CS412 HW1

Wangfei Wang

1 Getting started

Only 1's and 5's were selected and separated randomly into training data (20%) and testing data (80%). R is used.

2 Choose Two Features

I chose to use mean intensity and intensity standard deviation as two features. **Figure 1.1** plots the two features with x axis mean intensity and y axis the standard deviation of intensity (red dots: digit 1 and blue dots: digit 5). These two features mean intensity and standard deviation of intensity separate the two groups of digits pretty well. Both features were normalized to [-1, 1] using a linear transformation.

3 1-Nearest Neighbor

The regions of the graph by which is the closest in the two-dimensional space were colored with the groups' respective colors (**Figure 1.2**).

1a) Do you believe that this model suffers from underfitting or overfitting? Why or why not?
The 1-nearest neighbor for separating digit "1" and digit "5" in our training data in the two-dimensional space generated by two features seems to be performing well. It shows a little overfitting when standard deviation of intensity is in the range of [-1, -0.5]. But overall, this model is pretty good.

1b) Is the error for this model equivalent to or different from the 1-Nearest Neighbor model in the 256-dimensioned space? Explain your answer.
The cross-validation errors for this two-dimensional model is different from the 1-Nearest Neighbor model in the 256-dimensioned space. This can be explained by the fact that when we build features, we are using the feature to summarize the information obtained by the whole 256 dimensions. Therefore, the Ecv for 256-dimension is smaller than that for 2-dimension model.

1c) Comment on any differences you see in the results and what may have resulted in them.
Euclidean distance - 2 dimensions vs. Euclidean distance - 256 dimensions has been explained

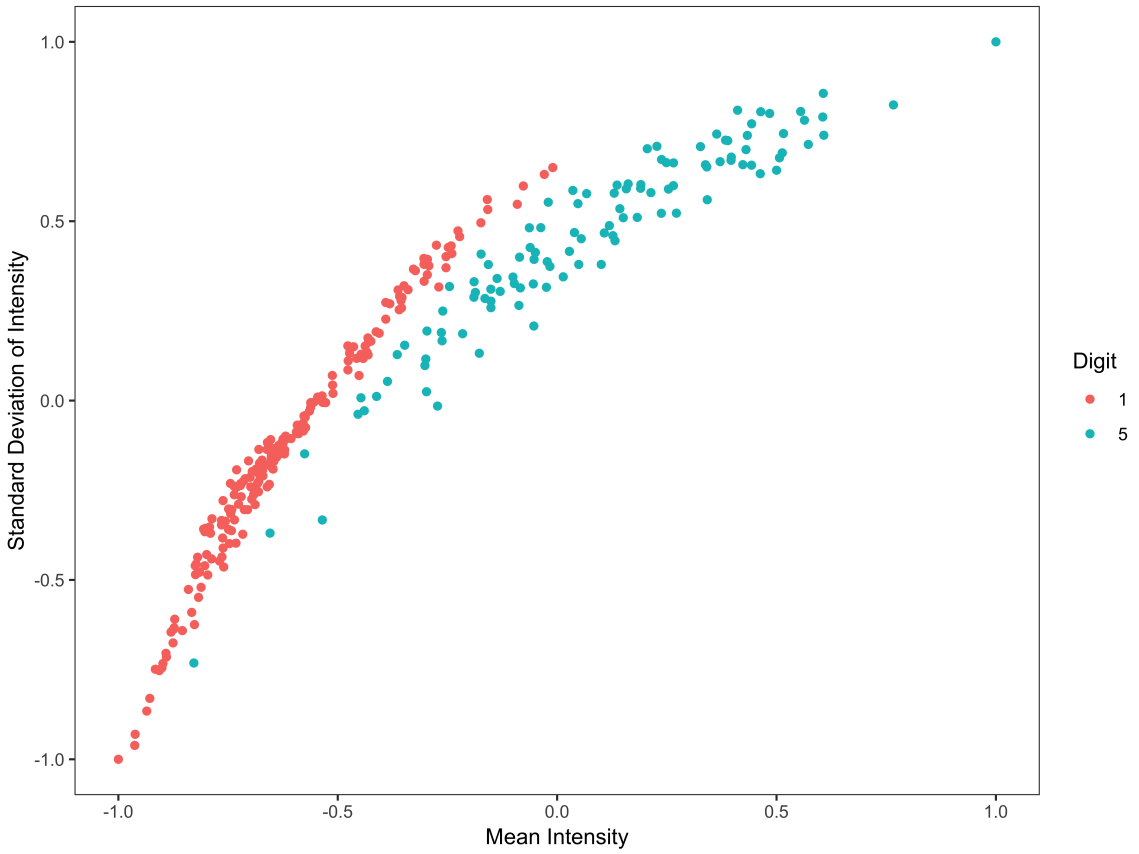

Figure 1.1. Two features (mean intensity and standard deviation of intensity) separating two groups of digits. Features normalized to [-1, 1].

above in 1b). The Ecv is ≈ 0 for 256-dimensions, while Ecv is 0.01270161 for 2 dimensions. The use of features for pooling the information from the original data set inevitably loses some information. For the all 2 dimensional cases, the Ecv are not varied too much because there aren't a lot of features. The Ecv is more different in 256 dimensions.

4 k-Nearest Neighbor

The odd k-Neighbor (k between 1 and 49) models were considered.

2a) **Figure 1.3** shows the influence of the choice of k on the cross validation errors for the 256-dimensional space. The Ecv's are increasing as k increases. Ecv reaches its minimum value when $k = 1$. Ecv is ≈ 0 when $k = 1$. Therefore, $k = 1$ yields the best result.

2b) Since $k = 1$ minimizes Ecv, I chose $k = 1$ to plot the graph of the 2-dimensional region for the nearest neighbor model (**Figure 1.4**). This figure is the same as Figure 1.2. This model shows a little bit overfitting in a small portion of the data when standard deviation is between -1 and -0.5. It classifies two digit groups pretty well.

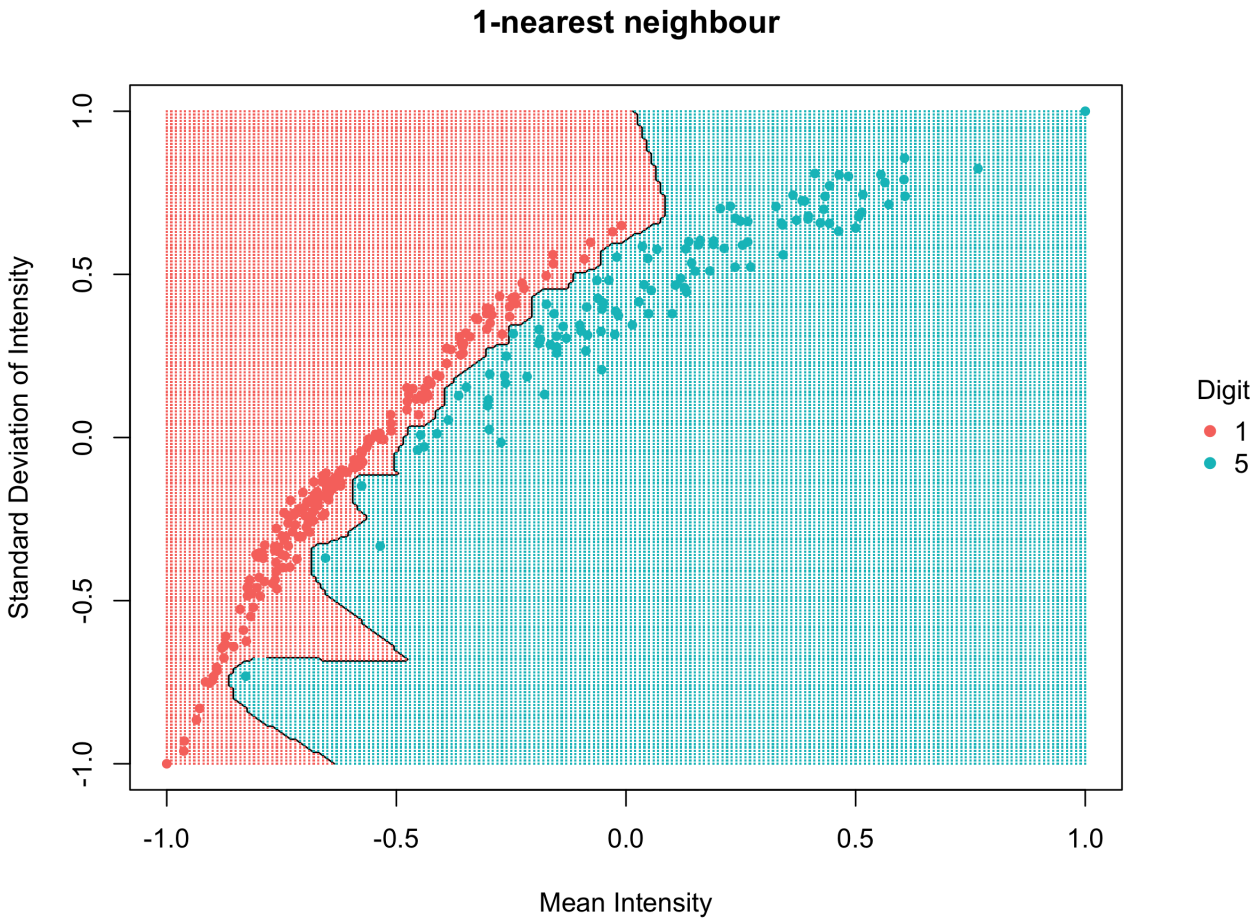

Figure 1.2. 1-Nearest neighbor with two features selected above.

2c) For the 25 models, 95% confidence level were generated by mean $\pm$ 1.96 $\times$ sd (**Table 1**, rounded to 3 digits). The lowest 95% CI upper bound is reached at the first model when k = 1. Again, this is the 1-nearest neighbor model, which shows a little overfitting in a small portion of the data when the standard deviation is between -1 and -0.5.

5 Extra credit

5.1 Smartphone-Based Recognition of Human Activities and Postural Transitions Data Set

Data set was downloaded from UCI Machine Learning Repository. In this data set, accelerometers and gyroscopes embedded in smartphones measured 3-axial linear acceleration and 3-axial angular velocity. The 561 features provided in the dataset were calculated from raw time series data for 30 subjects performing 12 different activities (standing, sitting, lying, walking, walking downstairs, walking upstairs, stand-to-sit, sit-to-stand, sit-to-lie, lie-

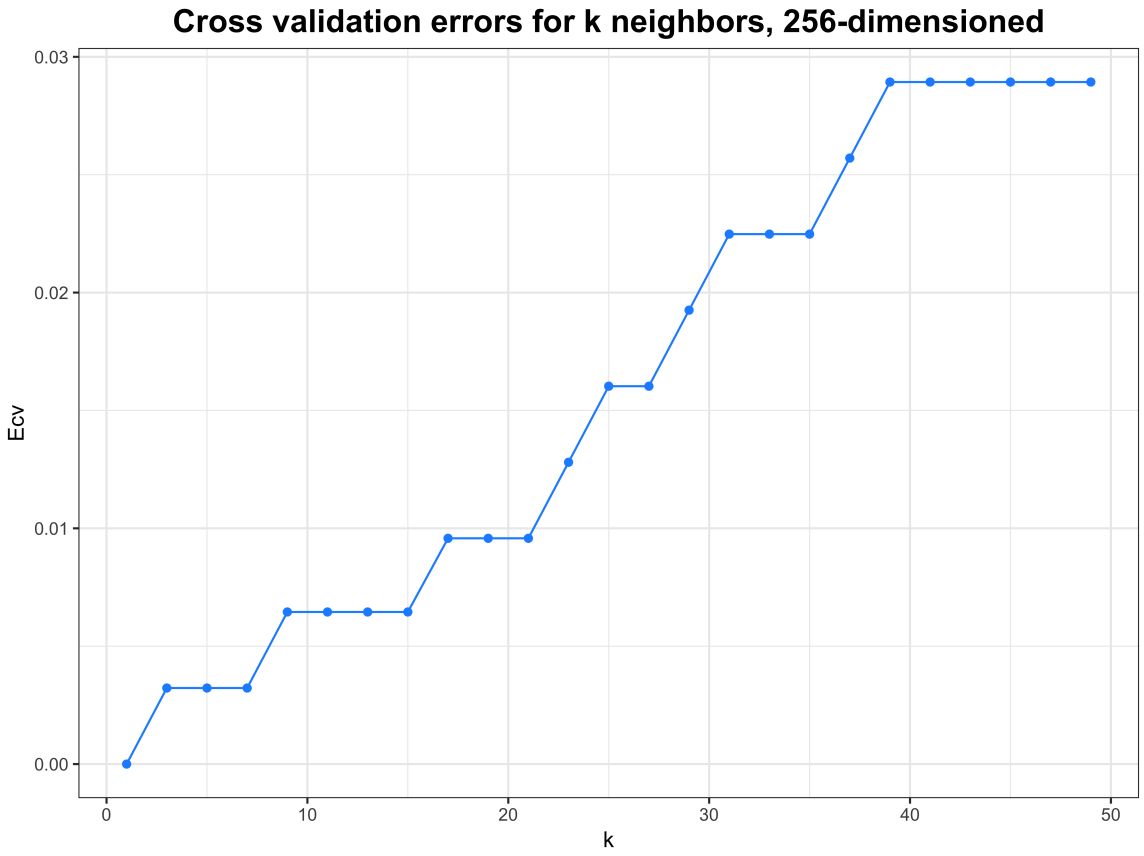

Figure 1.3. Cross validation errors for k neighbors (k taken odd numbers from 1 to 49). Original 256 dimensions were used.

	CI lower bound	CI upper bound
1	0.000	0.000
2	-0.017	0.024
3	-0.017	0.024
4	-0.017	0.024
5	-0.021	0.034
6	-0.021	0.034
7	-0.021	0.034
8	-0.021	0.034
9	-0.021	0.040
10	-0.021	0.040
11	-0.021	0.040
12	-0.020	0.046
13	-0.018	0.050
14	-0.018	0.050
15	-0.026	0.064
16	-0.021	0.066
17	-0.021	0.066
18	-0.021	0.066
19	-0.025	0.077
20	-0.028	0.085
21	-0.028	0.085
22	-0.028	0.085
23	-0.028	0.085
24	-0.028	0.085
25	-0.028	0.085

Table 1: 95 % Confidence interval lower bound and upper bound for odd k from 1 to 49.

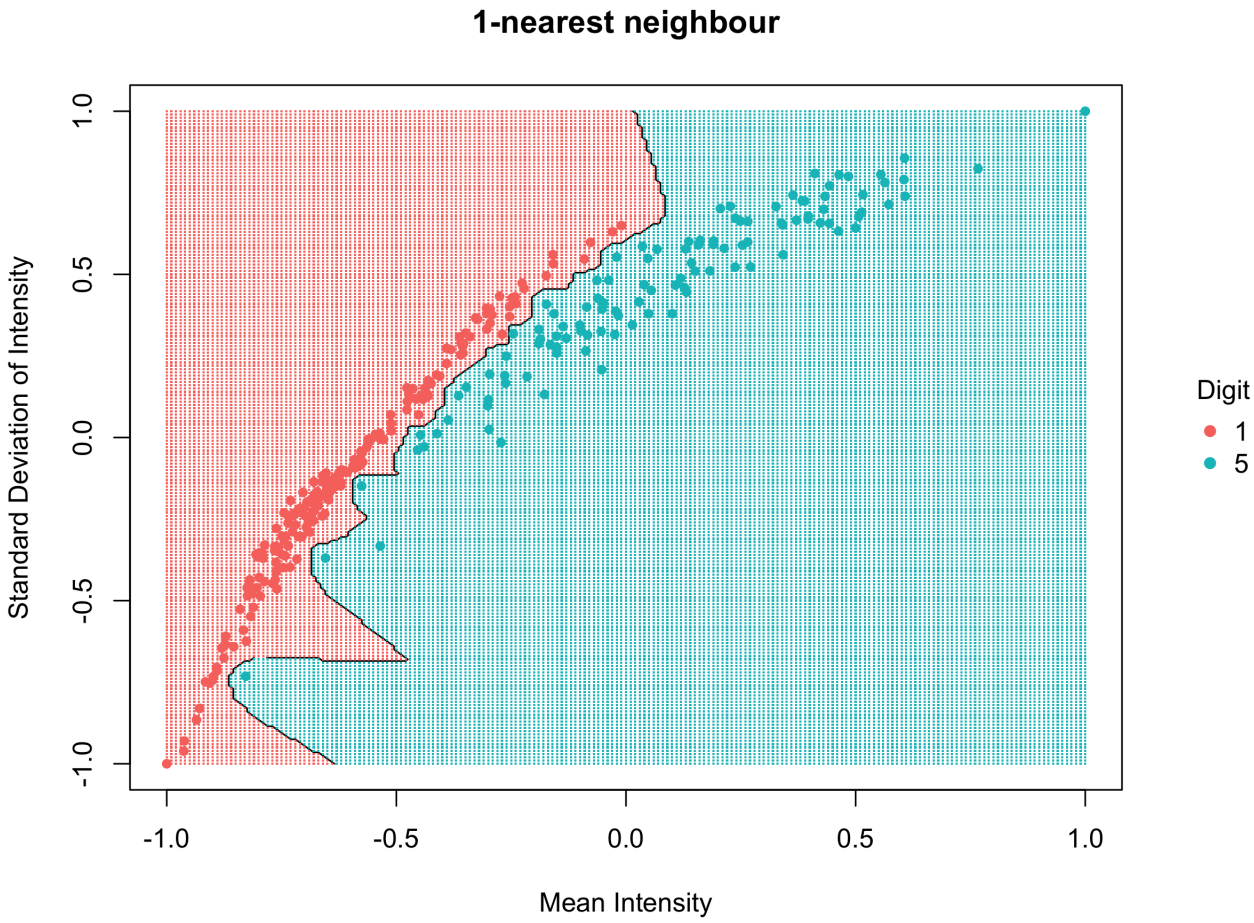

Figure 1.4. 1-Nearest neighbor in 2-dimensional space. k = 1 minimizes Ecv.

to-sit, stand-to-lie, and lie-to-stand), which can be more broadly classified into 3 activity types: static (standing, sitting, lying), dynamic (walking, walking downstairs, walking upstairs), and postural transitions (the remainder of the activities). Data was collected for each subject performing each activity multiple times. For each static and dynamic activity, the number of repetitions varied between activity and subject, but ranged from 30-90 repetitions per subject. For each postural transition activity. The number of repetitions ranged from 0-8 repetitions per subject per activity. For simplicity, I consolidated 12 small activity groups into three bigger groups: static, dynamic and postural transitions. I plotted the data set based on the two features "fBodyGyro-SMA-1" and "fBodyAcc-SMA-1", which generates the biggest eigenvalues from Principal Component Analysis (PCA) (**Figure 1.5**).

k-nearest neighbor models were generated from all the 561 features. Ecv vs k curve was generated and plotted in **Figure 1.6**. When k is small, the error is pretty big, and while k increases, the error drops and stabilized for a big range of k. k = 13 minimizes the cross validation error.

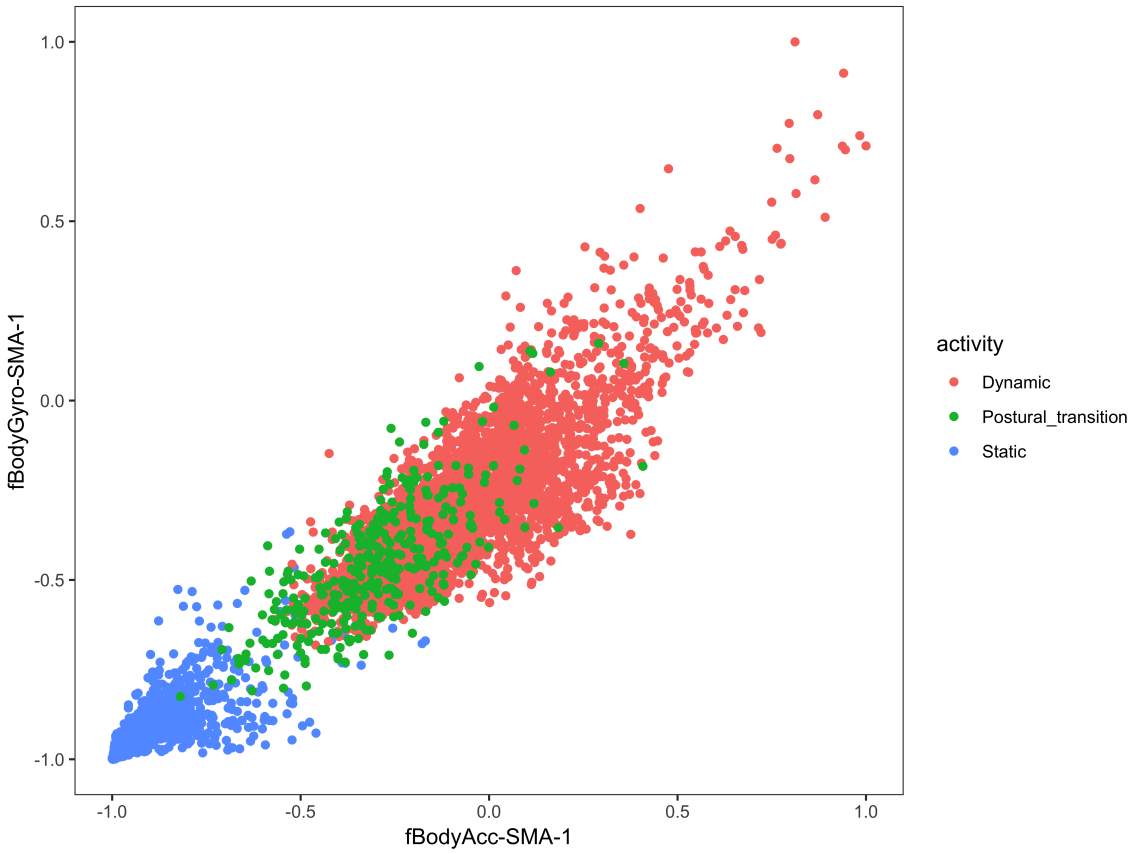

Figure 1.5. Training data represented by two features "fBodyGyro-SMA-1" and "fBodyAcc-SMA-1". Activity groups are labeled; red: dynamic, green: static, blue: postural transitions.

5.2 Glass Identification Data Set

Glass data set was downloaded from UCI Machine Learning Repository. It is a simple data set that consists 9 attributes for 7 different glass types. The study of classification of types of glass was motivated by criminological investigation. At the scene of the crime, the glass left can be used as evidence and therefore it is important to classify the glasses correctly. Again, for simplicity, I plotted two attributes Refractive Index vs. Calcium composition of the glass. The glass types were labeled in different color in **Figure 1.7**.

The optimal k for this data set would be 2. The error rate is pretty high, showing that knn is not very good at classifying the 7 groups of glasses. The error increases as k grows bigger.

5.3 Comparison of activity recognition data set and glass data set

The comparison of optimal k for these two different data set was listed in **Table 2**. The error for glass data set is very big, indicating that knn in this data set is not sufficient to classify different groups of glass. A small k = 2 optimizes knn in "glass" data set, while "activity recognition" data set prefers a bigger k that is equal to 14. This can be explained by the fact that in "activity recognition" data set (2-dimension), three activity groups are

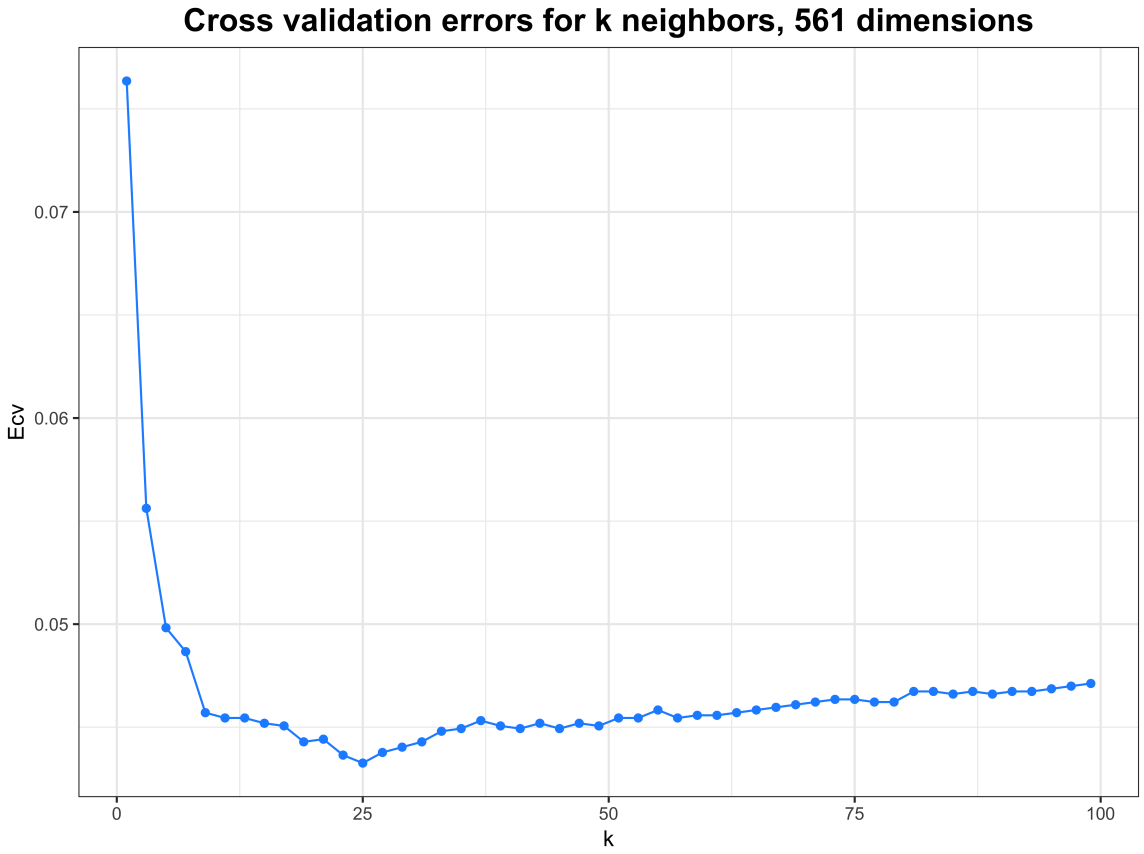

Figure 1.6. Cross validation error for odd k neighbors (k from 1 to 100). Data set: activity recognition.

separated pretty well, especially between postural transition group and static group (**Figure 1.5**). A relative big k would be more robust. When we select neighbors, if we select small k, it would cause a very big testing error. While in "glass" data set, the data are not separable in 2-dimension with feature refractive index and calcium (**Figure 1.7**). If k is too big for this data set, the training error would be extremely big. But again, knn alone is not a good choice for classifying this data set.

Data set	Optimal k	Ecv
Activity Recognition	14	0.04326017
Glass	2	0.2700047

Table 2: Comparison of optimal k for two different data set. All features were used for both of the data set.

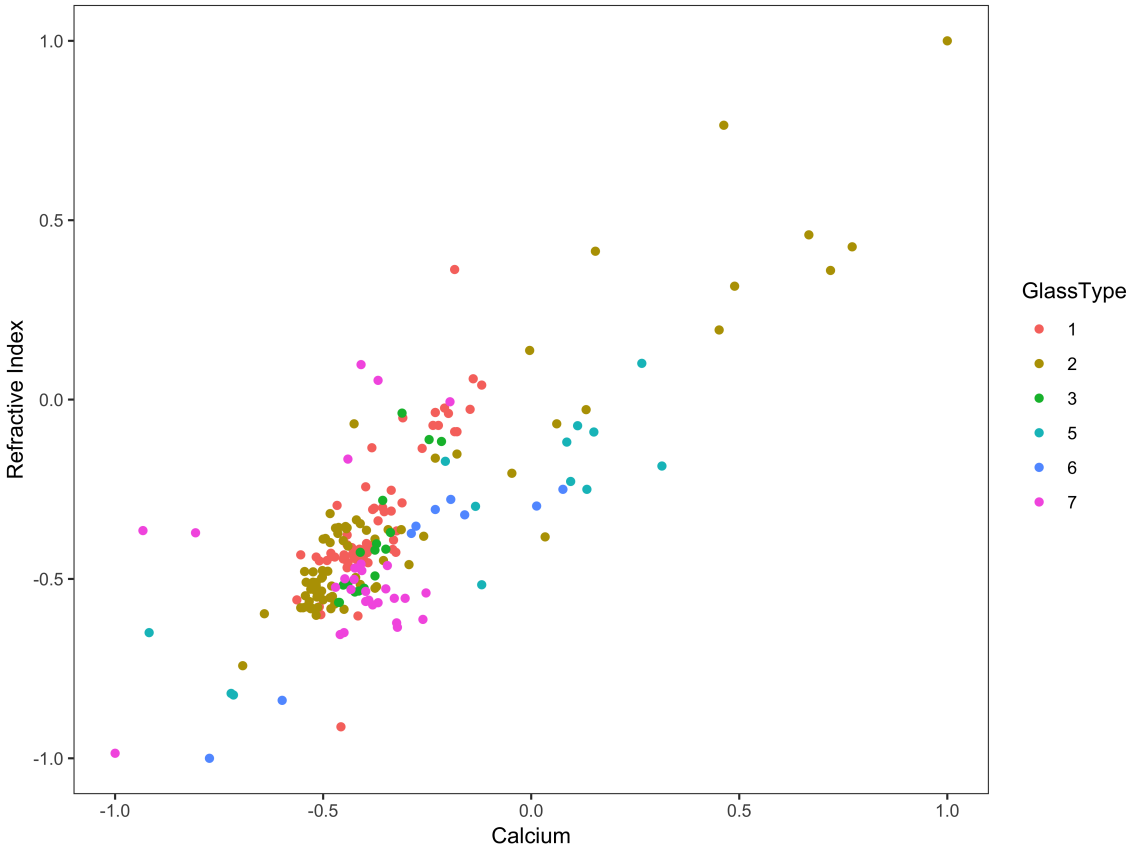

Figure 1.7. Refractive index vs. calcium for glass data set. Data set downloaded from UCI database.

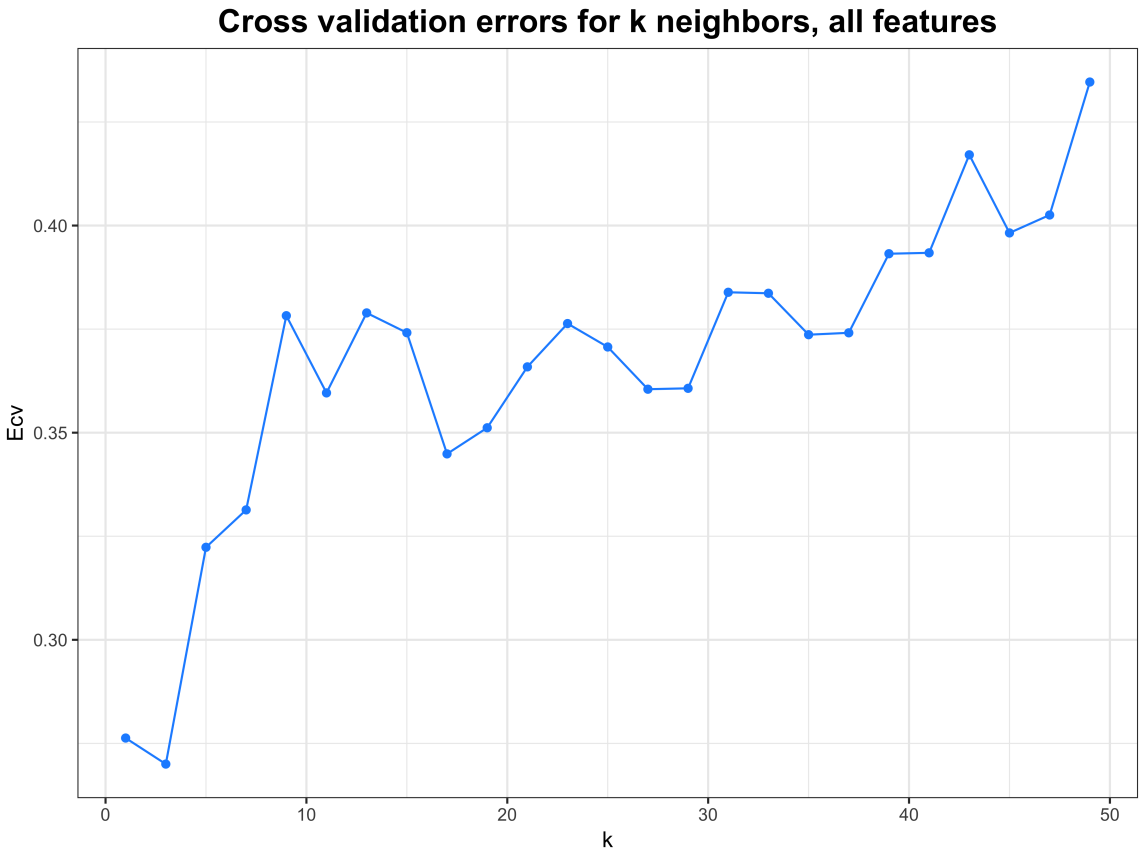

Figure 1.8. Cross validation error for odd k neighbors (k from 1 to 100). Data set: glass.