Melissa LaHoud

Grand Canyon University

DSC-540: Machine Learning

Dr. Aiman Darwiche

3/31/2021

k-NN Algorthim Using MNIST Dataset

I used the MNIST dataset which includes a training set of 60,000 images and a test set of 10,000 Images. “The proposed method uses k-nearest neighbor (knn) classification algorithm for classifying the MNIST digit images in test database using the feature vector of training database. The k-nearest neighbor algorithm (k-NN) is classification technique to classify the objects base on training features space. The functionality of k-NN algorithm is to define the computations until classification is done irrespective of the learning techniques.” (Babu, U, etc., 2014) I used Euclidean distance measures to compute the distance between the values of the test sample and the training image. The majority among the k-nearest training samples was also based on Euclidean distance. I calculated the k by seeing which K had the highest accuracy. Below is a graph representation of are my results for the k values:

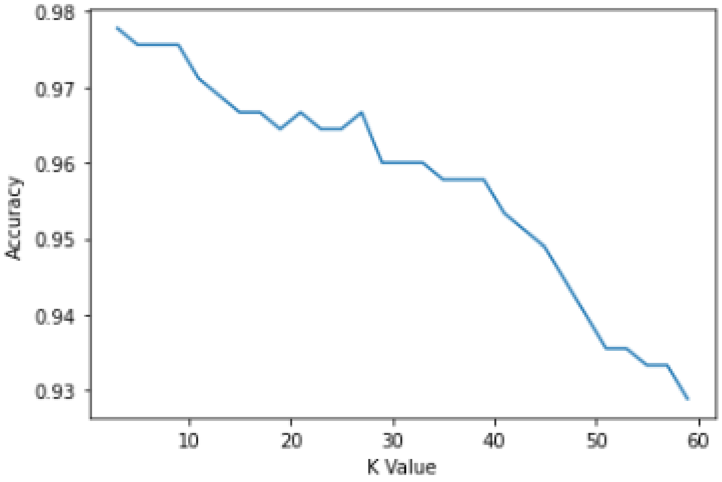
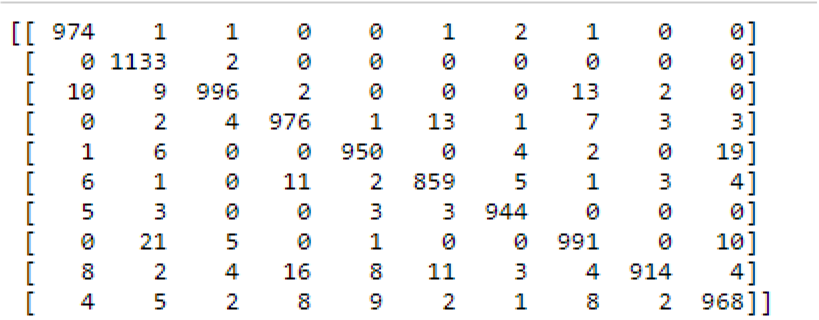


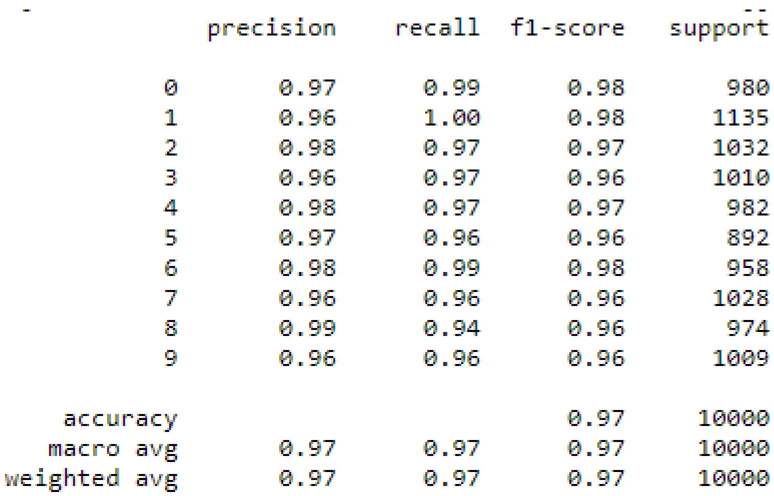
Figure 1. Effect of different testing samples on accuracy by taking different values of k

I executed the algorithm with the value of k is 3. This algorithm produced a high percentage of recognition for the algorithm at 97% as we can see in the table 2. We can also see in table 2 that the precision and recall values are also very large meaning the k-NN algorithm did a good job predicting the handwritten images.



```
[[ 974    1    1    0    0    1    2    1    0    0]
 [   0 1133    2    0    0    0    0    0    0    0]
 [  10    9  996    2    0    0    0   13    2    0]
 [   0    2    4  976    1   13    1    7    3    3]
 [   1    6    0    0  950    0    4    2    0   19]
 [   6    1    0   11    2  859    5    1    3    4]
 [   5    3    0    0    3    3  944    0    0    0]
 [   0   21    5    0    1    0    0  991    0   10]
 [   8    2    4   16    8   11    3    4  914    4]
 [   4    5    2    8    9    2    1    8    2  968]]
```

Table 1. Values of correct recognition for 10,000 test set



```
  -                                             --
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       980
           1       0.96      1.00      0.98      1135
           2       0.98      0.97      0.97      1032
           3       0.96      0.97      0.96      1010
           4       0.98      0.97      0.97       982
           5       0.97      0.96      0.96       892
           6       0.98      0.99      0.98       958
           7       0.96      0.96      0.96      1028
           8       0.99      0.94      0.96       974
           9       0.96      0.96      0.96      1009

    accuracy                           0.97     10000
   macro avg       0.97      0.97      0.97     10000
weighted avg       0.97      0.97      0.97     10000
```

Table 2. Conclusion of algorithm executed with k = 3

ROC curves are a nice way to see how any predictive model can distinguish between the true positives and negatives. The ROC curve does this by plotting sensitivity, the probability of predicting a real positive will be a positive, against 1-specificity, the probability of predicting a real negative will be a positive. The further the curve is from the diagonal line, the better the model is at discriminating between positives and negatives in general. When we look at Figure 2, we can see the model is almost always going to be good at discriminating between positives and negative.

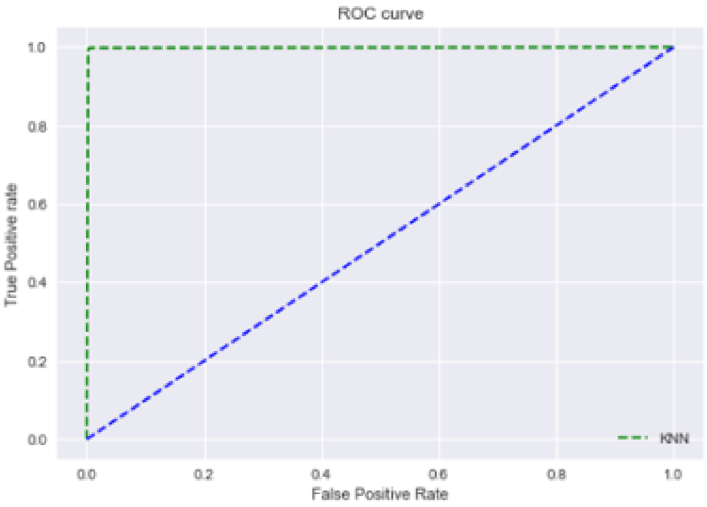


Figure 2. ROC curve

References

Babu, U., Venkateswarlu, Y, & Chintha, A. (2014) Handwritten Digit Recognition Using K-Nearest Neighbour Classifier. Retrieved from https://ieeexplore-ieee-org.lopes.idm.oclc.org/stamp/stamp.jsp?tp=&arnumber=6755106