1.1.

Given that x =

=

We know that K(x,z) = .

K(x,z) = .

=

K(x,z) =

K(x,z) =

1.2

1. If we map the input vector to the feature space and then do the dot product on the mapped features. We have

. = .

=

In the above equation we have 3 + 5 + 3 = 11 multiplications and 2 additions.

1. If we use the equation I got in question 1

K(x,z) =

If we consider m = then we have K(x,z) = m.m and in m we have 2 multiplications and 1 addition, so for K(x,z) we will have 3 multiplications and 1 addition, where 2 multiplications are needed for m and one multiplication needed for squaring m.

Actually both A and B refer to the same equation A is expansion of equation B so we actually should get the same no of multiplications and additions for both.

2.1.

K(x,x’) can be written as floor( which is 1 when x=x’ else 0. should be of the form [ 1 0 0……] it should be an m dimensional vector there only one element is 1 and rest all are zero, only then we can have . as 1 when x=x’ else 0. The closest function I can get for this scenario is = , since dimension of X is not given I am assuming it to be m-dimension.

2.2

The given kernel is a linear kernel since it is represented as K(x,z) = x.z and since this is a valid kernel we will have a linear separator which can classify the given data. We know that w.x can be written as w.x = if K(x,x’) is x.x’ then w.x will be a linear equation and hence we get a linear separator.

2.3

This will be a bad idea when m is very large or dimension of X is smaller than m. It will be computationally expensive and also will be hard to find a function which maps accordingly.

3.

Optimal Hyper Parameters are

Learning rate = 0.0001

No of epochs = 50

Regularization = 0.85

Training Accuracy = 92.531

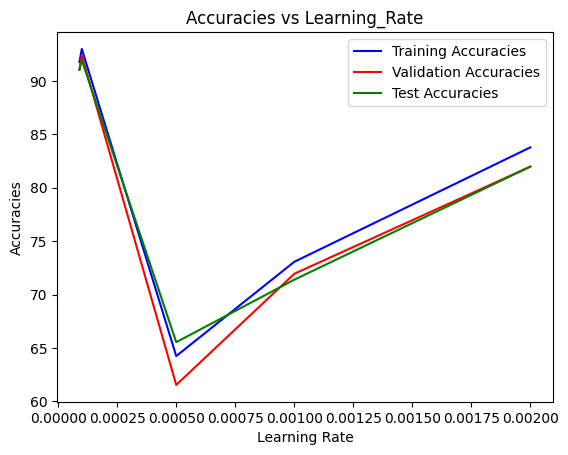
Validation Accuracy = 91.630

Testing Accuracy = 92.061

The code for the same is attached

The plots are as follows

Accuracies vs Learning rate



Accuracies vs Regularization Constant

