Programmimg assignment 1

Name: Ruiwan Xu

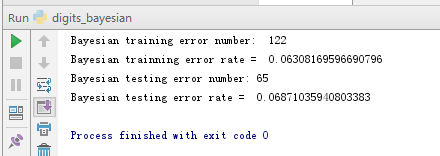
1. From the training data, pick the first sample for each digit, display them as binary images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D:\digits\trainingDigits\0.jpg | D:\digits\trainingDigits\1.jpg | D:\digits\trainingDigits\2.jpg | D:\digits\trainingDigits\3.jpg | D:\digits\trainingDigits\4.jpg |
| D:\digits\trainingDigits\5.jpg | D:\digits\trainingDigits\6.jpg | D:\digits\trainingDigits\7.jpg | D:\digits\trainingDigits\8.jpg | D:\digits\trainingDigits\9.jpg |

1. Naïve Bayesian classifier

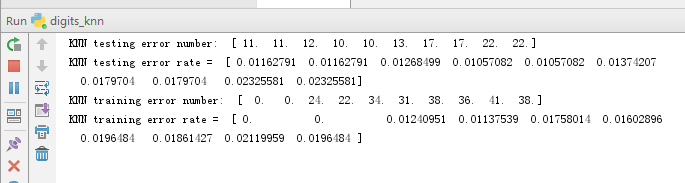
Training error rate: 0.06308169596690796

Testing error rate: 0.06871035940803383



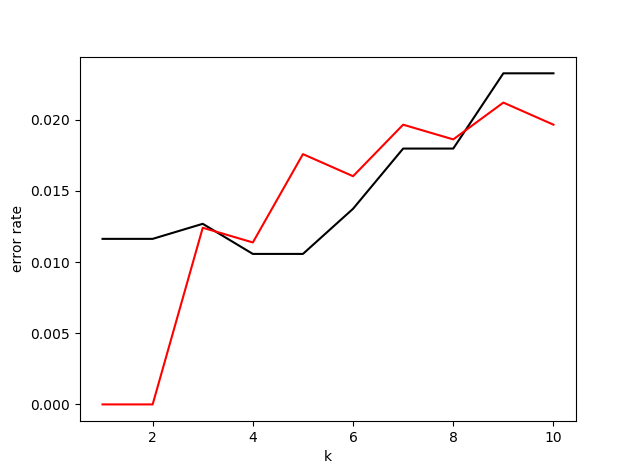
1. Testing error rate for KNN when k is from 1 to 10: [ 0.01162791 0.01162791 0.01268499 0.01057082 0.01057082 0.01374207 0.0179704 0.0179704 0.02325581 0.02325581]

Training error rate for KNN when k is from 1 to 10: [0 0 0.01240951 0.01137539 0.01758014 0.01602896 0.0196484 0.01861427 0.02119959 0.0196484 ]



For this training set, k = 4 or 5 works best.

Plot the testing error rates and training error rates with respect to k:



Where the red line is training error rate and the black line is testing error rate.

How does training error rates look different from testing error rates?

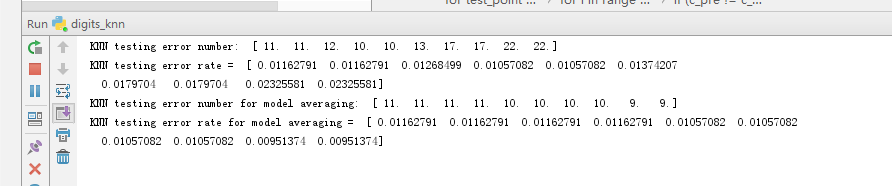
The training errors are always equal to zero when k = 1 or 2, because in the training set can always find a sample which is exactly the testing point, thus the class of the nearest neighbor is correct. When k is from 3 to 8, training error rates are greater while k = 9, 10, testing error rate is greater.

The trends of two plots:

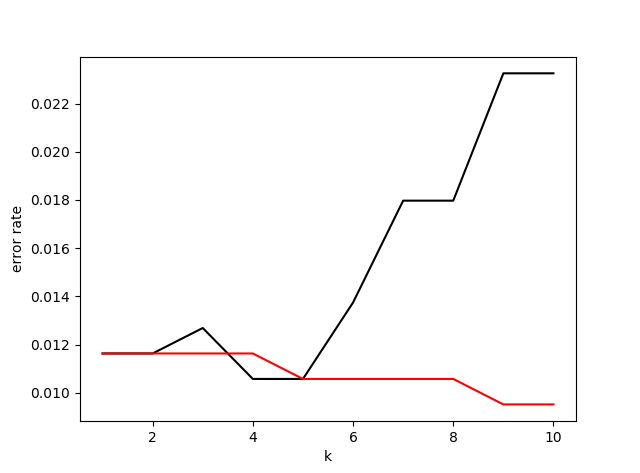
The training error rates and the testing error rates will become greater when k grows, because when k grows the KNN algorithm take more unrelated samples into consideration. When K equal to the total sample number, KNN algorithm will choose the class with highest prior probability, so the error rate will become extremely high.

Using majority voting to do the model averaging.

Testing error rates for k model averaging when k = 1 to 10: [0.01162791 0.01162791 0.01162791 0.01162791 0.01057082 0.01057082 0.01057082 0.01057082 0.00951374 0.00951374]



Plot the model averaging testing rate when k is from 1 to 10:



Where the black line is testing error rate and the red line is model averaging testing error rate.

Testing error rates after model averaging are lower than the original ones. When k grows, the testing error rate is lower. Because this model averaging use the most frequent predicted class of k models as the final class. When it considers more situations, the result is going to be more accurate.

The error rates of KNN are around 0.01-0.02 which are lower than those of NBC which are around 0.06.

Pros of NBC: 1. Compute fast 2. Works well with high dimensions 3. Simple to implement.

Cons of NBC: based on the assumption that each attributes are independent from each other.

Pros of KNN: 1. Simple to implement 2. Has no assumptions on data 3. Works well when the data is representative

Cons of KNN: 1. Compute slowly when dimension and training sets grow 2. Consider the k nearest samples as same weights 3. Must have a distance function

Ways to improve the performance of NBC:

1. NBC choose the maximal product of likelihood and prior probabilities. When we have too many attributes, the product will be too small to calculate. So, we can use the sum of log instead and choose the minimal sum of – log.
2. When an attribute value has not existed in the training set, then the posterior probability of this sample is 0 which is incorrect. So, we can add a tiny value to each value of the attribute’s likelihood to avoid this situation.
3. Apply PCA to original data and redo KNN.

When the error ratio for PCA is 0.1, the reduced dimension is 50.

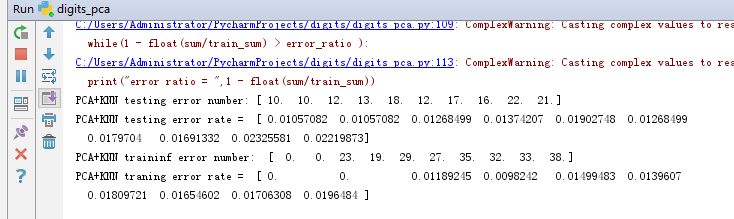
The testing error rate is: [0.01162791 0.01162791 0.01268499 0.00951374 0.01268499 0.01162791 0.01374207 0.01268499 0.01902748 0.0179704 ]

The training error rate is: [0 0 0.01137539 0.01034126 0.01292658 0.01189245 0.01602896 0.01499483 0.01654602 0.01654602]

When the error ratio for PCA is 0.05, the reduced dimension is 163.

The testing error rate is: [0.01057082 0.01057082 0.01268499 0.01374207 0.01902748 0.01268499 0.0179704 0.01691332 0.02325581 0.02219873]

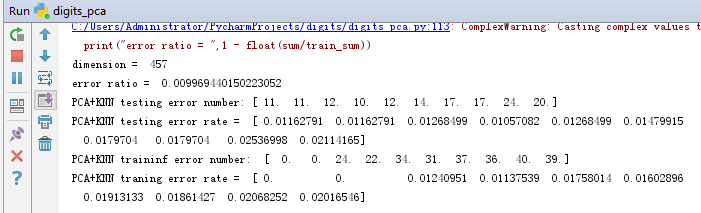
The training error rate is: [0 0 0.01189245 0.0098242 0.01499483 0.0139607 0.01809721 0.01654602 0.01706308 0.0196484 ]



When the error ratio for PCA is 0.01, the reduced dimension is 457.

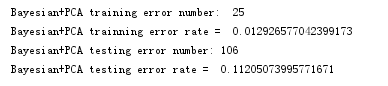
The testing error rate is: [0.01162791 0.01162791 0.01268499 0.01057082 0.01268499 0.01479915 0.0179704 0.0179704 0.02536998 0.02114165]

The training error rate is: [0 0 0.01240951 0.01137539 0.01758014 0.01602896 0.01913133 0.01861427 0.02068252 0.02016546]

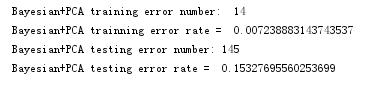


Apply PCA to original data and redo NBC.

When the error ratio for PCA is 0.1, the testing error rate is 0.11205073995771671 and the training error rate is 0.012926577042399173.

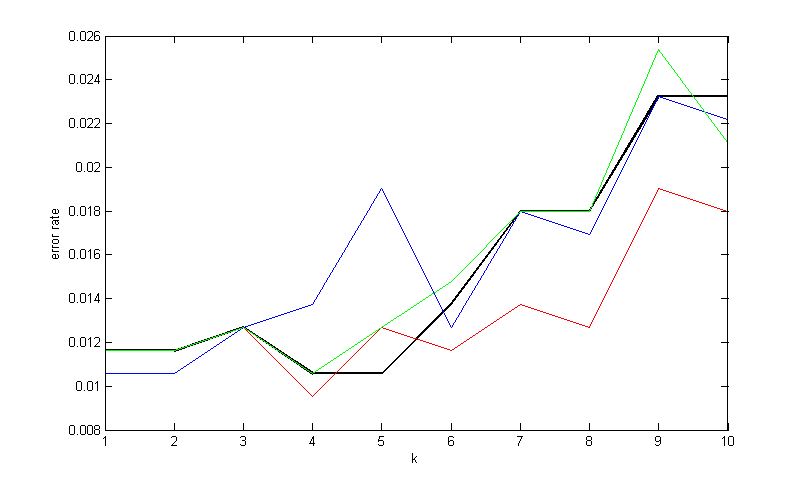


When the error ratio for PCA is 0.05, the testing error rate is 0.15327695560253699 and the training error rate is 0.007238883143743537.



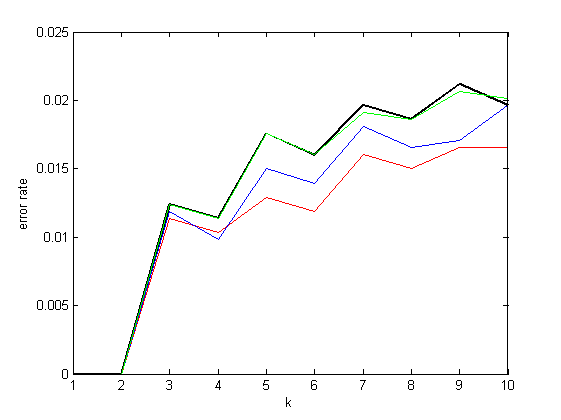
When the error ratio for PCA is 0.01, the testing error rate is 0.2917547568710359 and the training error rate is 0.011375387797311272.

Plot the testing error rate of KNN and KNN+PCA:



Where the black line is KNN testing error rates, the red, blue, green line is KNN+PCA testing error rates that the PCA error ratio is 0.1, 0.05 and 0.01.

Plot the training error rate of KNN and KNN+PCA:



The training error rates become lower and testing error rates become higher when apply PCA to NBC, especially when PCS error rate is very small.

Advantages of PCA: 1. reducing the data dimension and compute faster 2. Choose the representative features and remove data noises

Disadvantage of PCA: leading to data over fitting, the testing error rates increase and the training error rates decrease