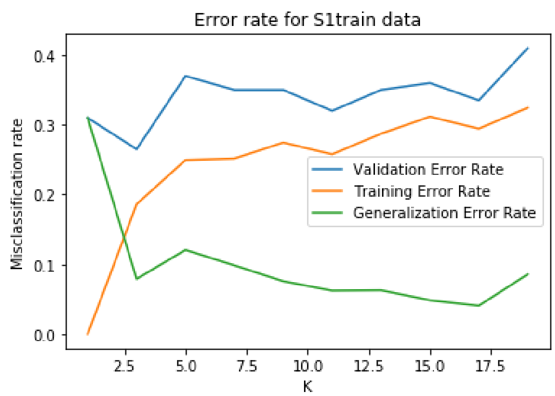
STAT 339 Spring 2020
Homework 1 Solutions
Garrett Robins

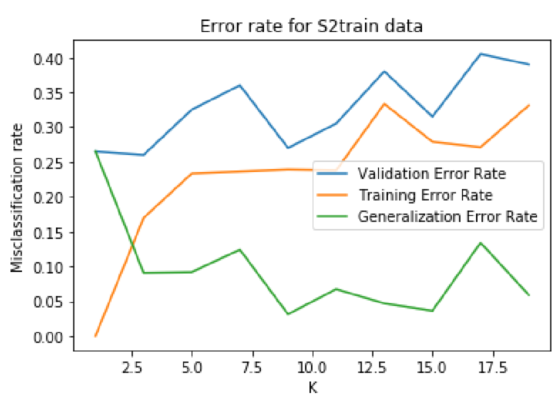
Honor Code: I affirm I have adhered to the Honor Code in this assignment - Garrett Robins

Part 1: KNN Implementation

Below is the graph I made for the KNN neighbors on the S1Train dataset. As you can see, the misclassification rate of the classifier was minimized on the validation set when K = 3. The misclassification rate of the training set was trivially minimized at K =1 and non-trivially minimized at K = 3. Thus, as the graph has a general upward trend for larger values of K, I believe K = 3 to be the value that minimizes misclassification rate.



Below is the graph I made for the KNN neighbors on the S2Train dataset. As you can see, the misclassification rate of the classifier was minimized on the validation set when K = 3. The misclassification rate of the training set was trivially minimized at K =1 and non-trivially minimized at K = 3. Thus, as the graph has a general upward trend for larger values of K, I believe K = 3 to be the value that minimizes misclassification rate.



From the sealed test sets, we should be able to find generalized performance for our K = 3 classifiers. In fact, when we train a KNN algorithm, with K = 3, with the training datasets, we see that the test datasets have these errors:

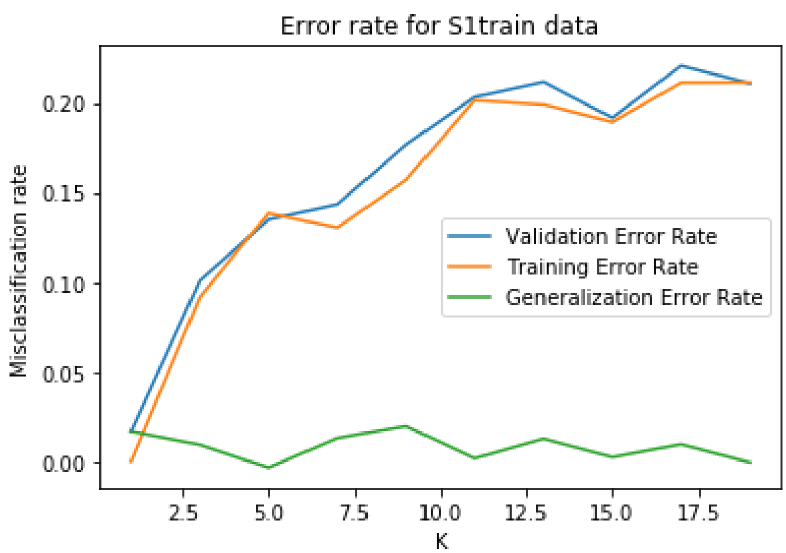
a. When the model is trained with S1train, the S1test dataset has a generalization error of **.267**

b. When the model is trained with S2train, the S2test dataset has a generaization error of **.295**

Although both of these data distributions had minimized classification error at K = 3, I don't know if I believe this to be true in general. It would seem that a more complex distribution would require more neighbors as reference points to minimize misclassification rate. To that end, it remains to be seen whether a more complex or simpler distribution leads to a higher K.

Part 2: KNN on Images

Find below the graph of KNN neighbors on the image data given. Notice that this graph has minimized error for both the training and validation sets when K = 1. Although I am hesitant to use K = 1, as it has trivially 0 training error, the benefits seem to outweigh the costs and thus K = 1 is the optimal choice for K. (Note: the graph's title is wrong)



When running a KNN algorithm trained on the training data on the test set, if we set K = 1 we get a **.0137** misclassification rate. This is quite small and makes sense. In 28 dimensional data, you don't need many neighbors to confirm the type of image you're seeing.