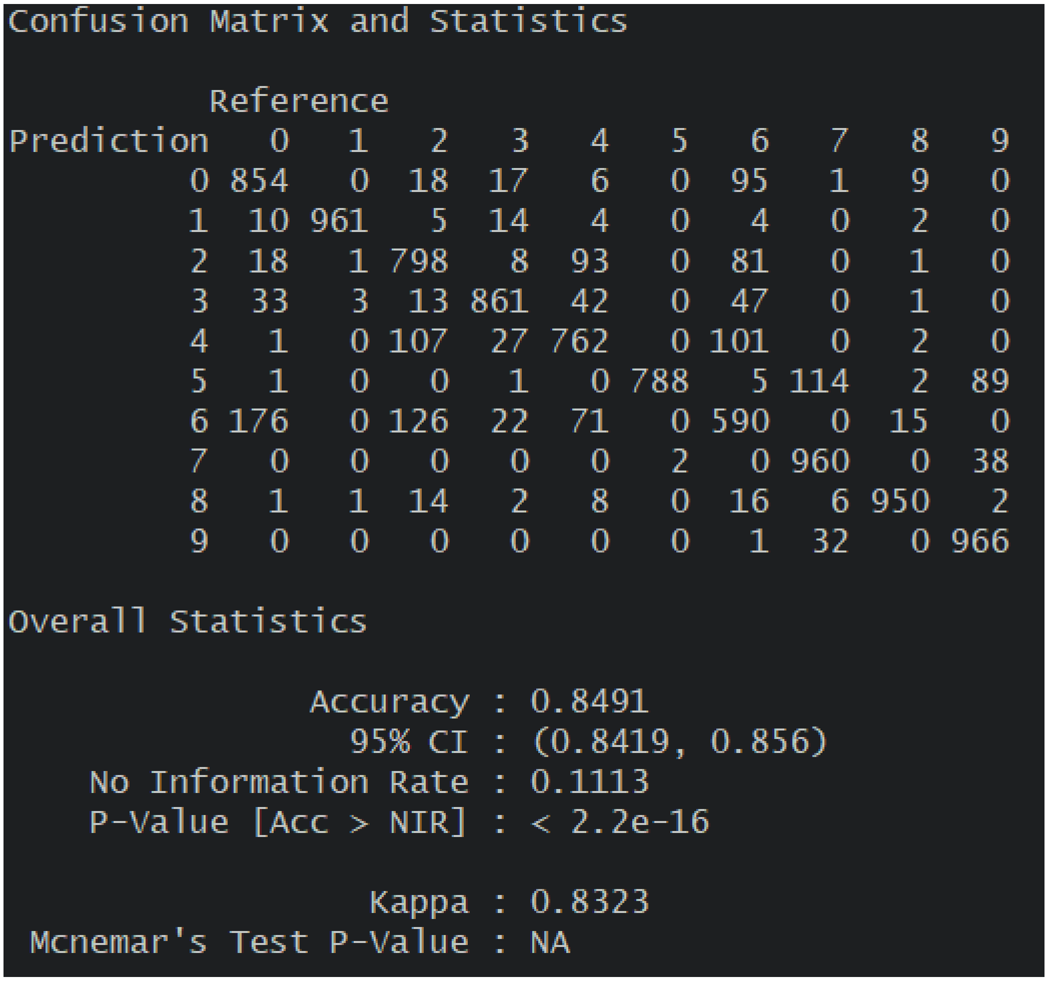
```
Statistics by Class:

                     Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6 Class: 7
Sensitivity           0.77897  0.98286  0.73016  0.88727  0.77020  0.99195  0.60568  0.90209
Specificity           0.97771  0.99722  0.97558  0.98341  0.97059  0.98499  0.95756  0.99430
Pos Pred Value        0.80000  0.97500  0.78200  0.85000  0.73400  0.86300  0.61900  0.94900
Neg Pred Value        0.97477  0.99811  0.96789  0.98800  0.97566  0.99922  0.95522  0.98855
Prevalence            0.10271  0.09921  0.10711  0.09581  0.09531  0.08701  0.10221  0.10521
Detection Rate        0.08001  0.09751  0.07821  0.08501  0.07341  0.08631  0.06191  0.09491
Detection Prevalence  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001
Balanced Accuracy     0.87834  0.99004  0.85287  0.93534  0.87040  0.98847  0.78162  0.94820
                     Class: 8 Class: 9
Sensitivity           0.98256  0.89527
Specificity           0.99535  0.99630
Pos Pred Value        0.95800  0.96697
Neg Pred Value        0.99811  0.98744
Prevalence            0.09751  0.10791
Detection Rate        0.09581  0.09661
Detection Prevalence  0.10001  0.09991
Balanced Accuracy     0.98895  0.94579
```

-

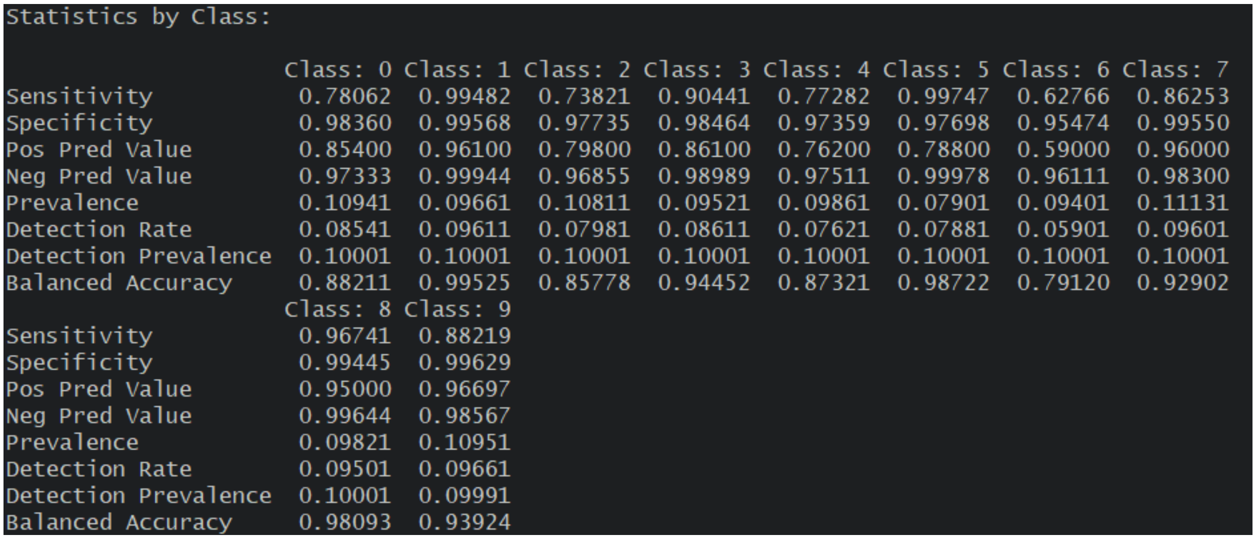
KNN with k=11

- Now we have taken k=11, that means could consider 11 nearest neighbors to decide the new data points.

- In this, When the observed label is 0, there are 854 observations predicted correctly by this model.
- Which were 800 when K value was 1.
- It has been observed that the number of incorrectly predicted values have increased for some classes.
- For this model, we have got 84.91% accuracy with a 95% confidence interval.



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1   2   3   4   5   6   7   8   9
         0 854   0  18  17   6   0  95   1   9   0
         1  10 961   5  14   4   0   4   0   2   0
         2  18   1 798   8  93   0  81   0   1   0
         3  33   3  13 861  42   0  47   0   1   0
         4   1   0 107  27 762   0 101   0   2   0
         5   1   0   0   1   0 788   5 114   2  89
         6 176   0 126  22  71   0 590   0  15   0
         7   0   0   0   0   0   2   0 960   0  38
         8   1   1  14   2   8   0  16   6 950   2
         9   0   0   0   0   0   0   1  32   0 966

Overall Statistics

               Accuracy : 0.8491
                 95% CI : (0.8419, 0.856)
    No Information Rate : 0.1113
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8323
 Mcnemar's Test P-Value : NA
```

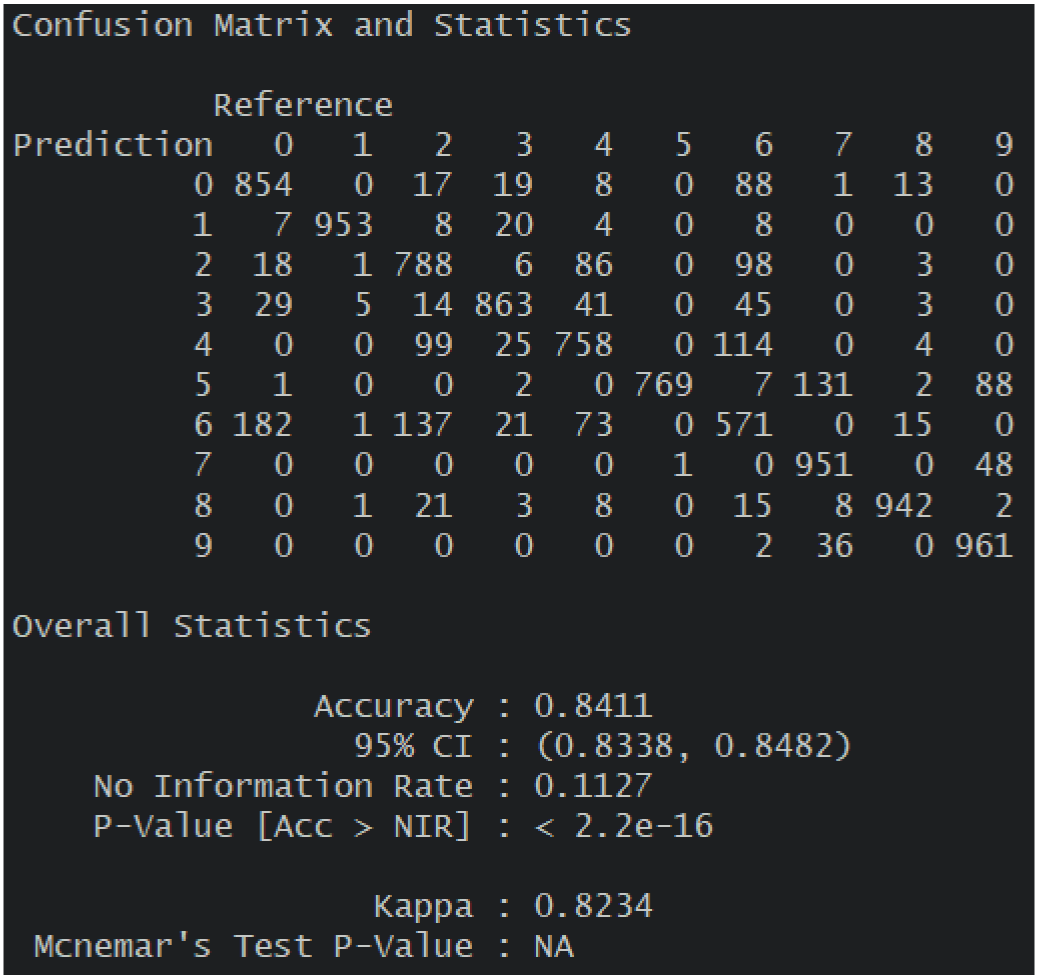


```
Statistics by Class:

                     Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6 Class: 7
Sensitivity           0.78062  0.99482  0.73821  0.90441  0.77282  0.99747  0.62766  0.86253
Specificity           0.98360  0.99568  0.97735  0.98464  0.97359  0.97698  0.95474  0.99550
Pos Pred Value        0.85400  0.96100  0.79800  0.86100  0.76200  0.78800  0.59000  0.96000
Neg Pred Value        0.97333  0.99944  0.96855  0.98989  0.97511  0.99978  0.96111  0.98300
Prevalence            0.10941  0.09661  0.10811  0.09521  0.09861  0.07901  0.09401  0.11131
Detection Rate        0.08541  0.09611  0.07981  0.08611  0.07621  0.07881  0.05901  0.09601
Detection Prevalence  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001
Balanced Accuracy     0.88211  0.99525  0.85778  0.94452  0.87321  0.98722  0.79120  0.92902
                     Class: 8 Class: 9
Sensitivity           0.96741  0.88219
Specificity           0.99445  0.99629
Pos Pred Value        0.95000  0.96697
Neg Pred Value        0.99644  0.98567
Prevalence            0.09821  0.10951
Detection Rate        0.09501  0.09661
Detection Prevalence  0.10001  0.09991
Balanced Accuracy     0.98093  0.93924
```

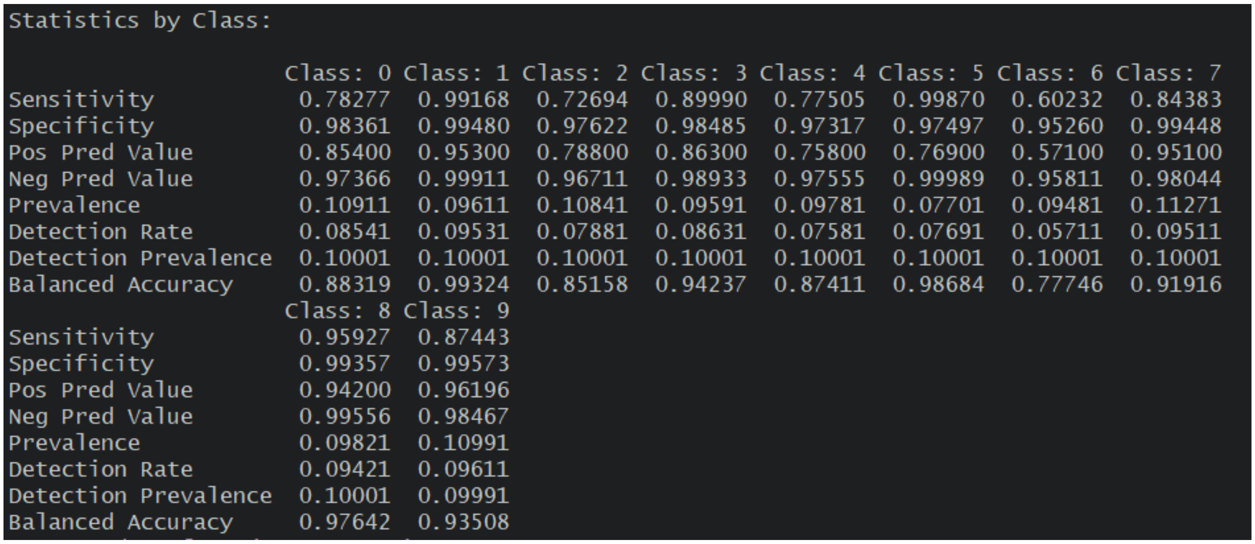
-

KNN with k=21

- Now we have taken k=21, that means could consider 21 nearest neighbors to decide the new data points.
- In this, When the observed label is 0, there are 854 observations predicted correctly by this model.
- Which are similar to that of the previous model with k=11.
- Here, it has been observed that the number of incorrectly predicted values have increased for some classes than the previous model.
- This model gave us 84.91% accuracy with a 95% confidence interval.



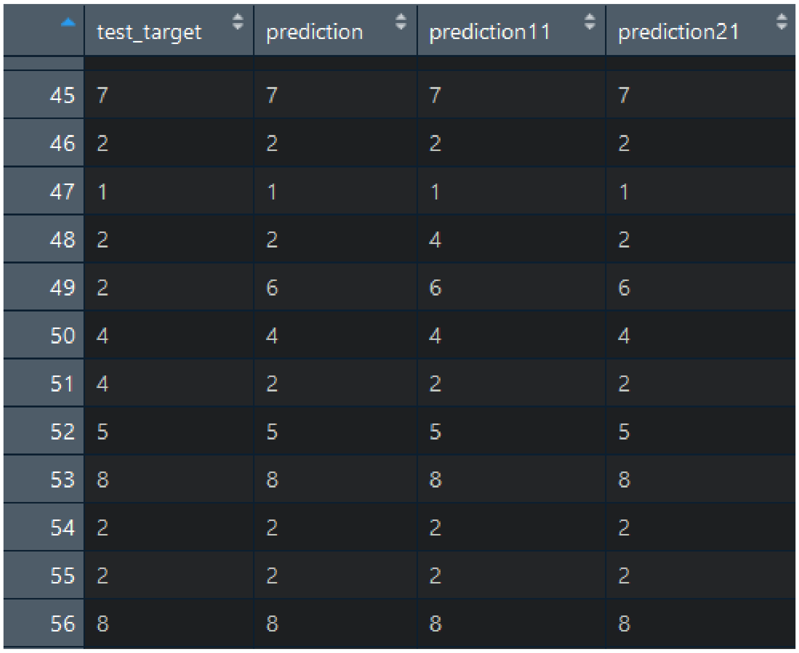
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1   2   3   4   5   6   7   8   9
         0 854   0  17  19   8   0  88   1  13   0
         1   7 953   8  20   4   0   8   0   0   0
         2  18   1 788   6  86   0  98   0   3   0
         3  29   5  14 863  41   0  45   0   3   0
         4   0   0  99  25 758   0 114   0   4   0
         5   1   0   0   2   0 769   7 131   2  88
         6 182   1 137  21  73   0 571   0  15   0
         7   0   0   0   0   0   1   0 951   0  48
         8   0   1  21   3   8   0  15   8 942   2
         9   0   0   0   0   0   0   2  36   0 961

Overall Statistics

               Accuracy : 0.8411
                 95% CI : (0.8338, 0.8482)
    No Information Rate : 0.1127
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8234
 Mcnemar's Test P-Value : NA
```



```
Statistics by Class:

                     Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6 Class: 7
Sensitivity           0.78277  0.99168  0.72694  0.89990  0.77505  0.99870  0.60232  0.84383
Specificity           0.98361  0.99480  0.97622  0.98485  0.97317  0.97497  0.95260  0.99448
Pos Pred Value        0.85400  0.95300  0.78800  0.86300  0.75800  0.76900  0.57100  0.95100
Neg Pred Value        0.97366  0.99911  0.96711  0.98933  0.97555  0.99989  0.95811  0.98044
Prevalence            0.10911  0.09611  0.10841  0.09591  0.09781  0.07701  0.09481  0.11271
Detection Rate        0.08541  0.09531  0.07881  0.08631  0.07581  0.07691  0.05711  0.09511
Detection Prevalence  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001  0.10001
Balanced Accuracy     0.88319  0.99324  0.85158  0.94237  0.87411  0.98684  0.77746  0.91916
                     Class: 8 Class: 9
Sensitivity           0.95927  0.87443
Specificity           0.99357  0.99573
Pos Pred Value        0.94200  0.96196
Neg Pred Value        0.99556  0.98467
Prevalence            0.09821  0.10991
Detection Rate        0.09421  0.09611
Detection Prevalence  0.10001  0.09991
Balanced Accuracy     0.97642  0.93508
```

Predictions:

- In predictions table, the test target of row numbers other than 48, 51 were predicted correctly in all the models with k =1, 11 and 21.
- In row number 48, test target is 2, but the model with k value 11 classified label 2 incorrectly as label 4.
- Moreover, in row 51, the test target label is 4, and all the models have incorrectly classified it as label 4.



	test_target	prediction	prediction11	prediction21
45	7	7	7	7
46	2	2	2	2
47	1	1	1	1
48	2	2	4	2
49	2	6	6	6
50	4	4	4	4
51	4	2	2	2
52	5	5	5	5
53	8	8	8	8
54	2	2	2	2
55	2	2	2	2
56	8	8	8	8

- Using logistic regression, we obtained 73.09% accuracy.
- However, using KNN with different values of k (1, 11, and 21) we got accuracy of around 84%.
- This could be because of non-parametric model training.
- KNN is considered to perform much better than logistic when all the variables in the train dataset are used to generate a model.
- Therefore, the accuracy of KNN is higher than logistic.
- For digit recognition I would recommend a KNN algorithm to be used to train the model.

-

Conclusion

- The accuracies for K=1, 11, and 21 are 84.97%, 84.91%, and 84.11% respectively.
- We can observe a decreasing accuracy trend from K=11 to K=21.
- This suggest that the optimum K value would lie in the range of K=1 to K=11.
- Using KNN algorithm for this type of dataset the accuracy would never increase more than 84.97% because when K=1, the errors are least because only 1 nearest neighbor is used to classify any new instance of test set which makes its implementation fast.