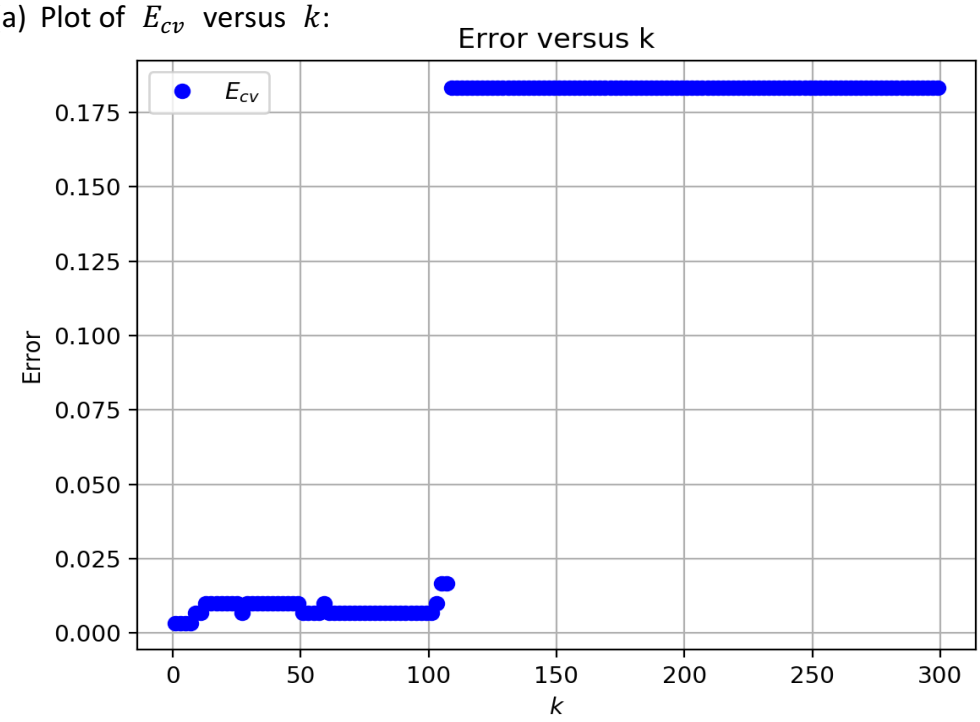
CSCI 4100 Assignment 11

Boliang Yang 661541863

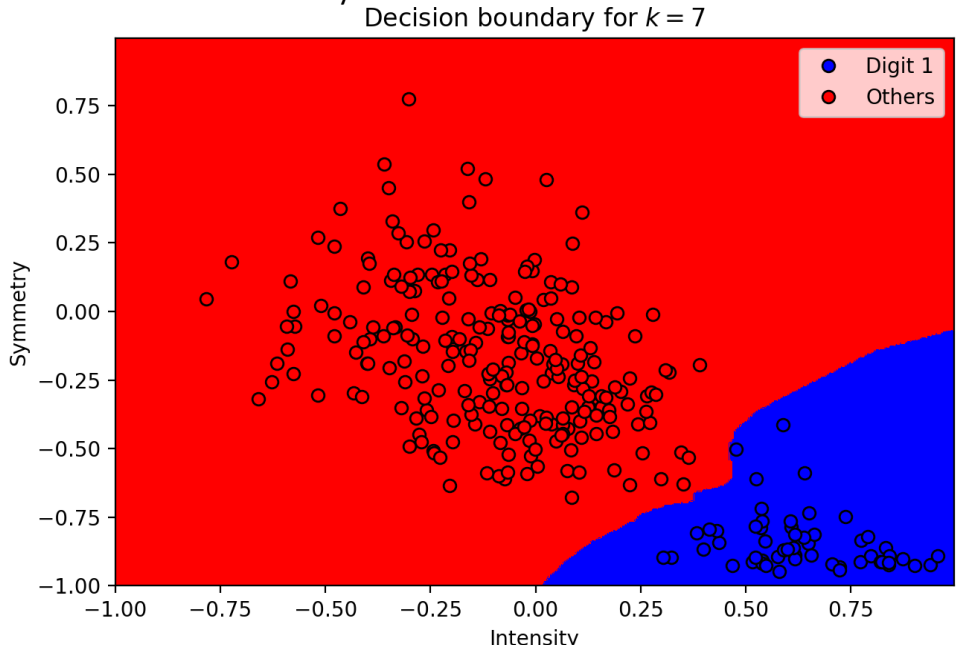
1. k-NN Rule

(a) Plot of E_{cv} versus k:



I choose $k = 7$ because when $k = 7$, $E_{cv} = 0.003335$ is smallest.

(b) Plot of the decision boundary:

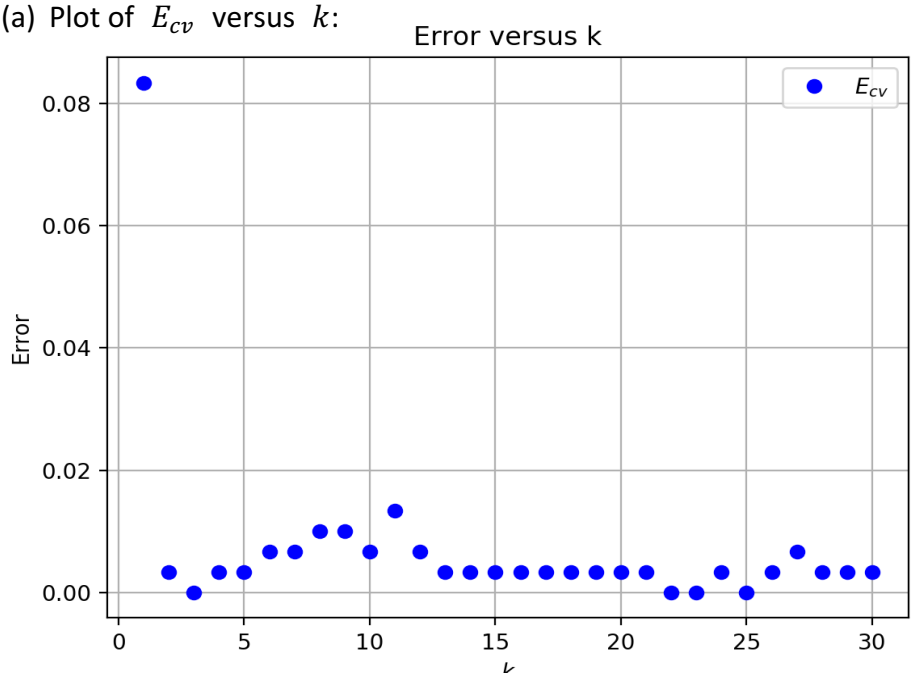


The in-sample error is $E_{in} = 0.0$, the cross validation error is $E_{cv} = 0.003335$.

(c) The test error $E_{test} = 0.009441$.

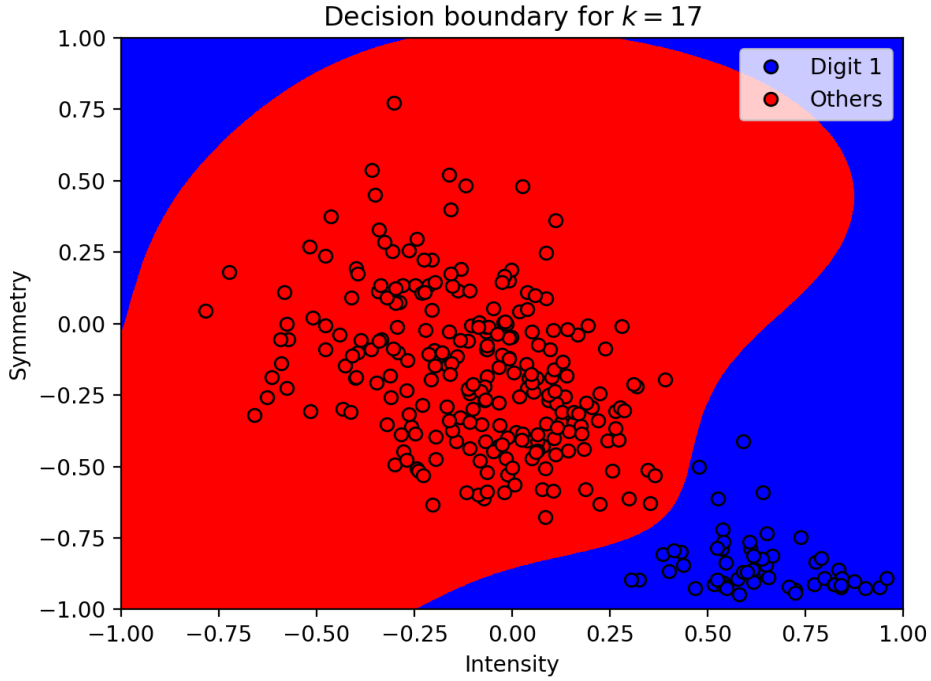
2. RBF-network

(a) Plot of E_{cv} versus k:



I choose $k = 17$ because when $k = 17$, $E_{cv} = 0.008869$ is smallest.

(b) Plot of the decision boundary:



The in-sample error is $E_{in} = 0.0$, the cross validation error is $E_{cv} = 0.008869$.

(c) The test error $E_{test} = 0.009726$.

3. Compare Linear, k-NN, RBF-network

The final test error from my three attempts are: for Linear model with 8th order polynomial transform, $E_{test} = 0.057199$; for k-NN rule, $E_{test} = 0.009441$; for RBF-network, $E_{test} = 0.009726$. We can see that k-NN rule gives us the smallest test error, followed by RBF-network, and then the Linear model.

However, k-NN is the slowest among these three algorithms and requires the largest memory space. Thus, k-NN is not a good choice.

The Linear model with 8th order polynomial transform is not a good choice as well. Not only it has the largest test error, but it also has a higher order polynomial transform which results in an increase in complexity.

For RBF-network, the running time is fast and it has a small test error without requiring large memory. Therefore, RBF is the best among these three.