# 6.867 Machine Learning Homework 2 & 3

## 1. Support Vector Machine Classification

In this section of the homework we explore the use of support vector machines(SVM) for use in the task of classification. We will focus on implementing the SVM dual optimization problem. Additionally, we will implement the soft-margin SVM which contains a slack variable to allow for data points to be on the incorrect side of the decision boundary with some amount of tolerance. This soft margin allows us to classify points even in the situation when there is no linear separation between classes. This slack is parameterized by the variable C. As C increases the SVM approaches a hard-margin.The complete optimization problem that we seek to solve is shown below.

$$\text{maximize} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i-1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} \left( x^{(i)} \cdot x^{(j)} \right) \quad \text{subject to}$$

$$0 \leqslant \alpha_i \leqslant C$$

$$\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$$

We utilize the cvxopt quadratic programming toolbox for python in order to solve this convex optimization problem. Once we optimize for the values of $\alpha$ we must solve for $w_0$ and $w$ shown in the equations below. In the equation below M is the set of support vectors.

$$w = \sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i)} \ .$$

$$w_0 = \frac{1}{|\mathcal{M}|} \sum_{j \in m} \left( y^{(j)} - \sum_{i \in S} \alpha_i y^{(i)} \left( x^{(j)^T} x^{(i)} \right) \right) \ .$$

After computing $w_0$ and $w$ we have our decision boundary and apply the decision rule below to predict the label for subsequent data.

$$y(x) = \sum_{i \in S}^{n} \alpha_i y^{(i)} \left( x^T x^{(i)} \right) + w_0$$

### 1.1.

In this section we test our implementation of the soft-margin SVM using the synthetic data with positive examples (1, 2), (2, 2) and negative examples (0, 0), (-2, 3). Table 1 shows the alphas resulting from solving the quadratic optimization problem, the Weight and W_0 vectors as well as the accuracy of this classifier on the training set which is 100%. Figure one shows the resulting decision boundary and classification of points. We can clearly see that our implementation is successful in producing the maximum margin separator for the training data.

| Alphas | W | W_0 | Accuracy |
|---|---|---|---|
| 5.30612273e-01 | 0.85714299 | -1.21428584 | 100% |
| 3.11022745e-08 | 0.57142862 | | |
| 3.67346975e-01 | | | |
| 1.63265328e-01 | | | |

**Table 1: Optimal Alpha values, Weight vector, W_0 vector and accuracy for basic dataset**

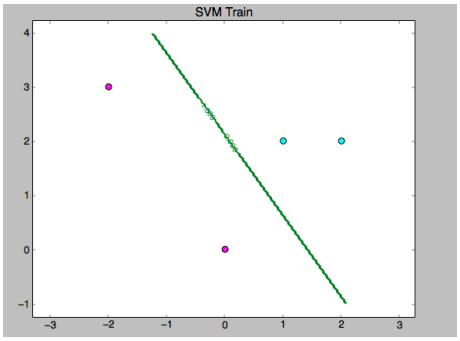


**Figure 1: SVM decision boundary using synthetic data points**

Notice here that none of our *alpha* values are zero, yet the data should have produced only three support vectors not four. The model creates a support vector of the point furthest to the right, which is should not; this point should not affect the decision boundary. The value of alpha for this point is very low at 3.11022745e-08. We decided to threshold all alpha values at 1e-5; anything below this value is set to zero.

### 1.2.

C = 1

### 1.3. Problem 1.3

C vs. geometric margin and alphas Beta and some RBF pictures Try and show the margin on the linear classifier

## 2. Logistic Regression

## 3. Multi-Class Classification

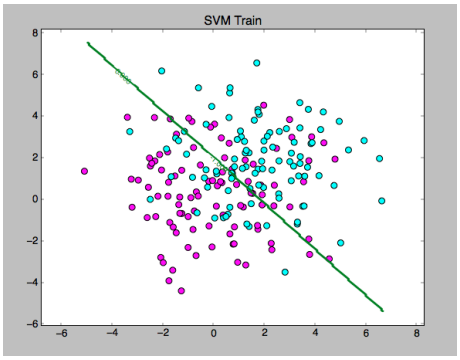


Figure 2: BigOverlap Data Training

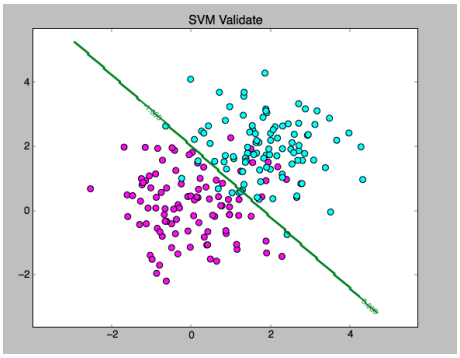


Figure 3: BigOverlap Data Testing

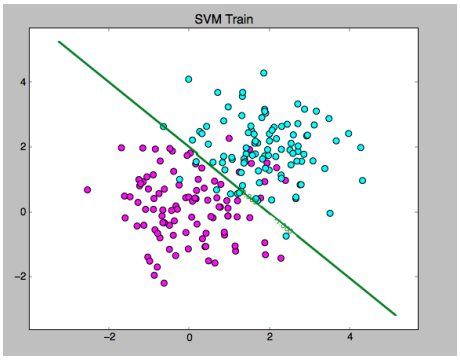


Figure 4: SmallOverlap Data Training

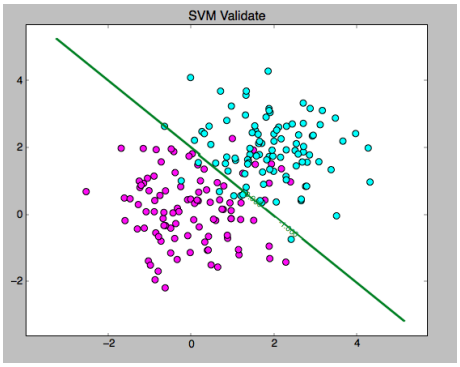


Figure 5: SmallOverlap Data Testing

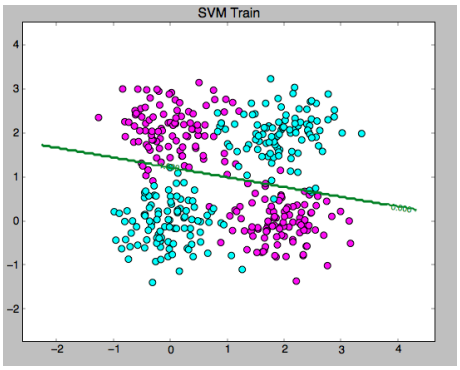


Figure 6: Nonsep Data Training

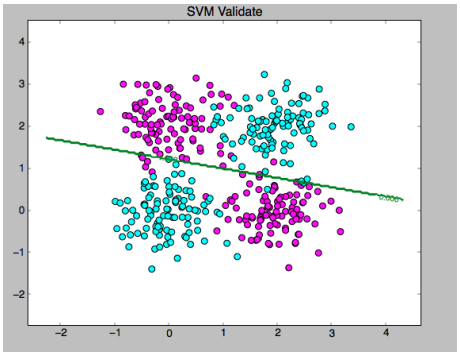


Figure 7: Nonsep Data Testing

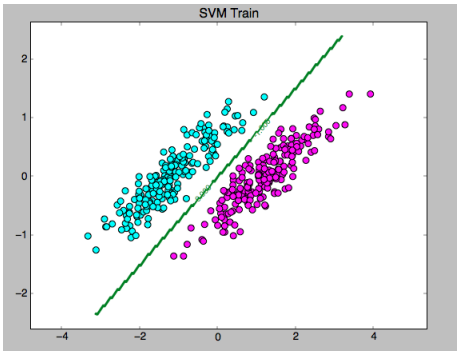


Figure 8: LS Data Training

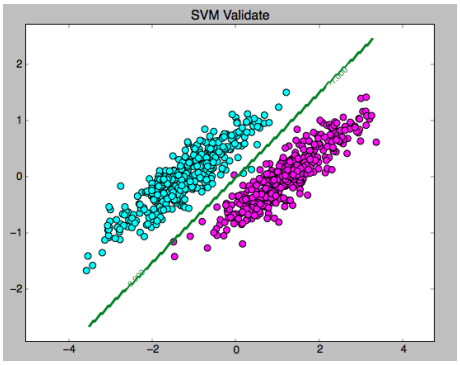


Figure 9: LS Data Testing

| Data | LS | Nonsep | smallOverlap | bigOverlap |
|---|---|---|---|---|
| **# Support Vectors** | 13 | 394 | 50 | 129% |
| **W** | -1.94, 2.58 | -0.11, -0.51 | 1.12, 1.20 | 0.39, 0.35 |
| **W_0** | 0.02 | 0.61 | -2.36 | -0.70 |
| **Error Train** | 0.00% | 0.503% | 0.09% | 0.275% |
| **Error Test** | 0.0037% | 0.503% | 0.09% | 0.085% |

**Table 2: Number of Support Vectors, Weight vector, W_0 vector and accuracy for various datasets**