Robotics 811 - Homework 4 - Resubmit

Xiang Zhi Tan

November 24, 2015

1 Q7

7(a)

We can first rearrange the constraints 2 and 3 into the following form by moving elements on the right handside to the left and multiplying constraints 2 by -1.

$$\begin{gathered} -w^T x_i - b + 1 - \xi_i \le 0 \\ \text{if } y_i = 1 w^T x_i + b + 1 - \xi_i \le 0 \\ \text{if } y_i = -1 \end{gathered}$$

Through observing the inequalities, we notice that the only difference is the element $w^T x_i$ which has a different sign that could be change by the values of y_i. This allows us to combine both inequalities into the following constraint.

$$-y_i(w^T x_i + b) + 1 - \xi_i \le= 0$$

7(b)

The Lagrangian for our optimization problem is as following:

$$L([w\ 0\ \xi]^T, \alpha, \beta) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\alpha_i g_i([w\ b\ \xi]^T) - \sum_{i=1}^{l}\beta_i\xi_i$$

7(c)

Following is the steps to minimize the Lagrangian which is ther primal form of the SVM. Here's the original Lagrangian.

$$L([w\ 0\ \xi]^T, \alpha, \beta) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\alpha_i(-y_i(w^T x_i + b) + 1 - \xi_i) - \sum_{i=1}^{l}\beta_i\xi_i \tag{1}$$

First, we find the partial of w,b and ξ_i. Following are the partials follow by setting them to 0.
Finding partial of w

$$\begin{gathered} \frac{\partial}{\partial w} = \frac{2}{2}w - \sum_{i=1}^{l}(\alpha_i y_i x_i) = 0 \\ w = \sum_{i=1}^{l}(\alpha_i y_i x_i) \end{gathered} \tag{2}$$

Finding the partial of b.

$$\begin{gathered} \frac{\partial}{\partial b} = -\sum_{i=1}^{l}(\alpha_i y_i) = 0 \\ \sum_{i=1}^{l}(\alpha_i y_i) = 0 \end{gathered} \tag{3}$$

Finding the partial of xi_i.

$$\begin{gathered}\frac{\partial}{\partial \xi_i} = C - \sum_{i=1}^{l}(\alpha_i) - \sum_{i=1}^{l}(\beta_i) = 0 \\ C = \sum_{i=1}^{l}(\alpha_i) + \sum_{i=1}^{l}(\beta_i)\end{gathered} \tag{4}$$

Now we insert equations 2, 4 in equation 1.

$$\begin{aligned}L &= \frac{1}{2}(\sum_{i=1}^{l}(\alpha_i y_i x_i))^2 + (\sum_{i=1}^{l}(\alpha_i) + \sum_{i=1}^{l}(\beta_i))\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\alpha_i(-y_i((\sum_{j=1}^{l}(\alpha_j y_j x_j))x_i + b) + 1 - \xi_i) - \sum_{i=1}^{l}\beta_i\xi_i \\ L &= \frac{1}{2}(\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i\alpha_j y_i y_j x_j^T x_i)) + \sum_{i=1}^{l}(\alpha_i\xi_i) + \sum_{i=1}^{l}(\beta_i\xi_i) + \sum_{i=1}^{l}(\alpha_i\alpha_j y_i y_j x_j^T x_i)) \\ &- b\sum_{i=1}^{l}(\alpha_i y_i) + \sum_{i=1}^{l}(\alpha_i) - \sum_{i=1}^{l}(\alpha_i\xi_i) - \sum_{i=1}^{l}\beta_i\xi_i \\ L &= \sum_{i=1}^{l}(\alpha_i) - \frac{1}{2}(\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i\alpha_j y_i y_j x_j^T x_i)) - b\sum_{i=1}^{l}(\alpha_i y_i)\end{aligned} \tag{5}$$

By applying equation 3 in the previous equation, we can derive the final form

$$L^*(\alpha) = \sum_{i=1}^{l}(\alpha_i) - \frac{1}{2}(\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i\alpha_j y_i y_j x_j^T x_i)) \tag{6}$$

7(d)

As we want to write the equation in terms of H and f such that $L^*(\alpha) = \frac{1}{2}\alpha^T H\alpha + f^T\alpha$. By looking at equation 6, we see a similar structure between them. For the first part $\frac{1}{2}\alpha^T H\alpha$

$$\begin{gathered}\frac{1}{2}\alpha^T H\alpha = -\frac{1}{2}(\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i\alpha_j y_i y_j x_j^T x_i)) \\ H = -(\sum_{i=1}^{l}\sum_{j=1}^{l}(y_i y_j x_j^T x_i))\end{gathered}$$

For second part $f^T\alpha$

$$\begin{gathered}f^T\alpha = \sum_{i=1}^{l}(\alpha_i) \\ f^T = [1, 1, ...1]\end{gathered}$$

7(e)(i)

We implemented the SVM in the matlab file **svmSolver.m** and the solver can be tested and visualized by the files **q7e_1.m** for the first part and **q7e_2.m** for the second part of the question. Following are graphs of using the SVM solver on the test data and the decision boundary and margins for C=1,0.1,0.01.

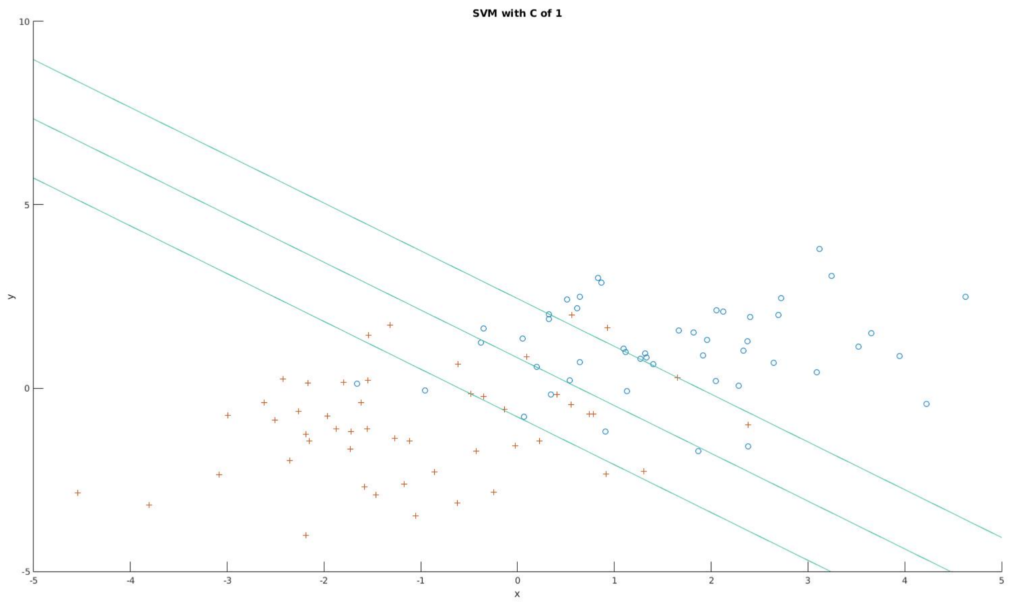

Figure 1: Plot of decision boundary and margins with C=1

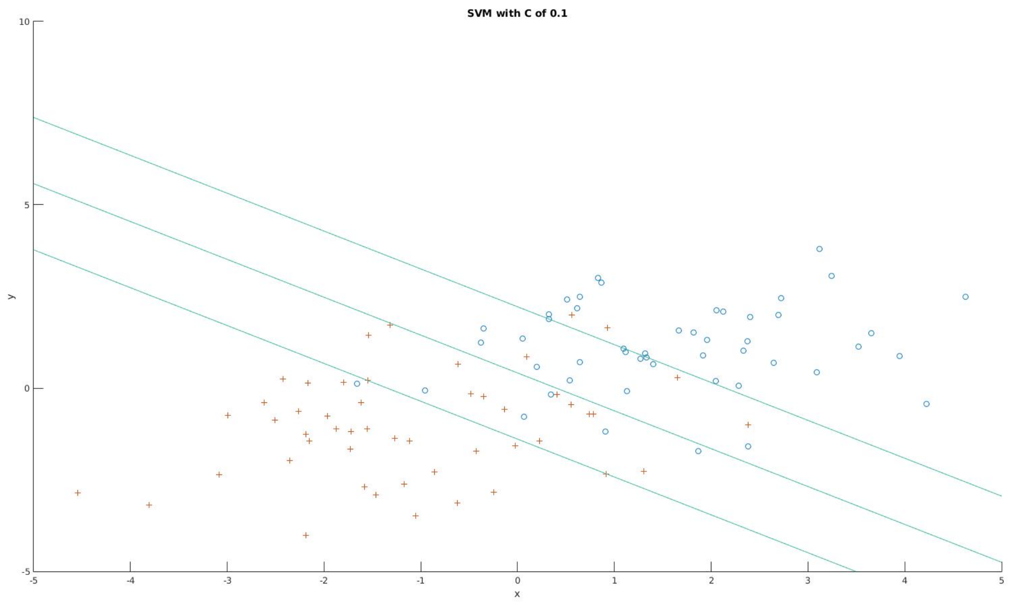

Figure 2: Plot of decision boundary and margins with C=0.1

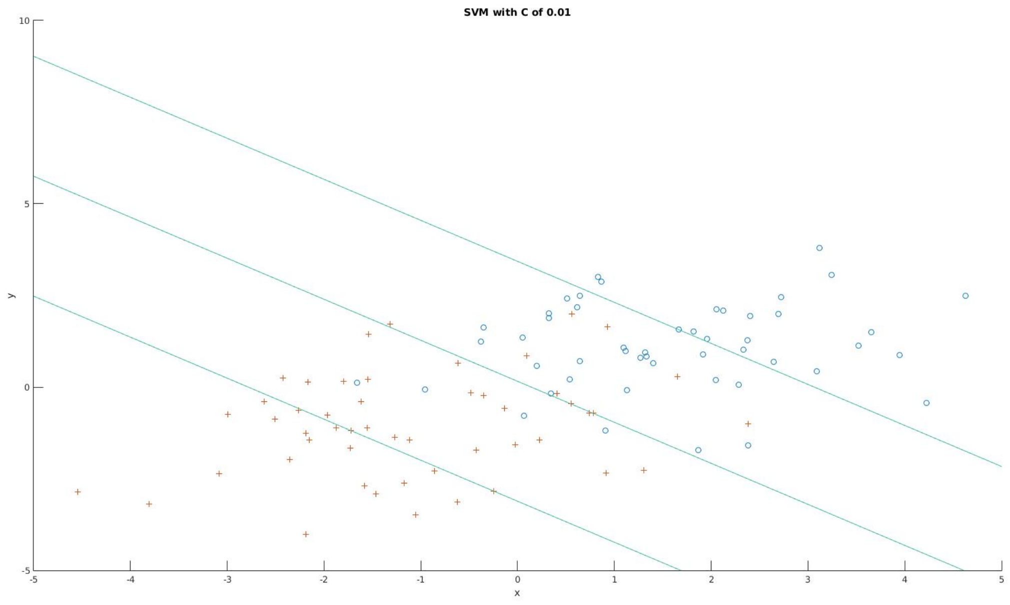

Figure 3: Plot of decision boundary and margins with C=0.01

As C increases, we observe that the locations of the support vector(points that lies on the margin lines) and margin lines gets further away from the decision boundary. This is the expected behavior because the larger the C parameter the more accurate the classification of the training data would be as it tries to create a decision boundary that minimize the training data error but might cause over fitting. Where a smaller C means creating a larger margin that would generalized more to other test sets.

7(e)(ii)

As observed in the previous few figures, we see that the decision margins increase as C decreases. Therefore, as C approaches 0, the sampling errors are negated.

$$\lim_{C\to 0} min\frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\xi_i = min\frac{1}{2}||w||^2 \tag{7}$$

Therefore, SVM will now ignore the sampling error and might choose points that have significant error as support vectors and give the wrong result

7(e)(iii)

This part of the question is implemented in the script,**q7e_2.m**. Following is the table showing the accuracy of my algorithm on a 10-fold cross validation run.

k	correct P	wrongly label	P detection %	correct N	wrongly label	N detection %	total accuracy
1	122	1	0.968254	126	4	0.992126	0.980237
2	131	0	0.970370	119	4	1.000000	0.984252
3	122	0	0.897059	117	14	1.000000	0.944664
4	141	0	0.972414	109	4	1.000000	0.984252
5	141	0	0.979167	109	3	1.000000	0.988142
6	129	0	0.941606	117	8	1.000000	0.968504
7	124	0	0.892086	114	15	1.000000	0.940711
8	116	0	0.983051	135	2	1.000000	0.992095
9	138	0	0.992806	114	1	1.000000	0.996047
10	133	0	0.970803	117	4	1.000000	0.984252

The first wrongly label column is the number of N that are labeled as P by the SVM, the second wrongly label column is the number of P that is wrongly label by N by the SVM
The average accuracy over the 10 fold validation is 0.9763 which is pretty good.

7(e)(iv)

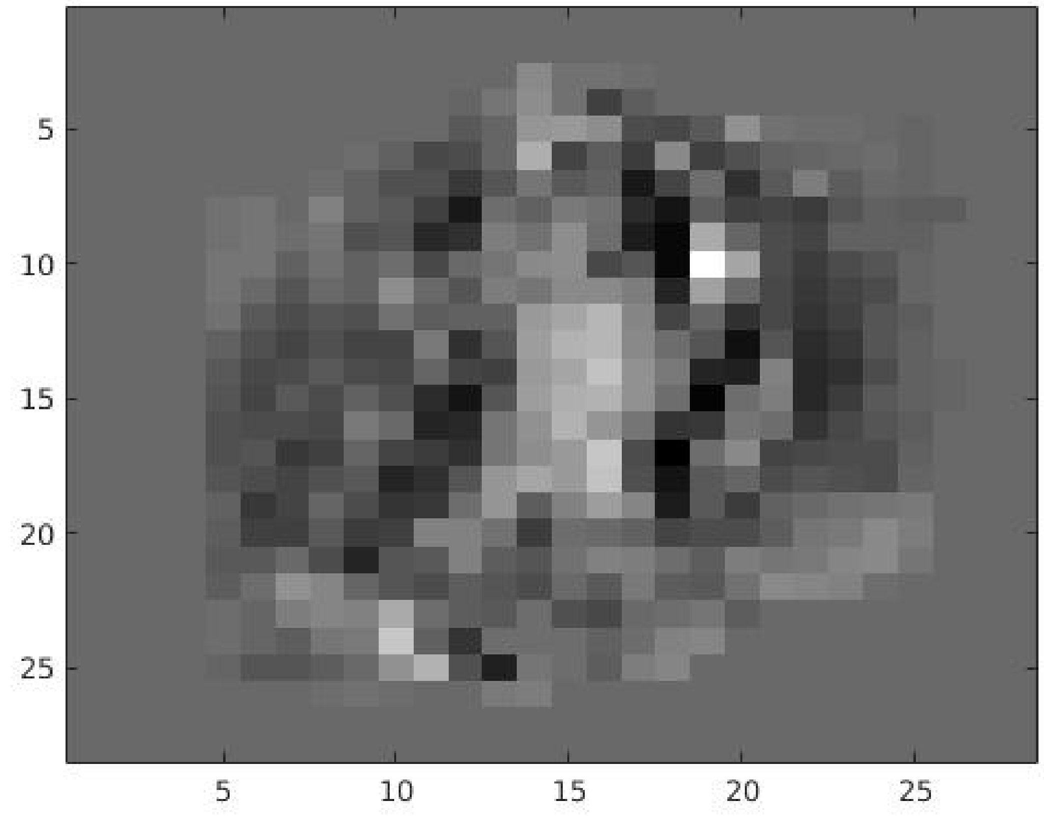

Figure 4: plot of weight function

The weight functions shows us what the algorithm learns. It assigns mainly positive weights in the center and at the edge in the middle. This shows that for detecting ones, it's important to have pixel values at the corners and the centers which is how a 1 is written. Where a negative weight resemblances a circle around which is the shape of the zeros which is what we are detecting.