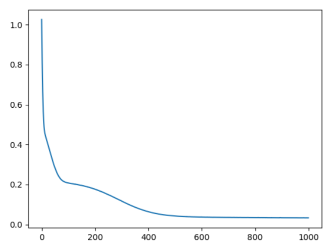
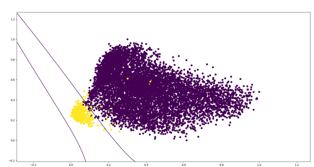
Homework 12

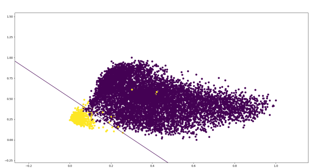
Karan Sarkar
sarkak2@rpi.edu

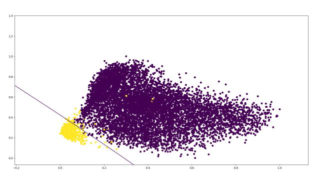
December 10, 2019

1. (a)
```
[array([[-0.03224736, -0.03224736],
       [-0.03224736, -0.03224736]], dtype=float32), array([-0.03224736, -0.03224736], dtype=float32), array([[-0.13732731],
       [-0.13732731]], dtype=float32), array([-0.21621275], dtype=float32)]
```
   (b)
```
[array([[-0.03222436, -0.03222436],
       [-0.03222436, -0.03222436]], dtype=float32), array([-0.03222436, -0.03222436], dtype=float32), array([[-0.13726544],
       [-0.13726544]], dtype=float32), array([-0.2160545], dtype=float32)]
```

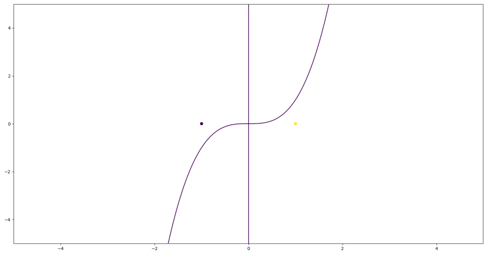
2. (a) 



   (b) 

   (c) 

3. (a) The optimal hyperplane is $x_1 = 0$.
   (b) i. The data points are $(1, 0)$ and $(-1, 0)$
   ii. The optimal hyperplane is $z_1 = 0$.

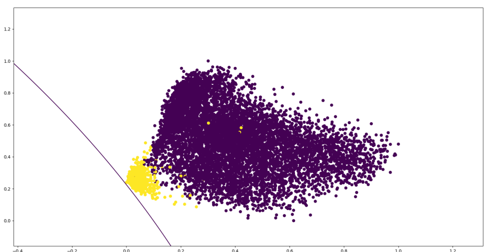


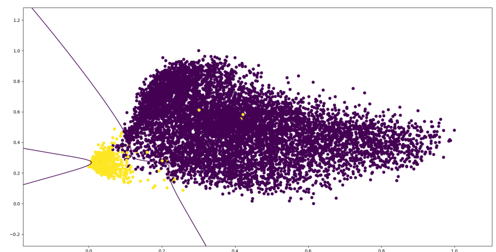
(c)

(d) $K(x, y) = (x_1^3 - x_2)(y_1^3 - y_2) + x_1 x_2 y_1 y_2$.

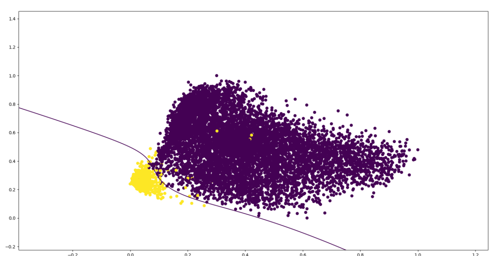
(e) The decision boundary is $x_1^3 - x_2 = 0$.

4. (a) Here is the decision boundaries for a small C of 0.01 and 1000000 respectively.





(b) The parameter C says how much margin the decision boundary should give data points. When C is small it gives a large boundary yielding a more simple less complex boundary. When C is large, it gives a small margin yielding a more curvy and complex function.



(c)

This uses C = 100.

5. The linear model performs the worst with a test error of 0.0299. The KNN performs exceptionally well at a test error of 0.00644. The RBF network performs similarly at 0.00722. The neural network also performs poorly at test error of 0.012. Finally, the SVM performs best at a test error of 0.005.

   The SVM is very successful. Unfortunately it is a cubic algorithm and thus may not scale well to larger datasets. The Neural Network's linear complexity may allow it to outperform the SVM for very large datasets.