Solution Sketch Homework 4

May 28, 2018

Vasileios Athanasiou
vasath@chalmers.se

1 Theoretical Problems

1.1 SVM

The decision boundary should pass between the two classes of points, and the distance to the closest point on each side should be the same. We see that this line should pass through e.g. (1; 2) and (3; 0). The weights and bias fall out as

$$w = (1, 1)^T \tag{1}$$

$$b = -3 \tag{2}$$

But $w = (-1, -1)^T$ and $b = 3$ is equivalent. The distance γ is the distance from the hyperplane to the closest point. The margin is $2\ \gamma$.

1.2 SVM cont'd

1.2.a Primal formulation

The primal formulation is:

$$\min_w \frac{1}{2}\ w^T\ w\ ,\ w \in R^d \tag{3}$$

$$constraints : y_n\ (w^T x_n + b) - 1 \geq 0 \tag{4}$$

For the specific example:

$$\min_w \frac{1}{2}\ w^T\ w\ ,\ w \in R^2 \tag{5}$$

$$y_n\ (w_1\ x_1 + w_2\ x_2 + b) - 1 \geq 0 \tag{6}$$

And,

$$(w_1\ 2 + w_2\ 2 + b) - 1 \geq 0 \tag{7}$$

$$(w_1\ 4 + w_2\ 4 + b) - 1 \geq 0 \tag{8}$$

$$(w_1\ 4 + b) - 1 \geq 0 \tag{9}$$

$$-b - 1 \geq 0 \tag{10}$$

$$-(w_1\ 2 + b) - 1 \geq 0 \tag{11}$$

$$-(w_2\ 2 + b) - 1 \geq 0 \tag{12}$$

The optimal solution of the primal formulation can be found analytically by finding the minimum absolute w parameters which satisfy the constraints.

Matlab's quadprog function can also be used to solve the problem. To do that, b is included by optimising over the three-dimensional vector $x = (w1, w2, b)$ instead of just w.

From help quadprog (but using c instead of b to avoid confusion):

min $0.5 * x' * H * x + f' * x$ subject to: $A * x \leq c$

The input matrices H and A and vectors f and c are constructed by adapting our problem to this form. Notably, H has no entries in row or column 3, and f is all-zero. The optimal solution is the same as the one found by inspection.

1.2.b SVM cont'd

The dual formulation of the SVM is given as follows. Find the parameters $\alpha_i, i = 1 \cdots N$, (where N is the number of the data points) for which the following expression is maximised,

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{n,m=1}^{N} a_n \ a_m \ y_n \ y_m \ x_n^T \ x_m \tag{13}$$

and it has to be true that:

$$\sum_{i=1}^{N} \alpha_i \ y_i = 0, and, \forall \ a_i \geq 0 \tag{14}$$

The particulars of the problem appears in the data points x_i, and we can re-write the sum they appear in on the form $\alpha^T H \alpha$, where H is a matrix found from the data points and their labels t_i. The problem can be formulated on the standard form and fed into quadprog, which yields a solution where α_2 and α_4 are both 0. This means these two values are not support vectors, but the other four (i.e. the ones with non-zero α values) are.

Despite being listed in the theoretical section of the homework, this task required the use of Matlab (or possibly some other tool for solving quadratic programming problems).

2 Practical problems

This solution sketch does not include a solution for the practical coding part. However, I do include some figures with the expected results. Since several things can be done differently, for example using "autoscaling" or not, using "svmtrain" or "fitcsvm", choice of C, many groups may get results other than these which can be considered correct.

2.1 SVM

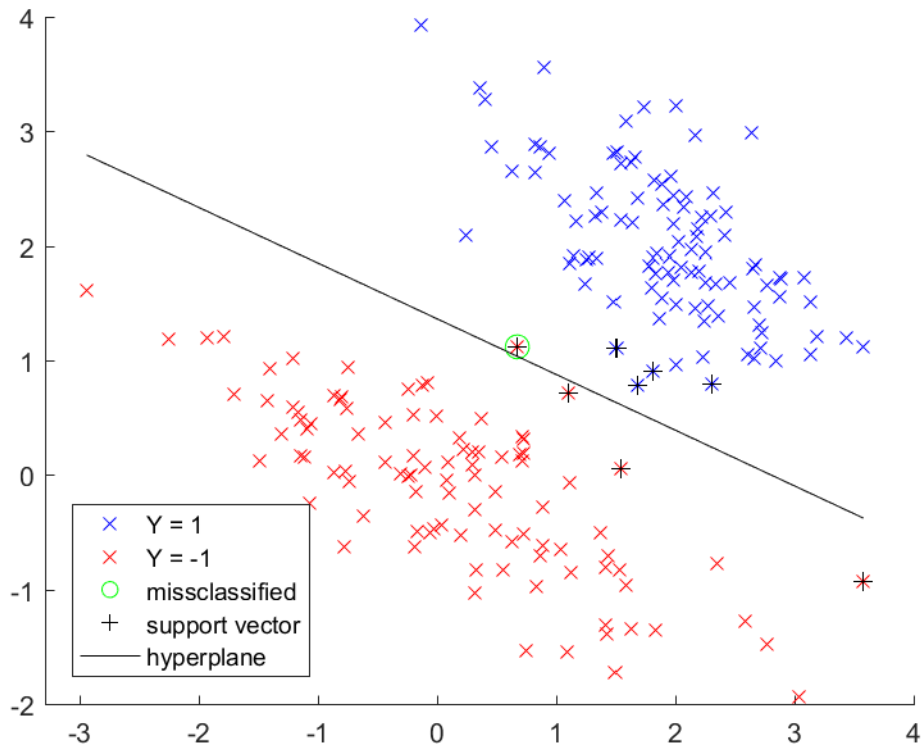**2.1.a**

One can choose $C = 1$.

2.1.b

Figure 1: The data of the two different classes are denoted as red and blue respectively. When the data points have been identified as support vectors, a plus sign has been added. When a data point is missclassified, it is rounded by a green circle.

2.1.c

The classifier has a bias of -2.4528. If the hyperplane is writen as a function in the form $y = a * x + b$, then, the bias is the value of b.

2.1.d

The soft margin can be calculated as $2\gamma = 0.9986$. It is calculated as $2\gamma = \frac{2}{\sqrt{w_1^2 + w_2^2}} = 0.9986$
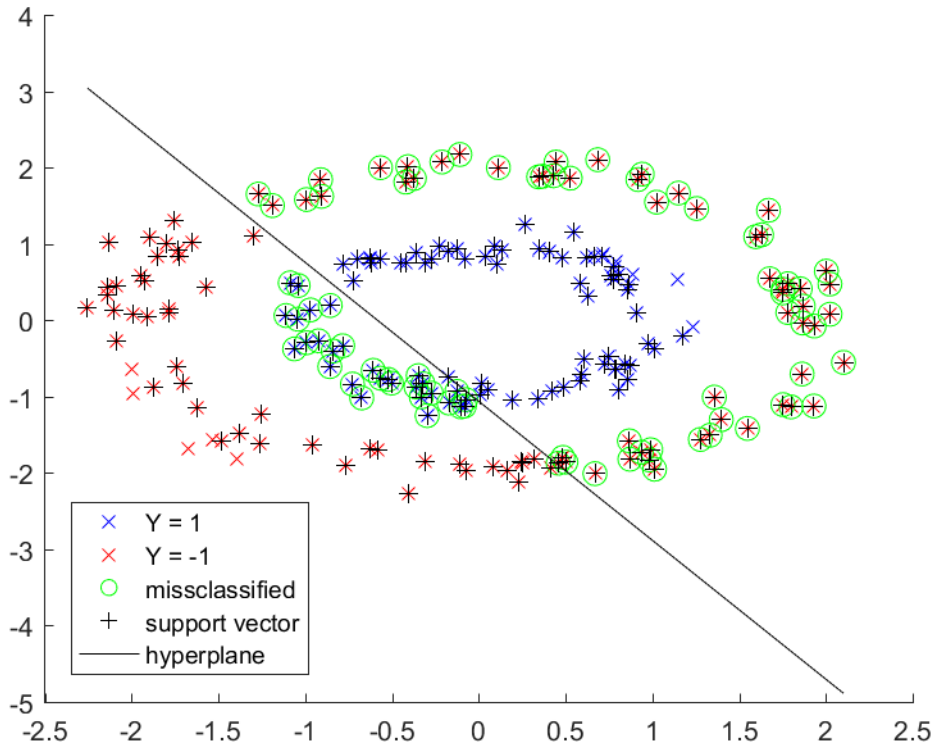
2.2 Kernels

2.2.a

One can set $C = 1$.

Figure 2: The data of the two different classes are denoted as red and blue respectively. When the data points have been identified as support vectors, a plus sign has been added. When a data point is missclassified, it is rounded by a green circle.

2.2.b

After training the machine with data, it is possible to extract the coefficients α_i and the *bias*. Therefore, it is possible to build the decision function f for a new data point x_{new} as:

$$f(x_{new}) = bias + \sum_i \alpha_i \, kernel(sv_i, x_{new}) \tag{15}$$

The plot of the hyperplane can be painted where the decision function f is smaller than a positive number ϵ, $-\epsilon \le f(x_{new}) \le \epsilon$.
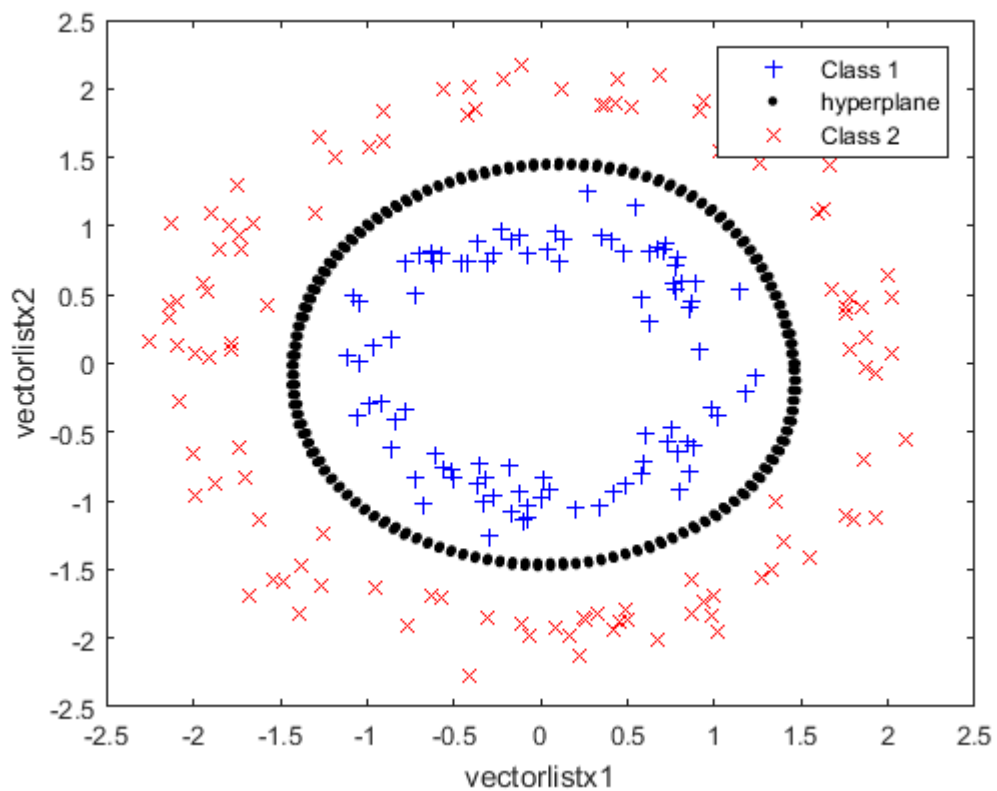
4

Figure 3: The hyperplane is denoted with black color and the dataset between corresponding to the two classes with red and blue color. Similar plots can be obtained by using the quadratic or the rbf kernel.