

---

# Machine Learning Homework #2

---

Joe Ellis - jge2105

October 13, 2013

## 1 PROBLEM 1

## 2 PROBLEM 2

## 3 PROBLEM 3

For this problem we calculated the gram matrix for a given kernel  $f(x, \lambda, k)$ . We define  $f(x, \lambda, k)$  as 0 for  $x < 0$ , and  $f(x, k, \lambda) = (\frac{k}{\lambda})(\frac{x}{\lambda})^{k-1}(e^{-(\frac{x}{\lambda})^k})$ . We define our kernel function  $K(x_i, x_j) = F(x_i^t x_j, \lambda = 0.75, k = 3)$ , where  $F(x, \lambda, k) = \int_0^x f(x, \lambda, k) dx$ .

When we integrate  $(f(x, \lambda, k))$ , we get for the indefinite integral,

$$\int f(x, \lambda, k) dx = F(x, \lambda, k) = -e^{(\frac{x}{\lambda})^k}. \quad (1)$$

Therefore, using this knowledge and computing the definite integral between two points, we created a Matlab script to calculate the desired gram matrix. The gram matrix can be seen here,

$$\begin{pmatrix} 0.0698 & 0.0456 & 0.0104 & 0.0072 & 0.0635 \\ 0.0456 & 0.0420 & 0.0043 & 0.0061 & 0.0191 \\ 0.0104 & 0.0043 & 0.0053 & 0.0007 & 0.0080 \\ 0.0072 & 0.0061 & 0.0007 & 0.0009 & 0.0039 \\ 0.0635 & 0.0191 & 0.0080 & 0.0039 & 0.2478 \end{pmatrix}$$

This is the gram matrix for the points given in problem 3.

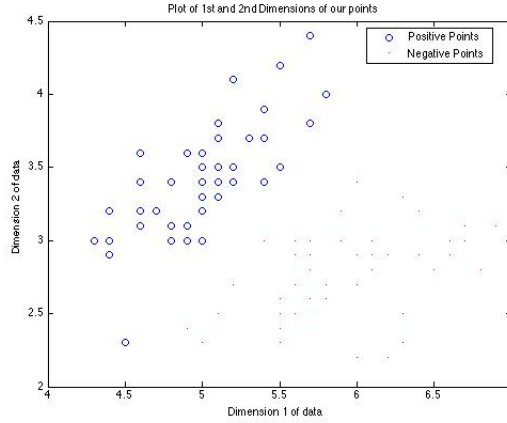


Figure 1: Plot of the 1st and 2nd dimensions of the dataset

## 4 PROBLEM 4

Here we create an object recognition system using SVM for classification. Multiple different SVM kernels were chosen and evaluated, and all of the kernels chosen were able to perform well with the proper parameters. We performed cross-validation across the data-set using a 50-50 train and test split for the sample data points. We performed cross-validation with 10 random splits for each parameter to show the performance of the SVM with each different kernel. Given the proper parameters we were able to get 100% accuracy using each of the described kernels, and this is because the data provided to us was seperable. Figure 4 shows the positive and negative samples from our dataset plotted with respect to their first two dimensions. We can see that they are seperable through a linear svm.

For classification we used linear, RBF, and polynomial SVM with different values for the cost variable ( $C$ ), and the number of degrees for the polynomial function and the size of sigma in the RBF. Each of these paramters effected the classification accruacy, but proper parameters for each kernel scheme could be found. The results of a different parameters can be seen in the upcoming figures.

As you can see, we have found parameters for each type of kernel that achieves maximum accuracy. However, the kernel that in general used the least amount of support vectors, and was quickest in classification was the linear SVM. Therefore, for this particular classification task I would reccomend using the linear SVM with  $C = 10$ . However, if the dataset is more difficult and not linearly seperable we would instead need to focus on using a more flexible kernel function such as the RBF.

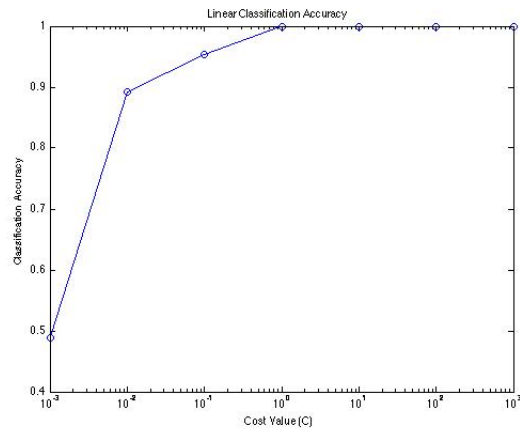


Figure 2: Linear SVM classification accuracy for different parameters

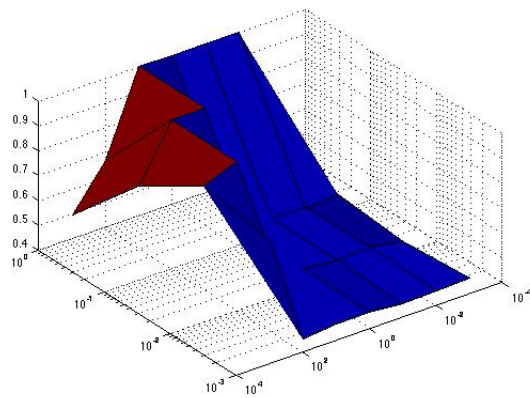


Figure 3: RBF SVM classification accuracy for different parameters

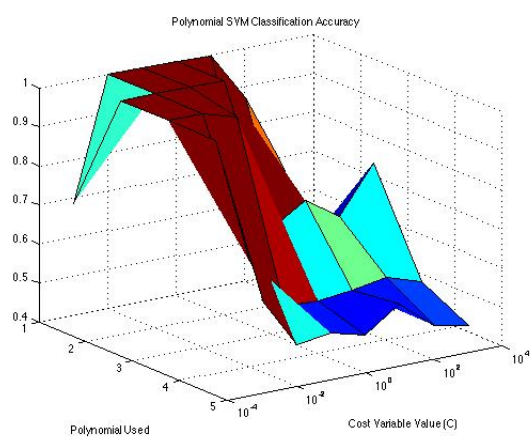


Figure 4: Polynomial SVM classification accuracy for different parameters