ECE 662 MP2

Jiadao Zou

∗ Jun hua created the dataset for task 1 and I used it for task 2 as well.

There must be a problem in 4 dimensional dataset cause no matter which method was used, that would be 100 correct in testing.

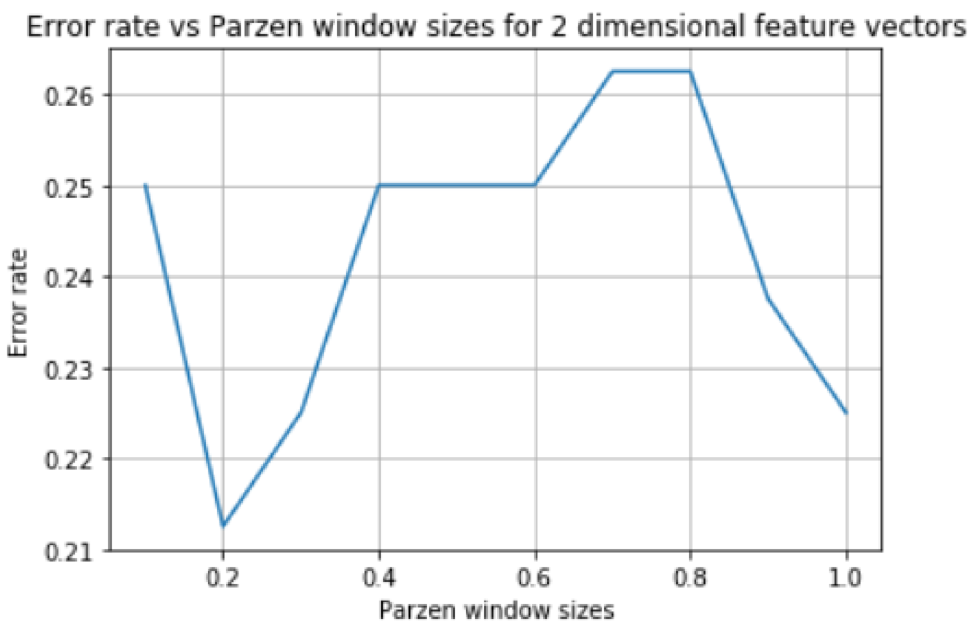
Task 1

Data generation method: Consider N-dimensional feature vectors coming from C classes. Assume that the distributions of the feature vectors for the two classes are (known) normal distributions (with same priors).

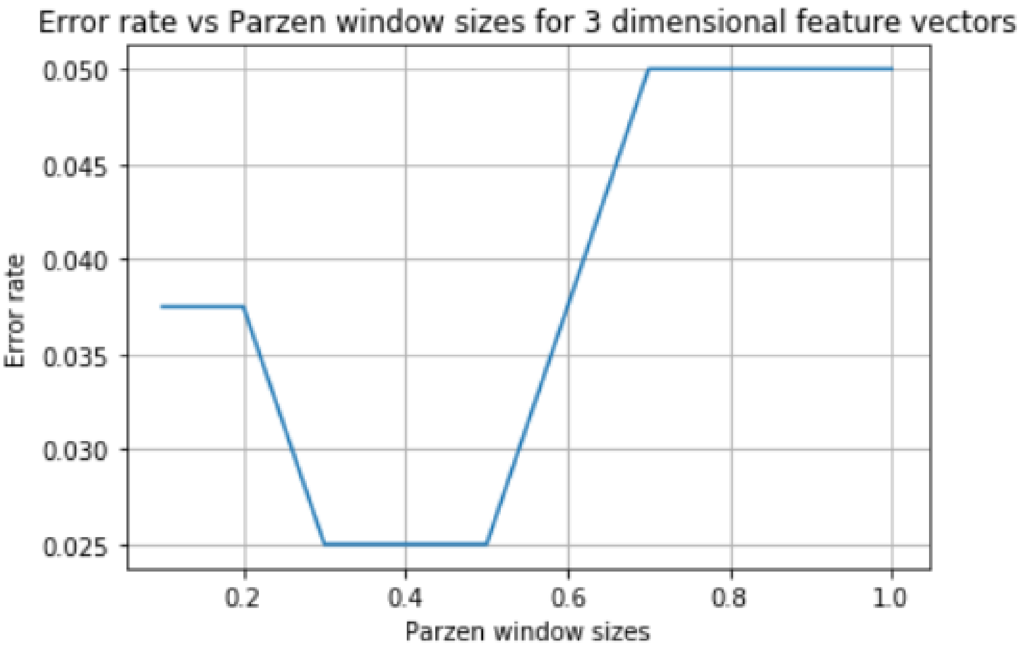
Q 1.1

C=2, 80 samples for each class (40 for training, 40 for testing): Evaluate and plot error rate (@ testing data) for varying Parzen window sizes (consider at least 4 different dimension : N)

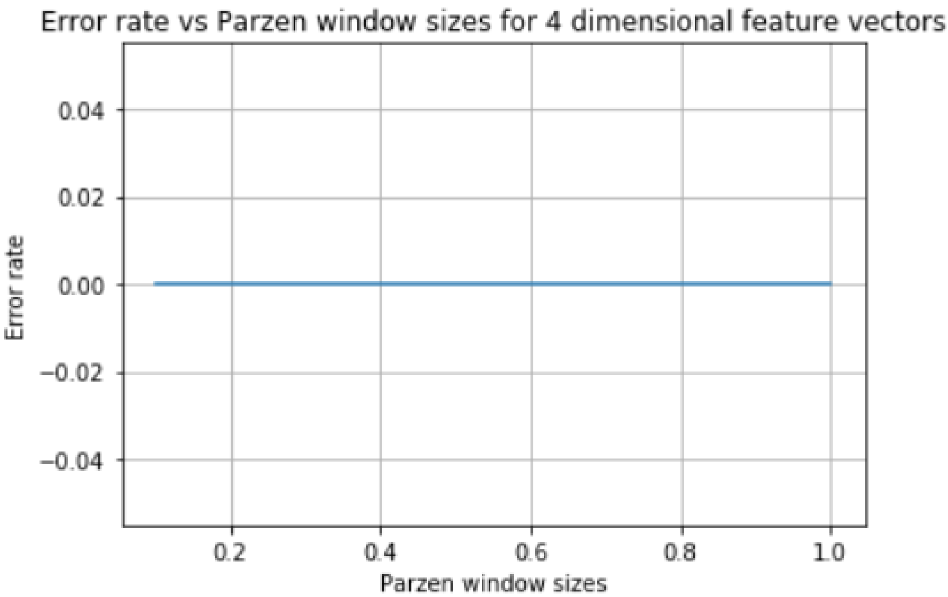
- 2 dimensional for 2 different classes



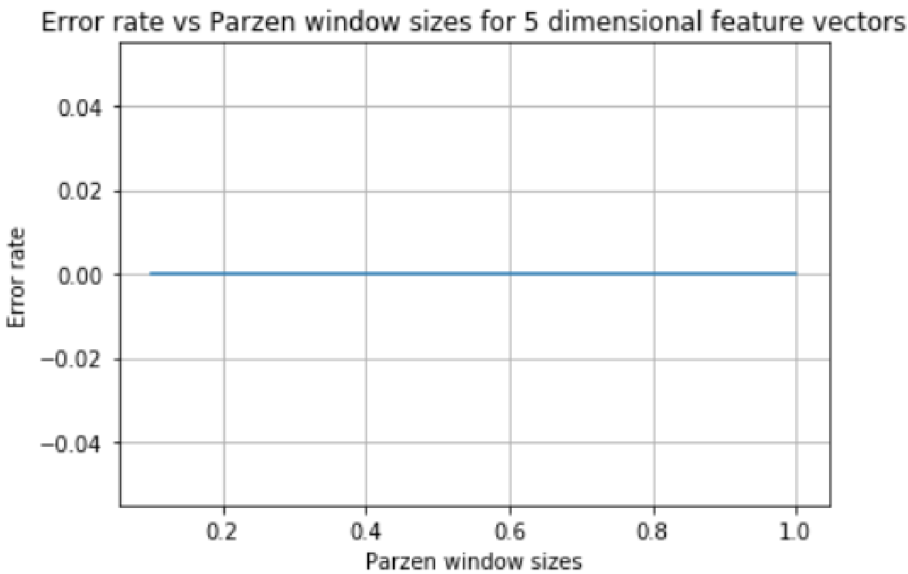
- 3 dimensional for 2 different classes



- 4 dimensional for 2 different classes



- 5 dimensional for 2 different classes

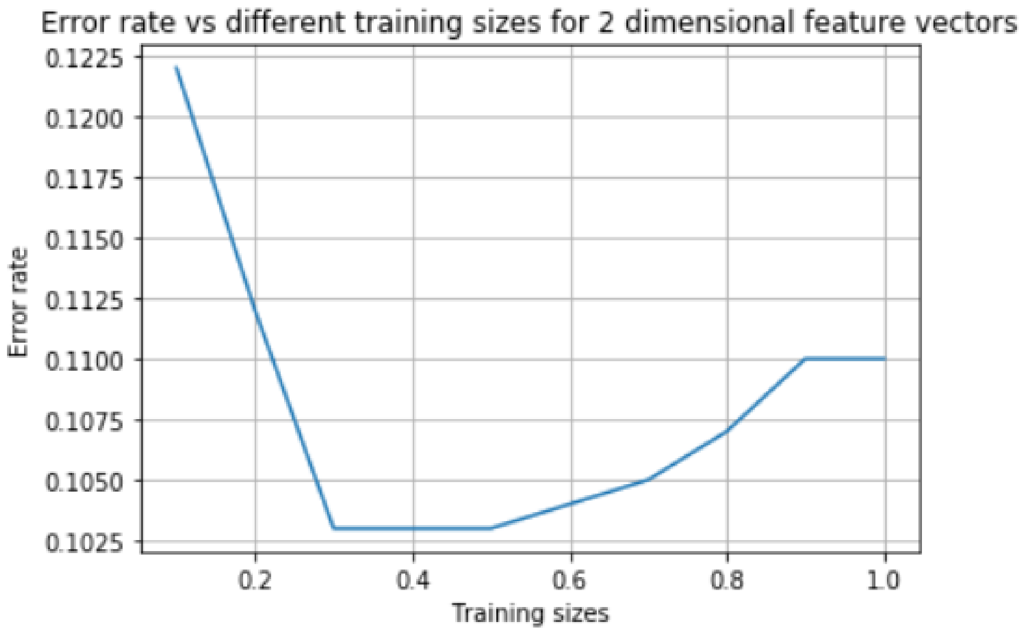


- As we could see, best Parzen window varies with the dimension of data changing. But the accuracy would going up when data dimension is increasing.

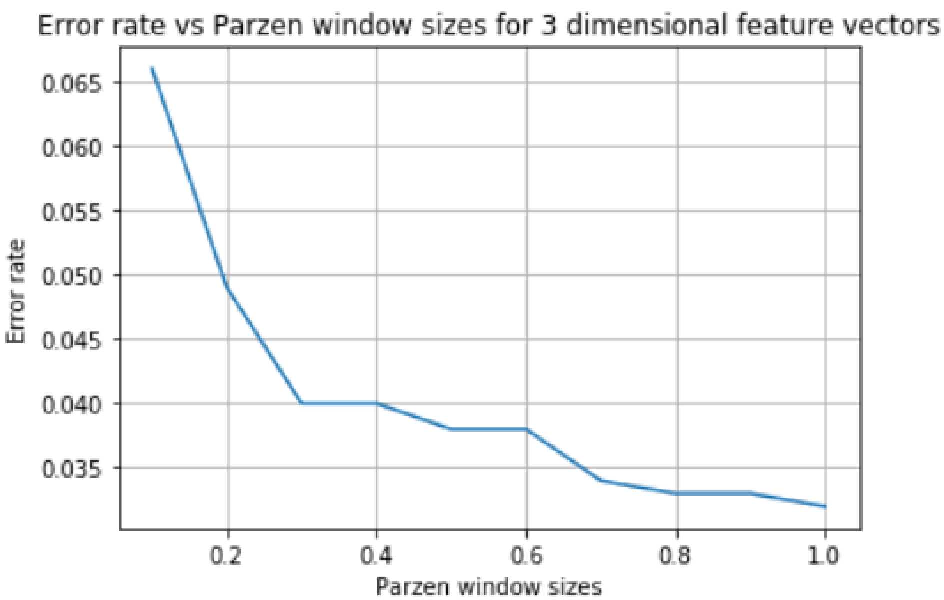
Q 1.2

C=2, 1000 samples for each class (500 for training, 500 for testing): Evaluate and plot error rate (@ testing data) for varying Parzen window sizes (consider at least 4 different dimensions : N)

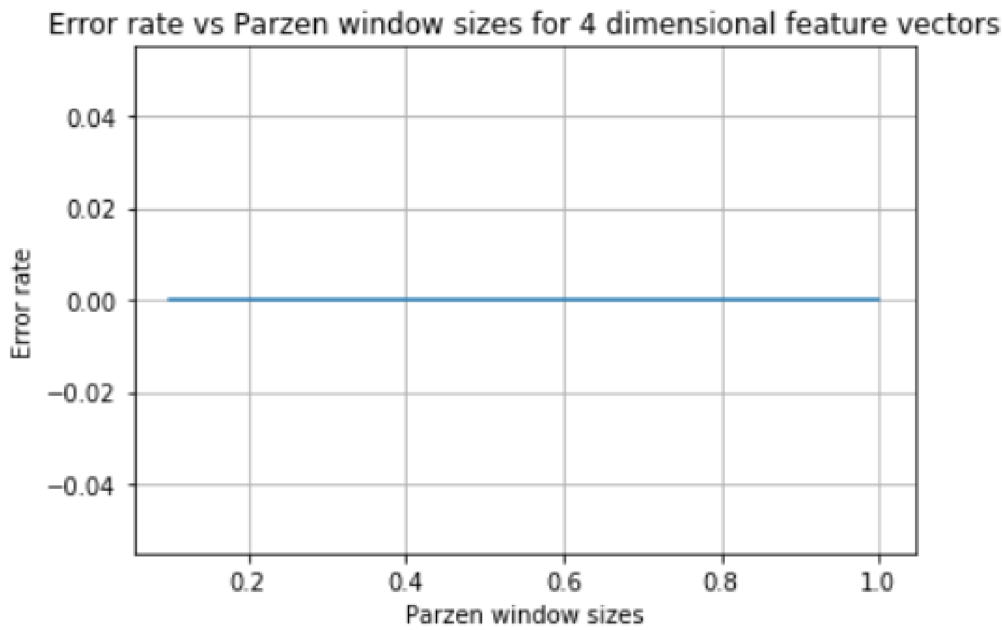
- 2 dimensional for 2 different classes



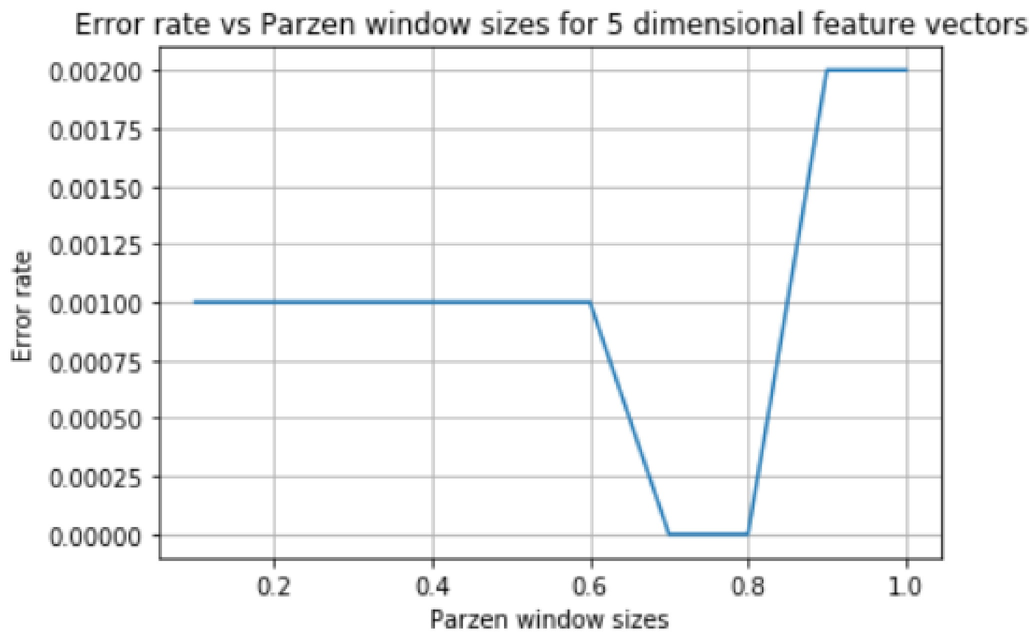
- 3 dimensional for 2 different classes



- 4 dimensional for 2 different classes



- 5 dimensional for 2 different classes

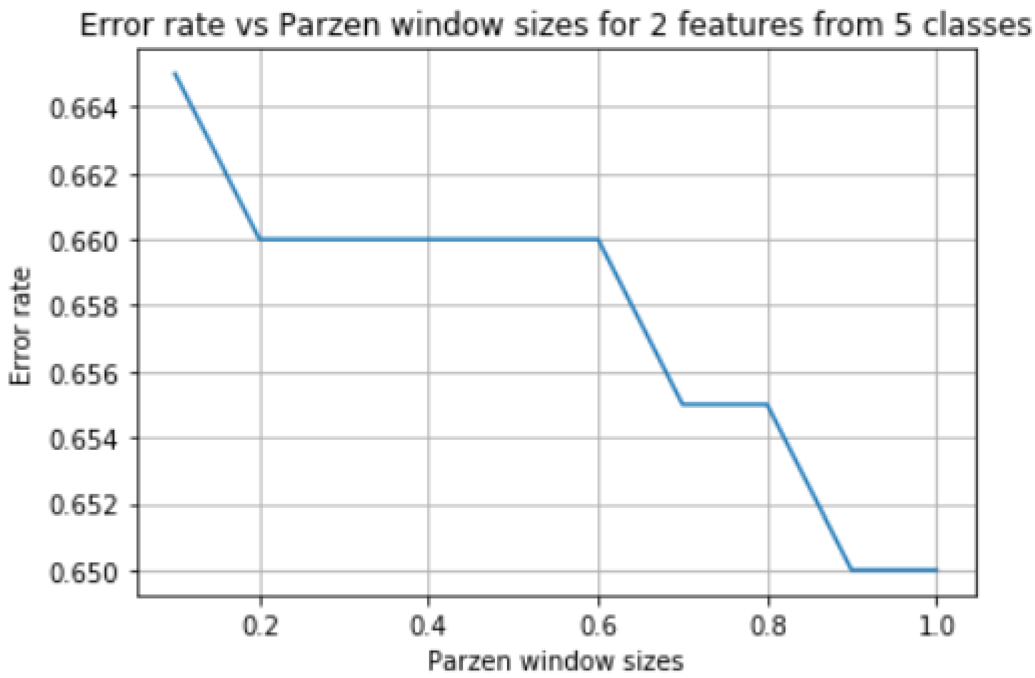


- As we could see, best Parzen window varies with the dimension of data changing.
- This method's accuracy is much higher when sampling number jumps from 80 to 1000.

Q 1.3

C=5, N =2, 80 samples for each class (40 for training, 40 for testing): plot error rate (@ testing data) for varying Parzen window sizes to analyze the best window size.

- when $C = 5, N = 2$, and the sample number is 80:

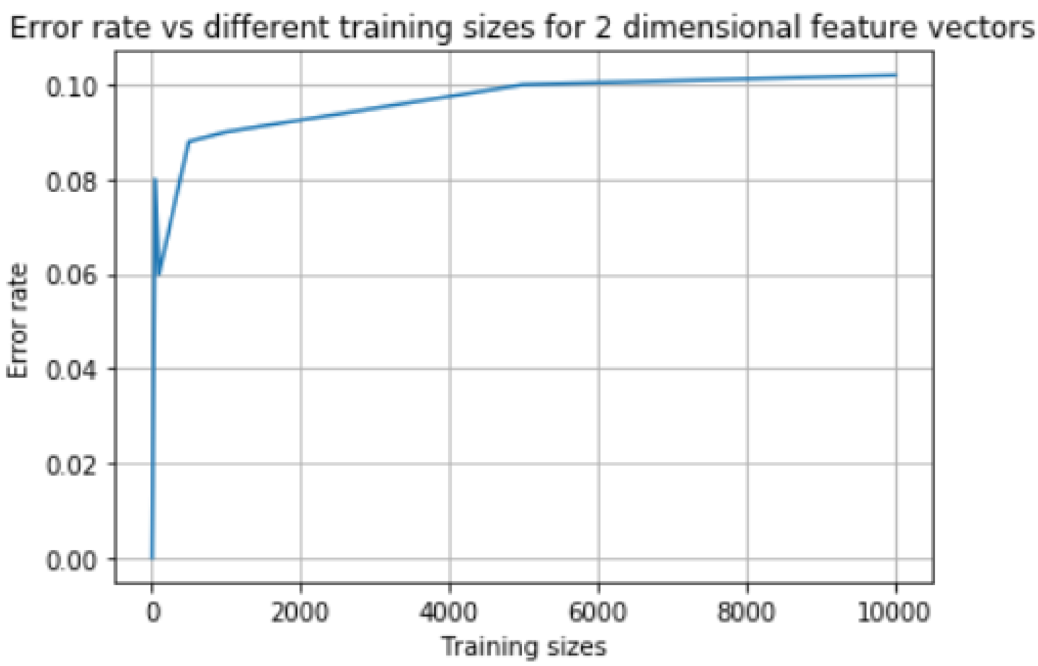


- The error rate drops with Parzen window size increasing.

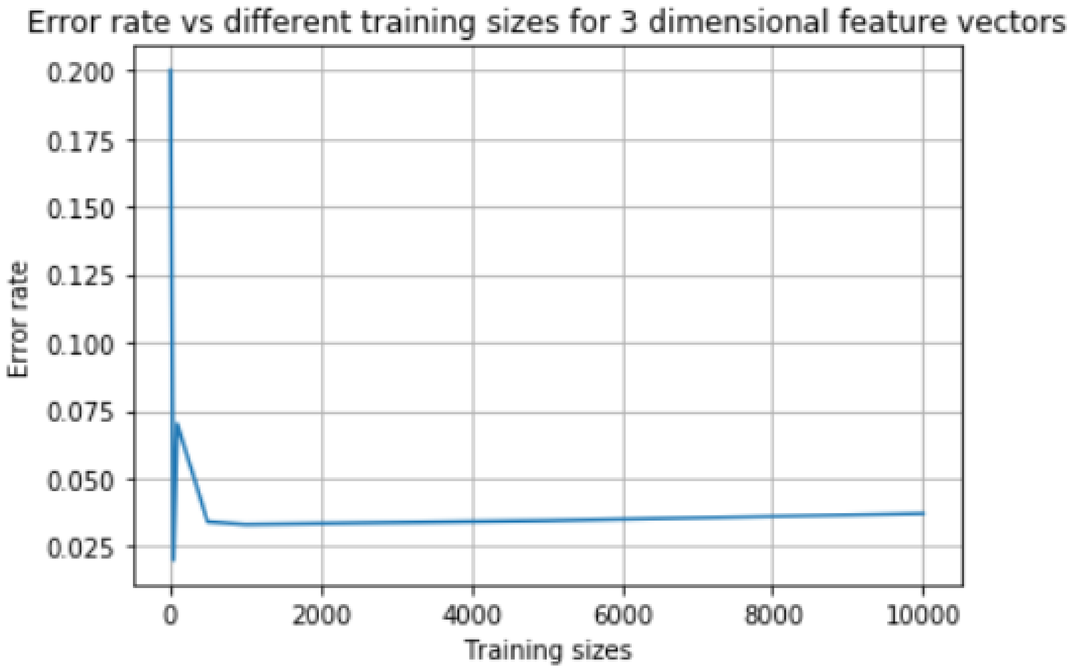
Q 1.4

Analyze how # of training samples (e.g., from 10 to 10k) impact the error rate (@ testing data), with different dimension: N. You can choose other parameters based on your own need.

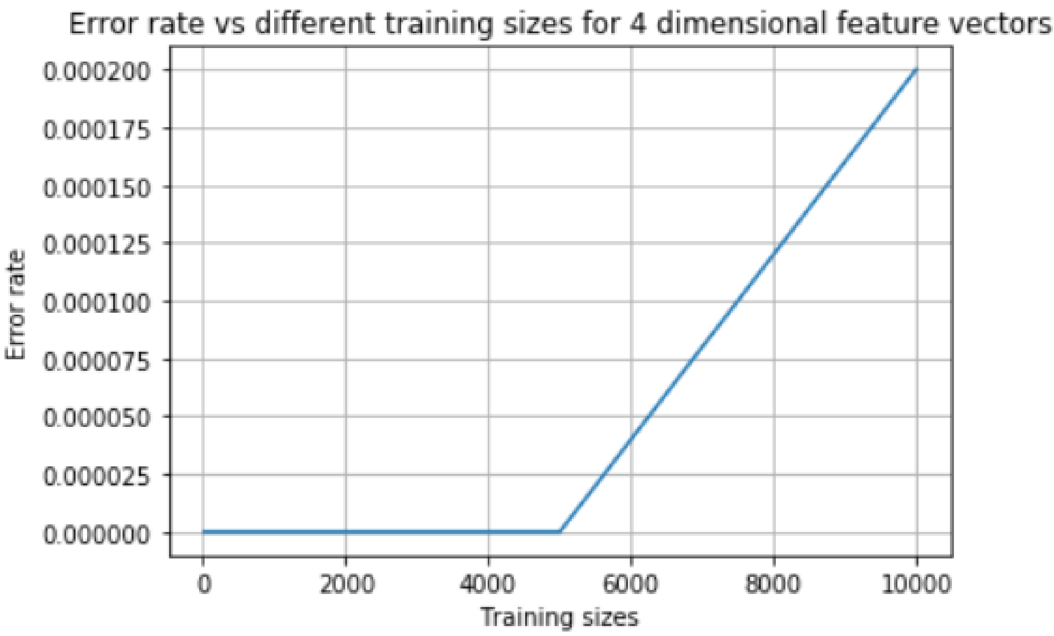
- number of samples would be [10, 50, 100, 500, 1000, 5000, 10000].
  - **2 dimensional for 2 different classes**



  - **3 dimensional for 2 different classes**



- **4 dimensional for 2 different classes**



- Seems number of training samples does not have a well defined impact in Parzen window method's accuracy.

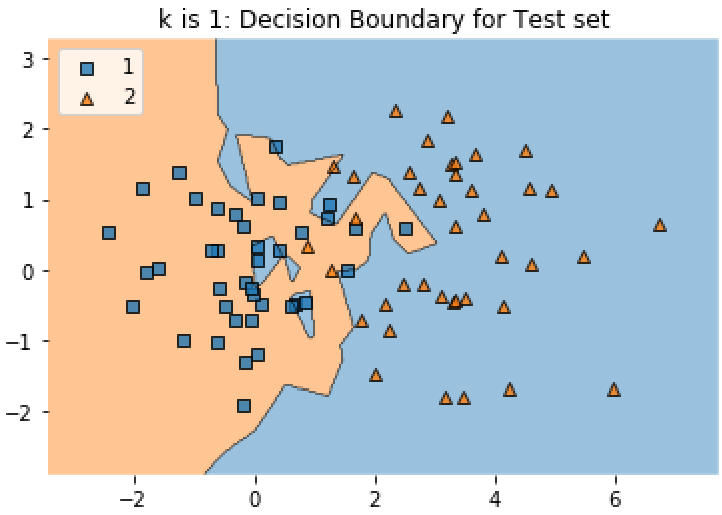
Task 2

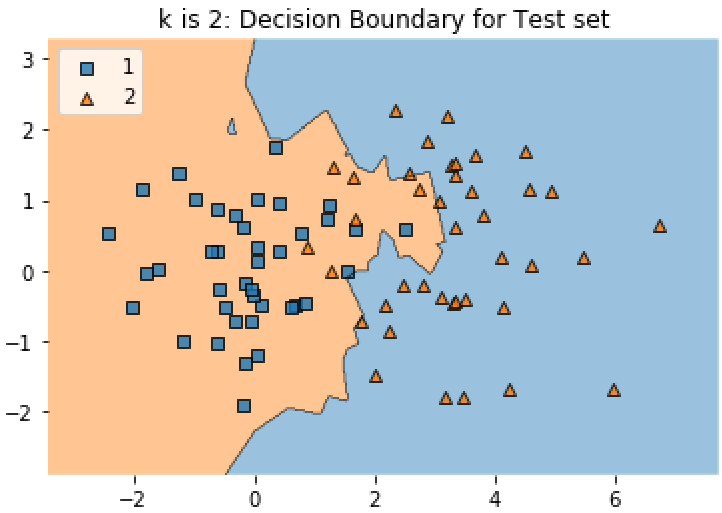
K would be [1, 2, 3, 5, 10, 20]

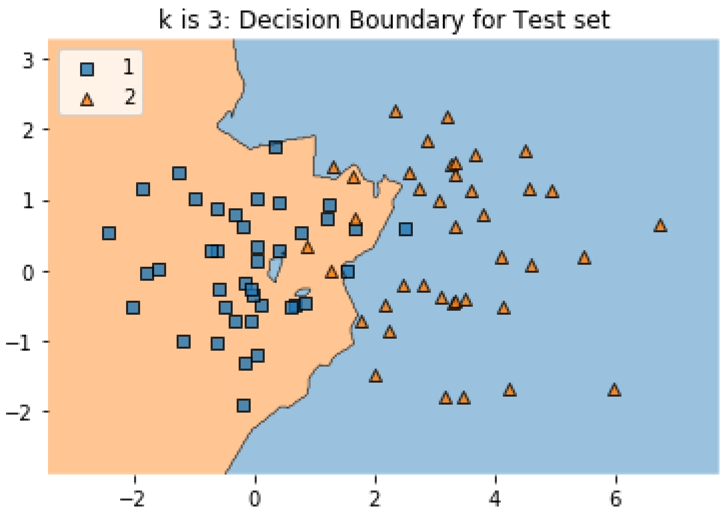
Q 2.1

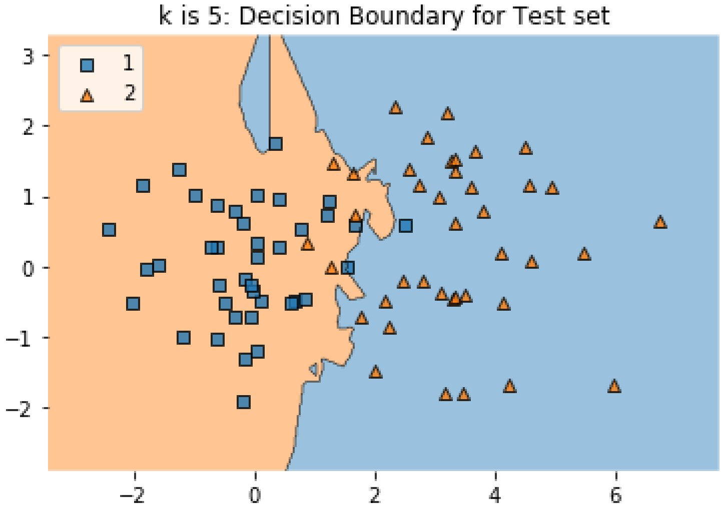
C=2, 80 samples for each class (40 for training, 40 for testing): Evaluate and plot error rate (@ testing data) for varying K (consider at least 4 different dimension : N)

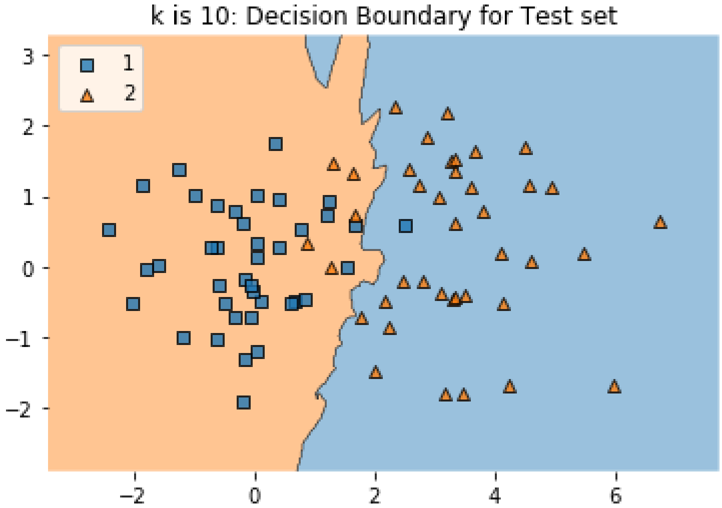
- 2 dimensional for 2 different classes

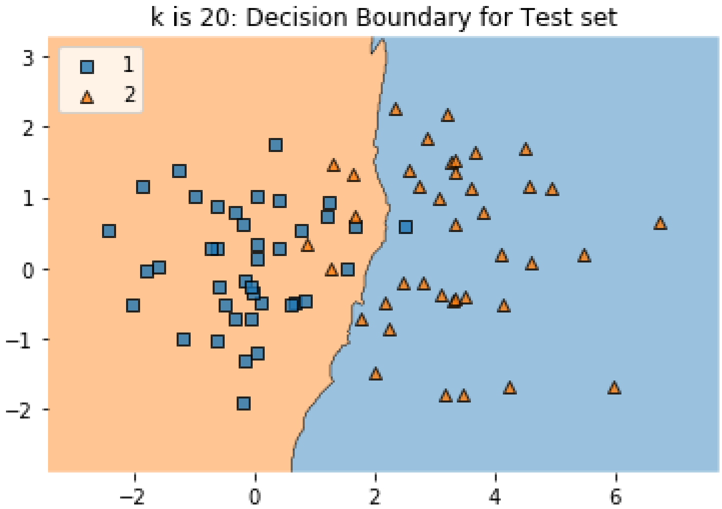


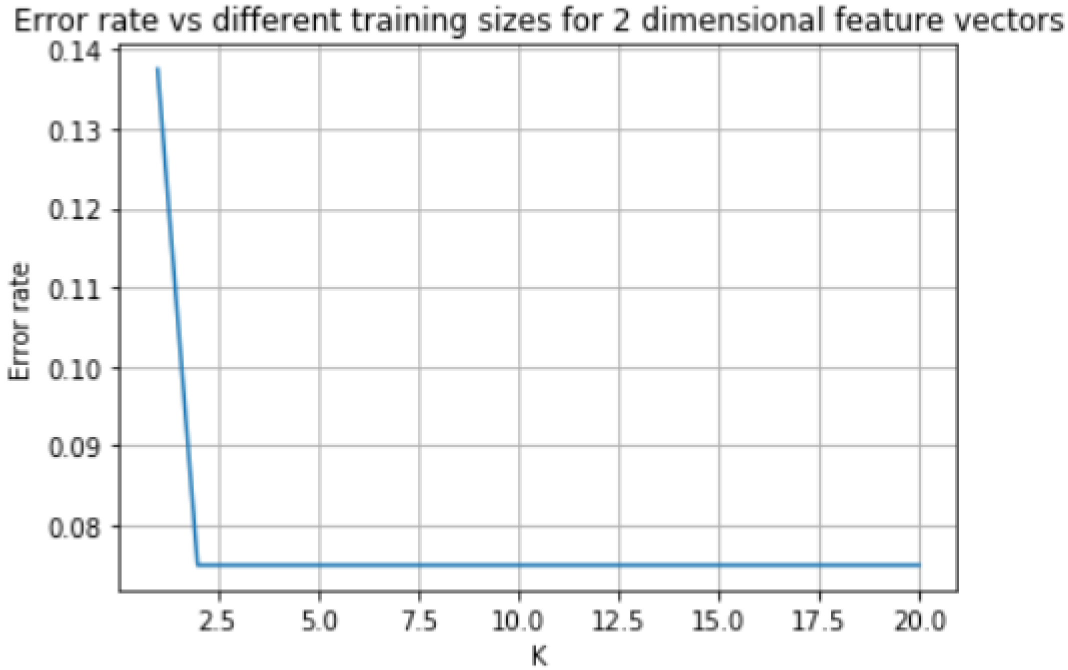




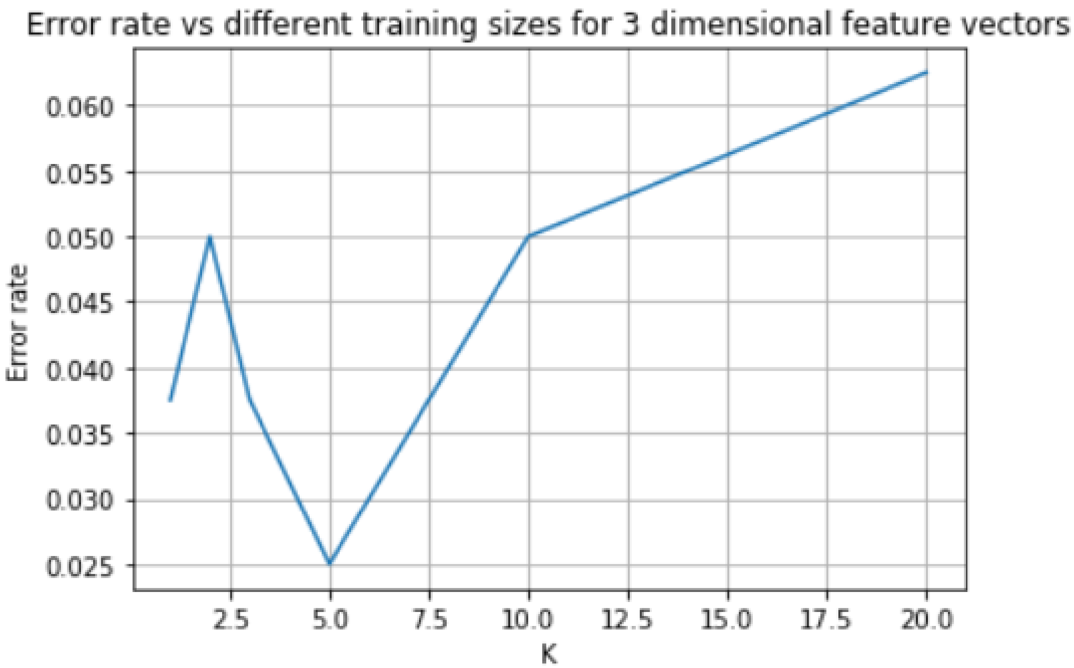

k is 5: Decision Boundary for Test set
1
2


k is 10: Decision Boundary for Test set
1
2

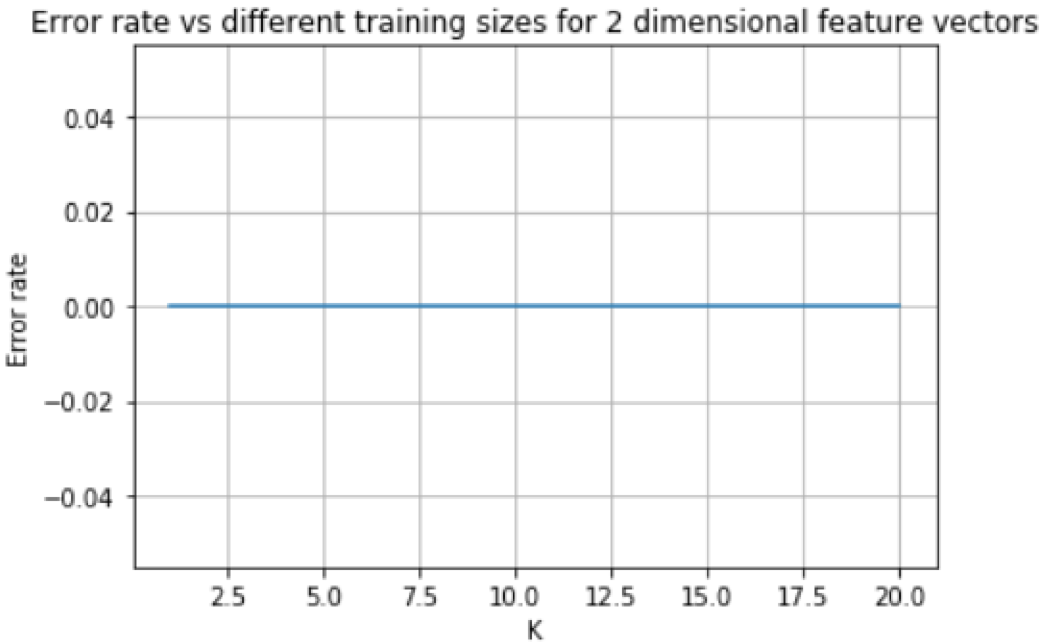

k is 20: Decision Boundary for Test set
1
2



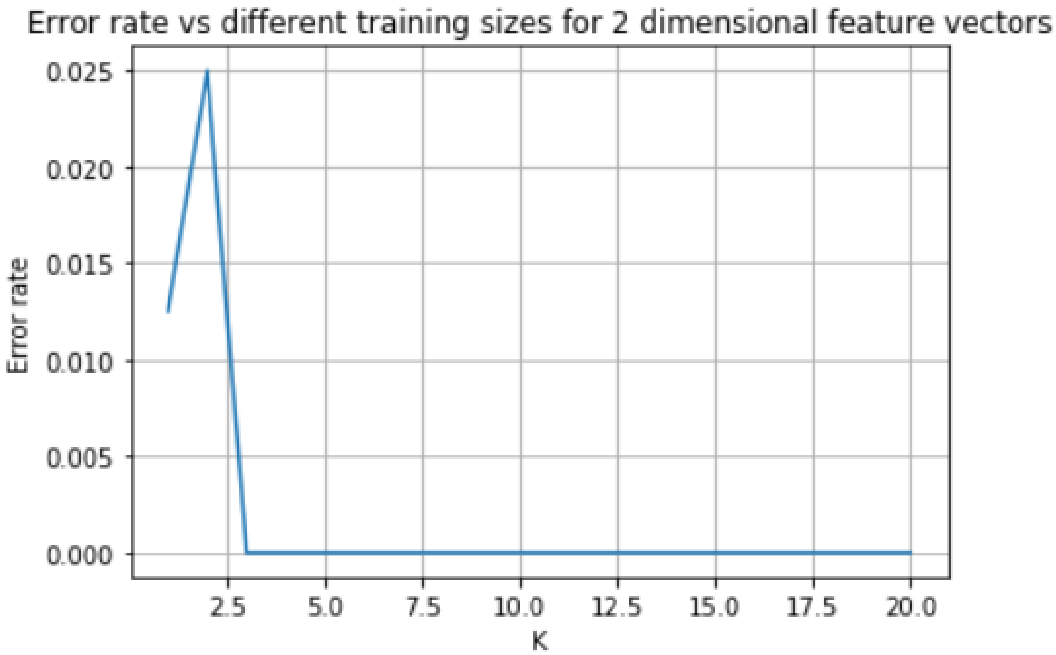
- 3 dimensional for 2 different classes



- 4 dimensional for 2 different classes



- 5 dimensional for 2 different classes

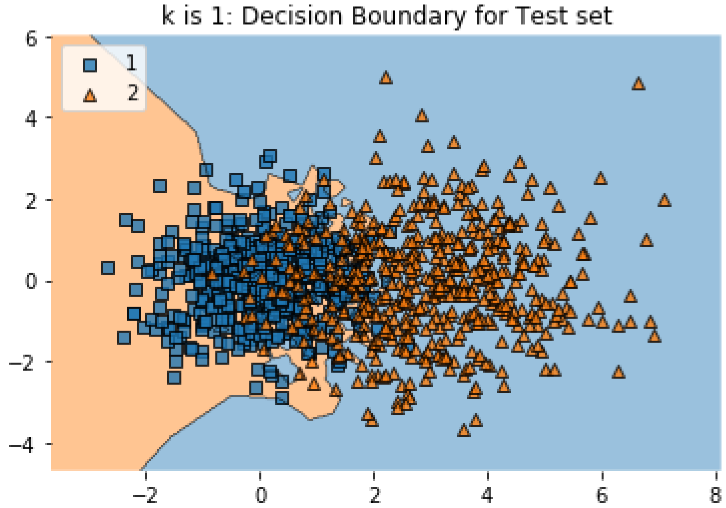


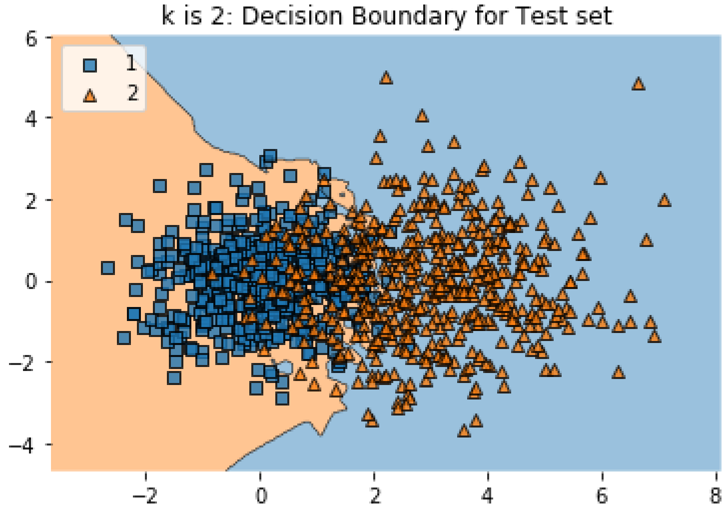
- Normal, $K \approx 5$ would achieve the best result. Other than that, for some dimension, large K would even hurt the accuracy.

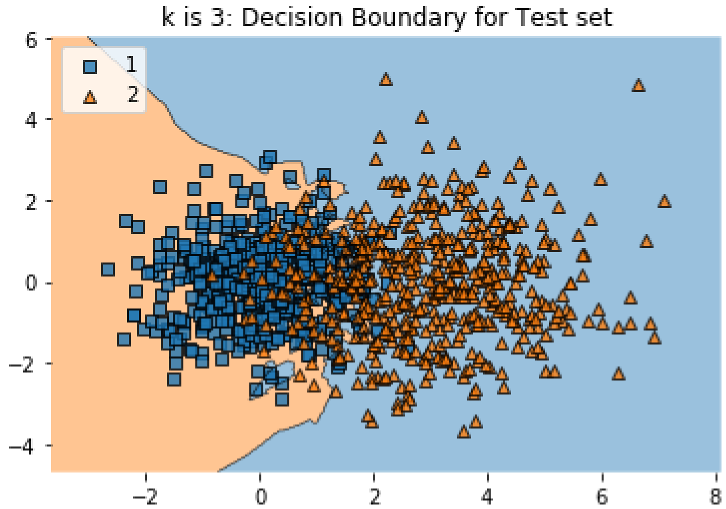
Q 2.2

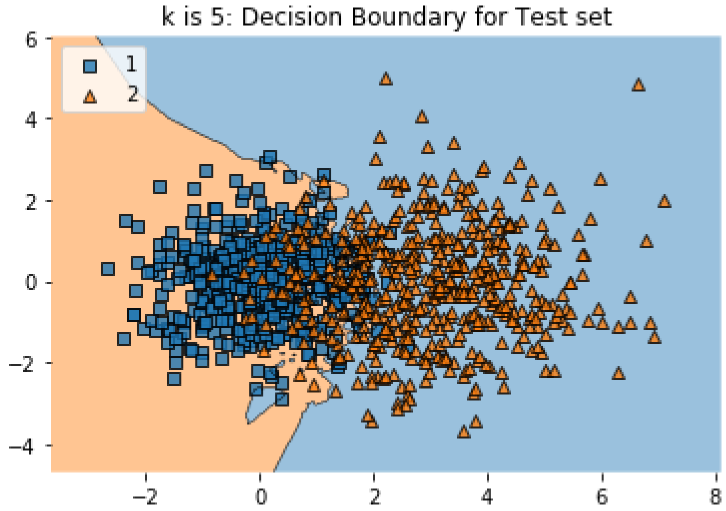
C=2, 1000 samples for each class (500 for training, 500 for testing): Evaluate and plot error rate (@ testing data) for varying K (consider at least 4 different dimension : N).

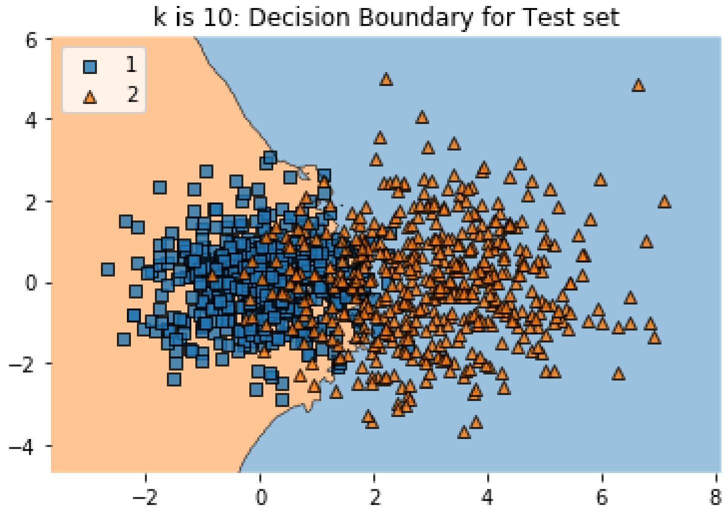
- 2 dimensional for 2 different classes

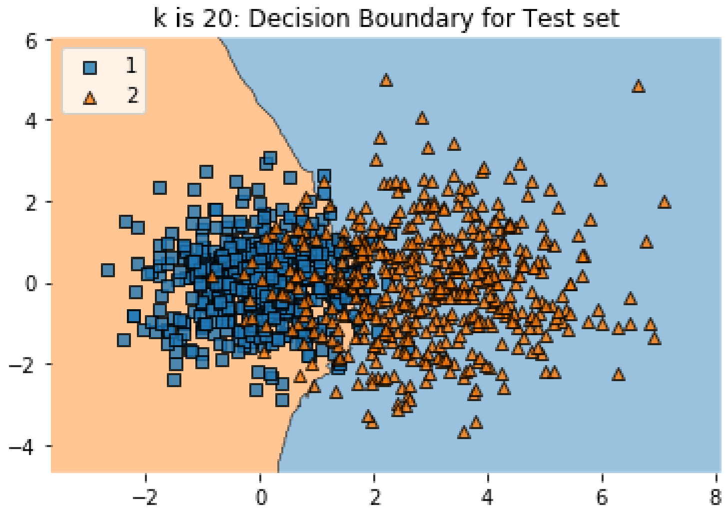

k is 1: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8

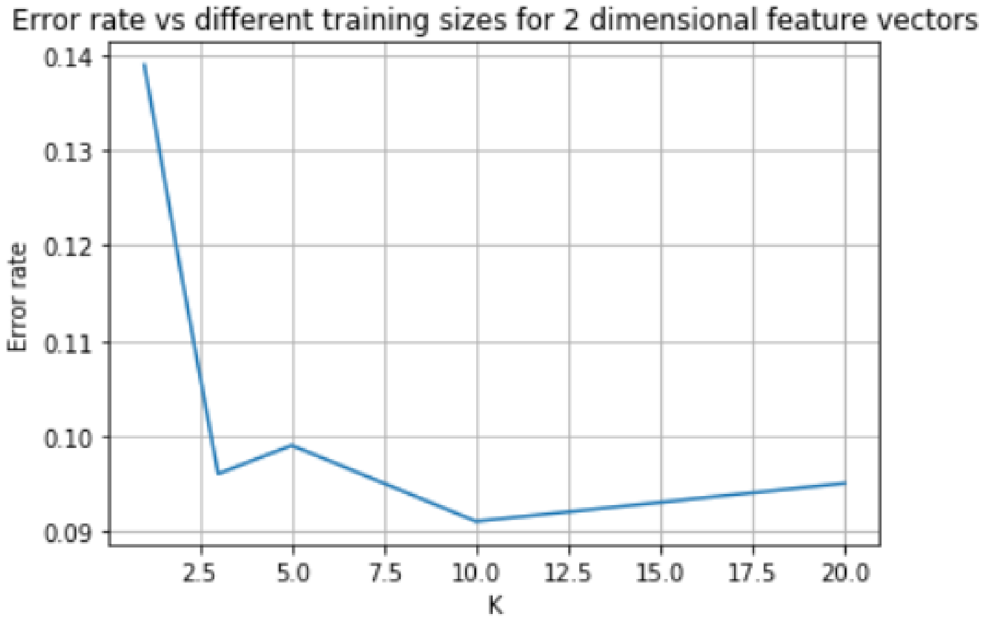

k is 2: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8


k is 3: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8

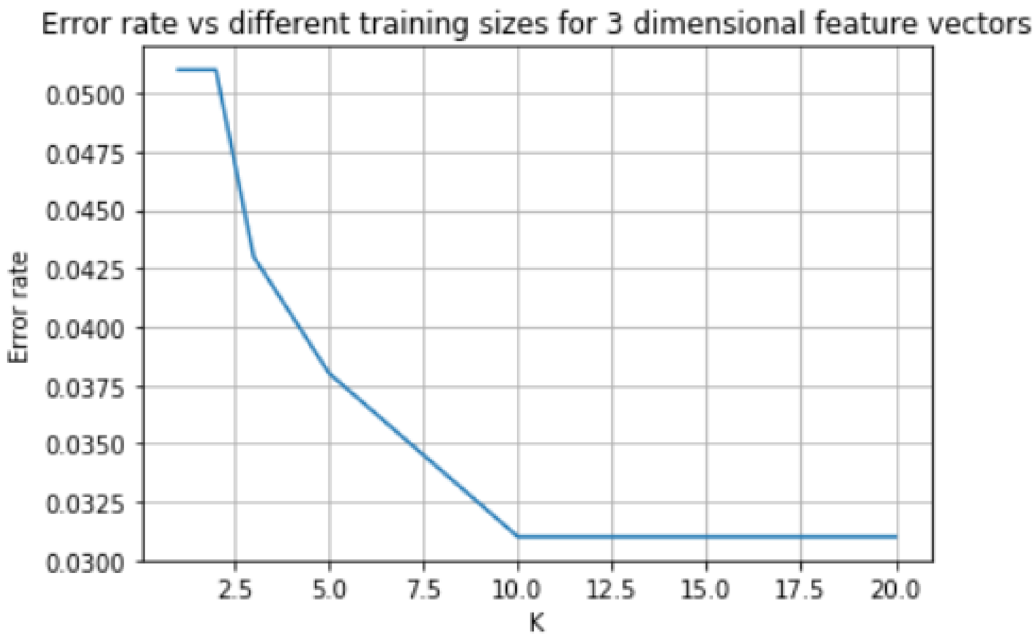

k is 5: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8


k is 10: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8

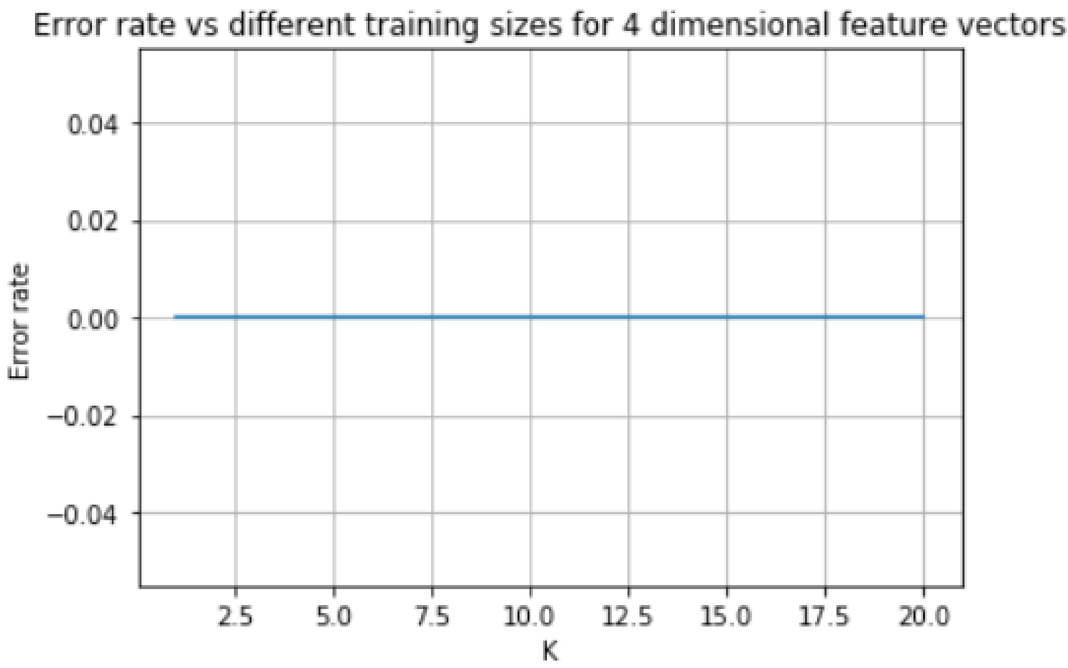

k is 20: Decision Boundary for Test set
1
2
6
4
2
0
−2
−4
−2
0
2
4
6
8



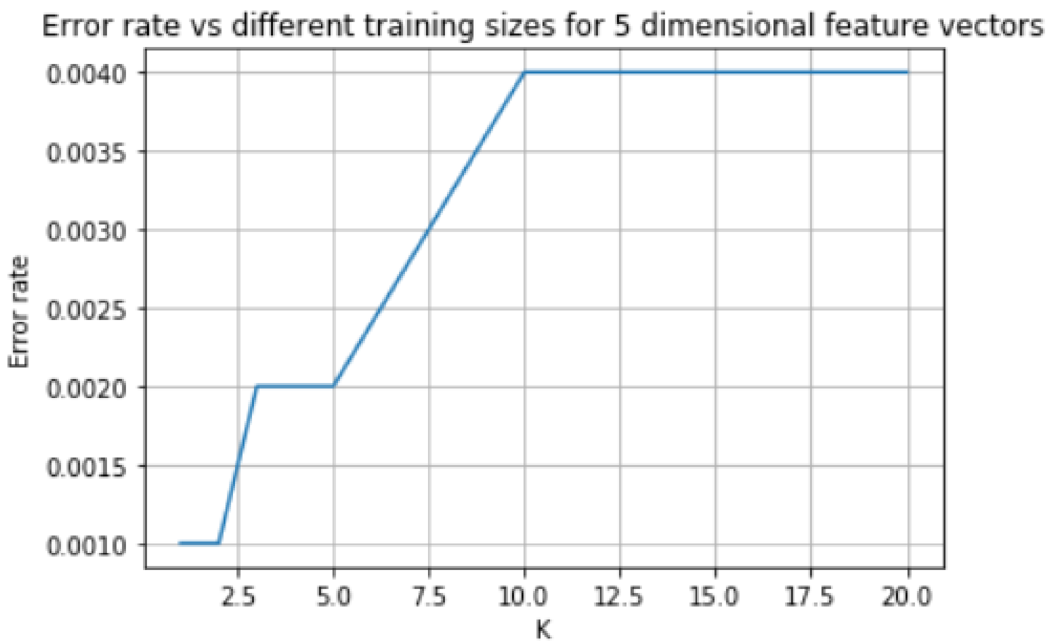
- 3 dimensional for 2 different classes



- 4 dimensional for 2 different classes



- 5 dimensional for 2 different classes

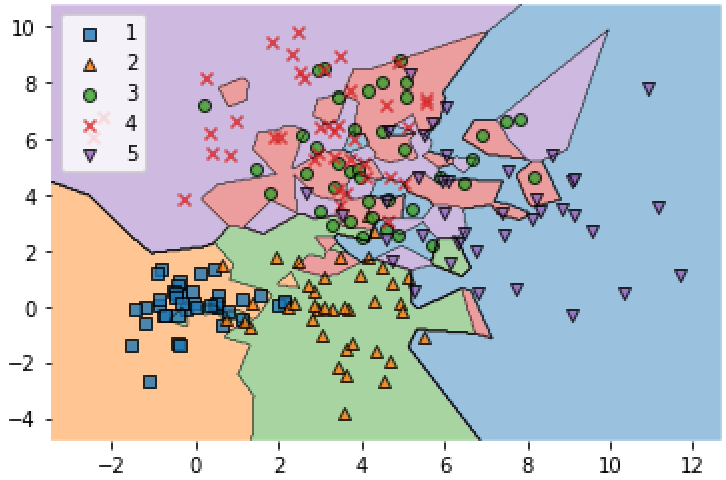


- There is no specific rule of K towards accuracy.

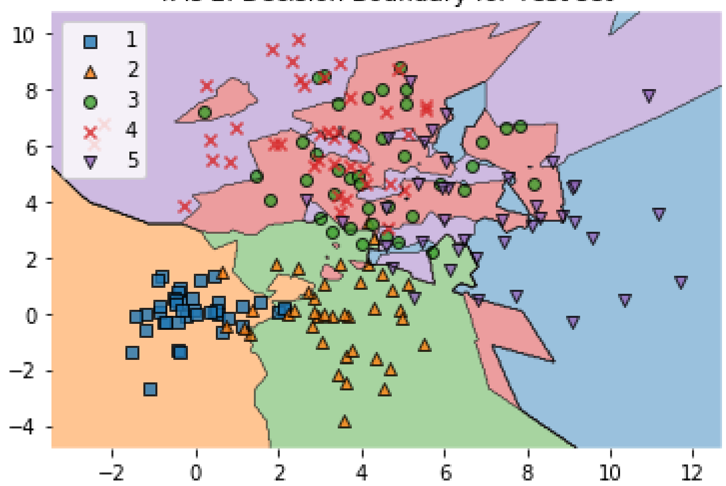
Q 2.3

C=5, N = 2, 80 samples for each class (40 for training, 40 for testing): plot error rate (@ testing data) for varying K to analyze the best K.

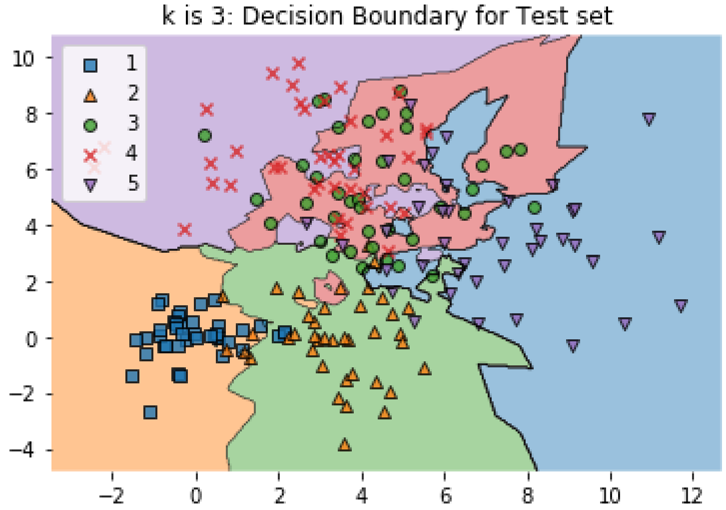
k is 1: Decision Boundary for Test set

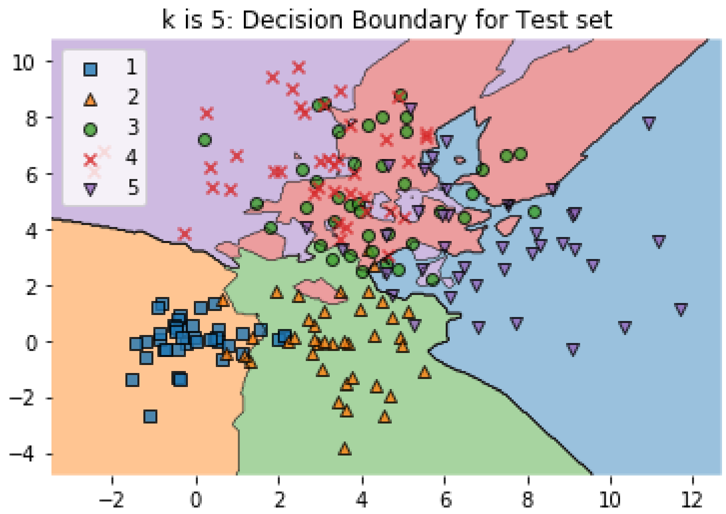


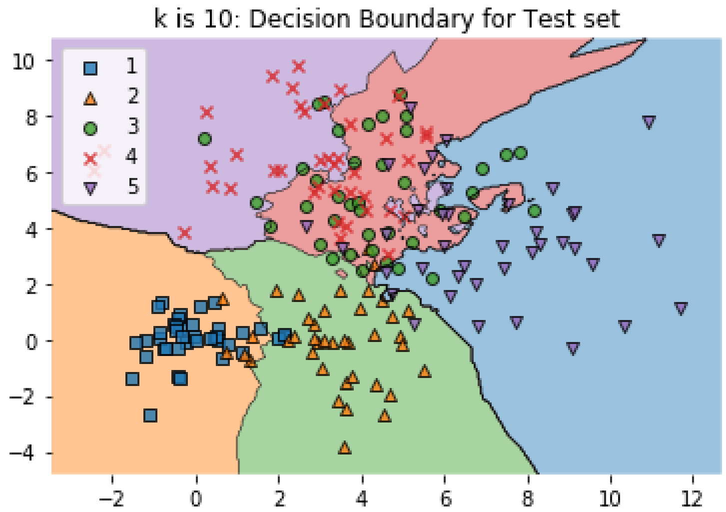
k is 2: Decision Boundary for Test set

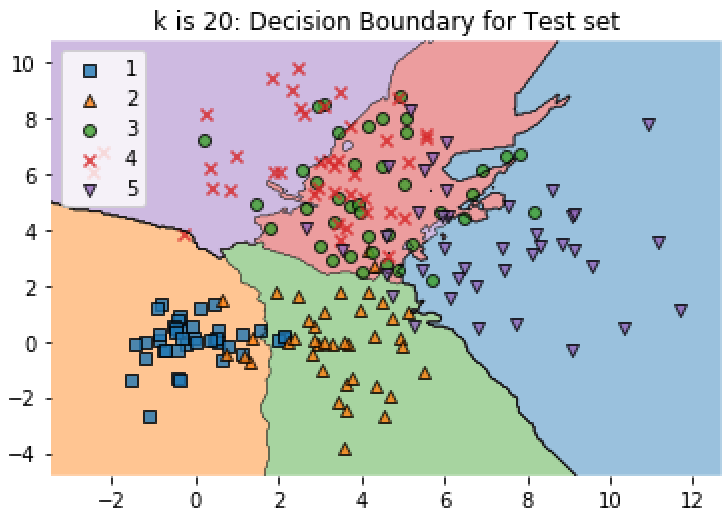


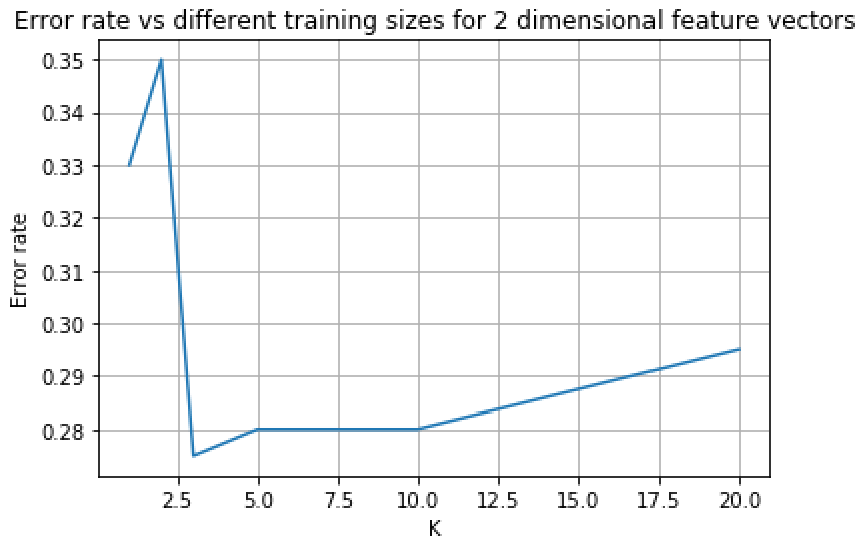
k is 3: Decision Boundary for Test set




k is 5: Decision Boundary for Test set
1
2
3
4
5


k is 10: Decision Boundary for Test set
1
2
3
4
5


k is 20: Decision Boundary for Test set
1
2
3
4
5



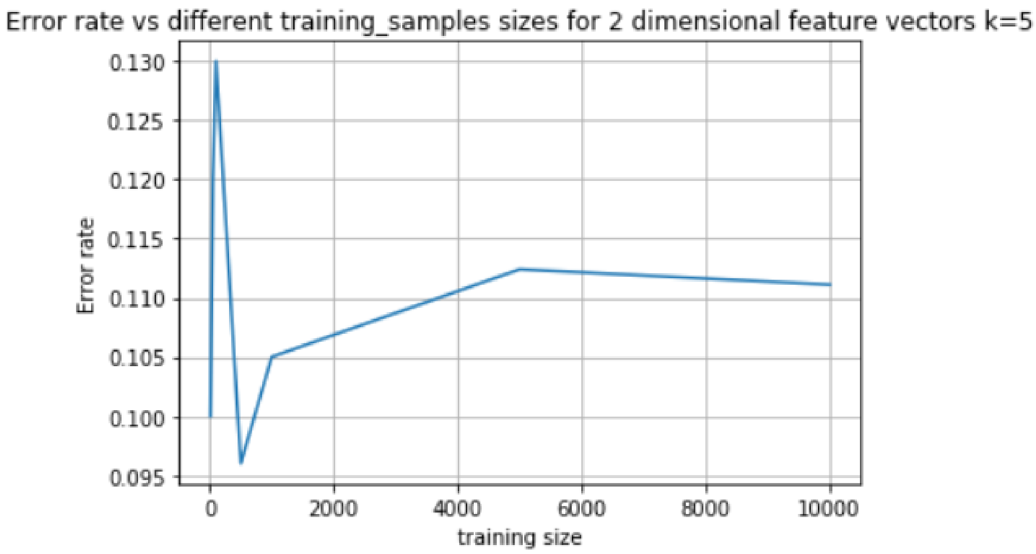
- The best $K = 3$.

Q2.4

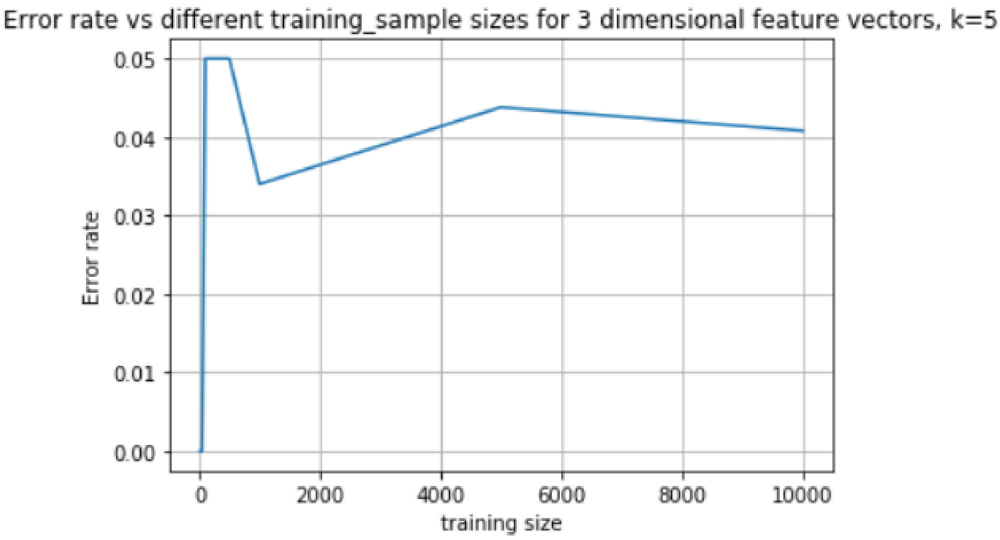
Analyze how # of training samples (e.g., from 10 to 10k) impact the error rate (@ testing data), with different dimension: N. You can choose other parameters based on your own need.

Now we set $K = 5$.

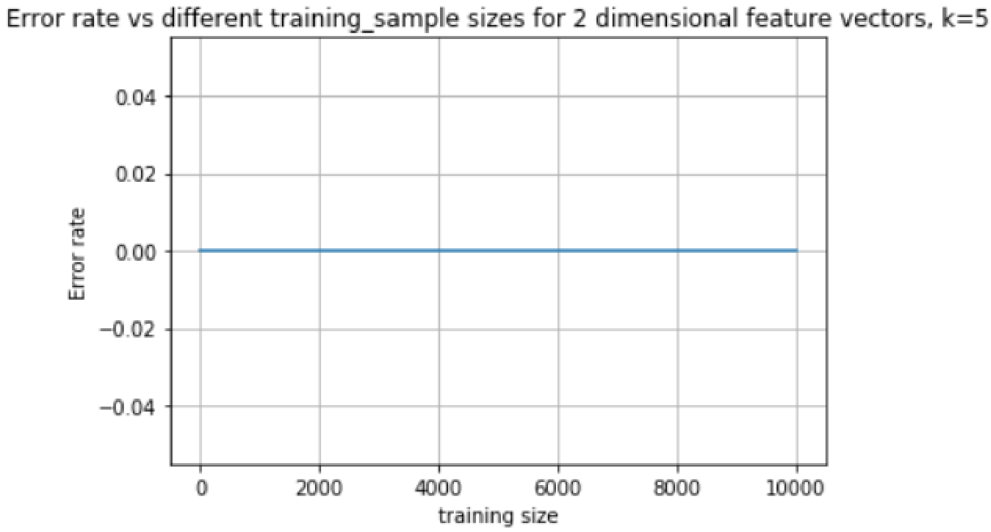
- 2 dimensional for 2 different classes



- 3 dimensional for 2 different classes



- 4 dimensional for 2 different classes



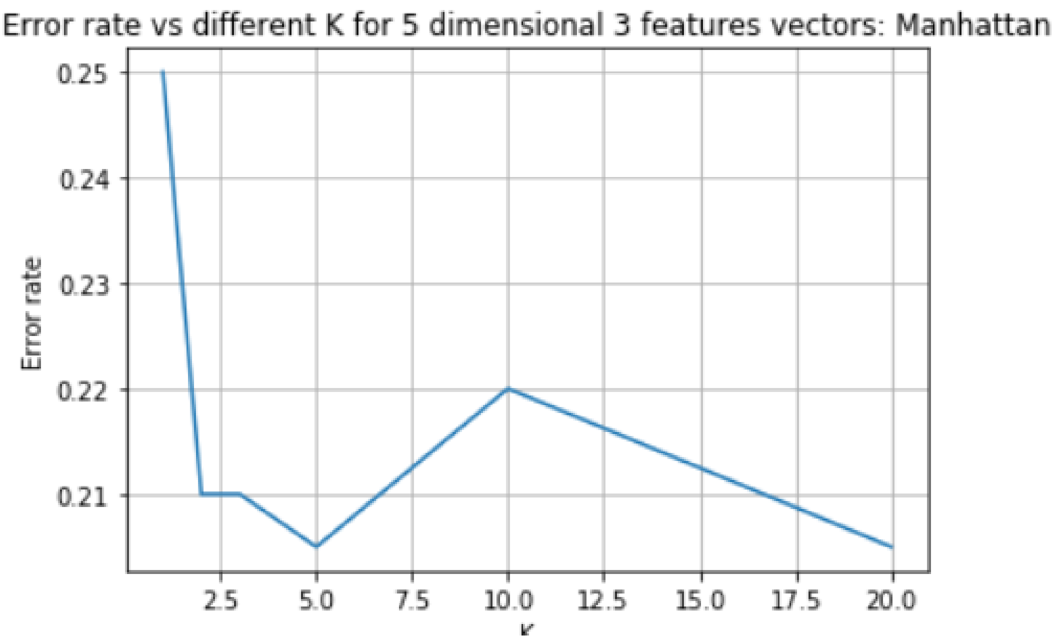
- Number of sampling does effect accuracy, but not that much like K's effect.

Q 2.5

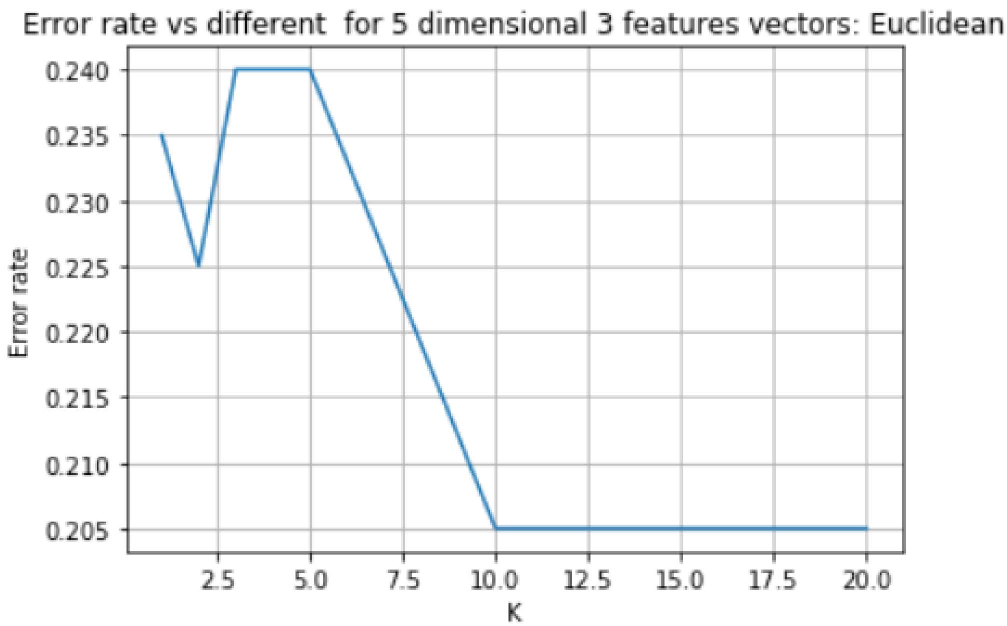
Study the difference of Euclidean distance and Manhattan distance.

- you can choose C=2, varying K. You can choose other parameters based on your own need.
- C = 3, N = 5, varying K

- Manhattan Distance:



- Euclidean Distance



- It just depends on the K and how many features we used.