Martin Lundfall, Henri Bunting, Malte Siemers, Patrik Bey

Exercise sheet 09 - Machine Intelligence I

9.1

The primal problem for C-SVMs is solved by using the Karush-Kuhn-Tucker conditions, a generalization of Lagrange multipliers. We aim to minimize the following equation:

$$L_{(w,b,\{\lambda_\alpha\})} = f_0(w,b) + \sum_{\alpha=1}^{p} \lambda_\alpha f_\alpha(x) \tag{1}$$

Where f_0 is the function to be minimized, in our case:

$$f(w,b) = \frac{||w||^2}{2} + \frac{C}{p} \sum_{\alpha=1}^{p} \varphi_\alpha \tag{2}$$

and f_k are the inequality constraints, in our case:

$$f_\alpha(w,b) = (1 - \varphi_\alpha) - y_T(w^T x + b) \tag{3}$$

$$f_\alpha(w,b) = -\varphi_\alpha \tag{4}$$

We minimize the KKT equation first with respect to w. Note that the slack variables φ_α are independent of w. This results in:

$$w = \sum_{\alpha=1}^{p} \lambda_\alpha y_T^{(\alpha)} x^{(\alpha)} \tag{5}$$

Then minimizing with respect to b we get the following constraint on λ_α:

$$0 = \sum_{\alpha=1}^{p} \lambda_\alpha y_T^{(\alpha)} \tag{6}$$

Putting these results back into the KKT equation, we attain:

$$L_{(x_k,\{\lambda_k\})} = \frac{1}{2} \sum_{\alpha,\beta=1}^{p} \lambda_\alpha \lambda_\beta y_T^{(\alpha)} y_T^{(\beta)} (x^{(\alpha)})^T x^{(\beta)} + \sum_{\alpha=1}^{p} \lambda_\alpha \tag{7}$$

We clearly see that minimizing this equation corresponds to maximizing the expression given in the exercise. The upper bound of the λ_α variables is given by the duality of the problem.

9.3 - C-SVM with standard parameters

a)

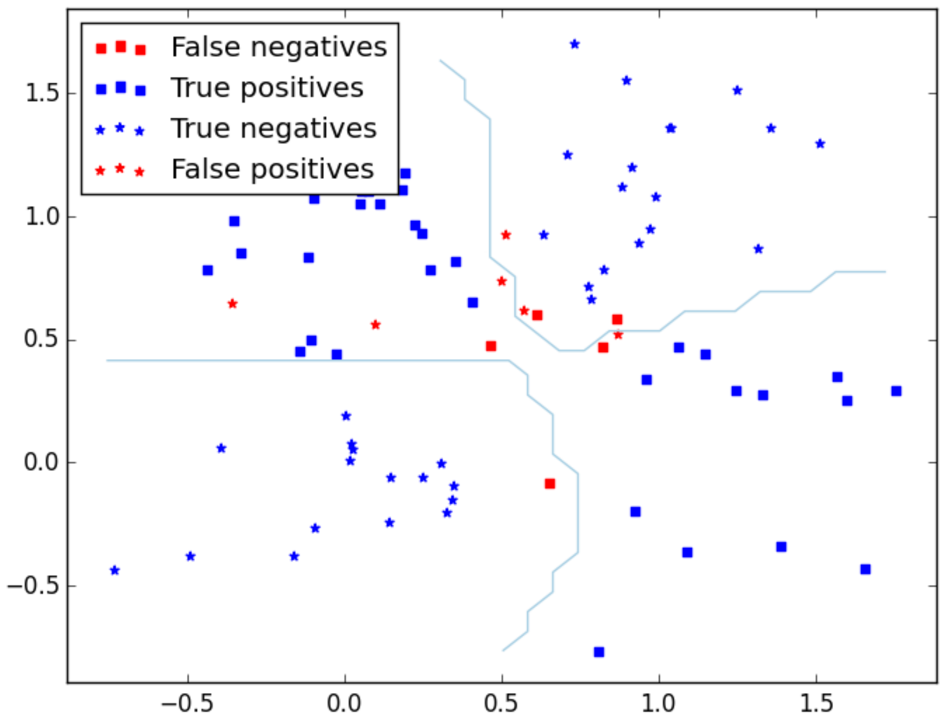
We used the scikit.learn python package to create and train a C-SVM as can be seen in the file 9.3.py.

b)

The mean error from the 0-1 loss function was 16.25%

c)

The plot of our classification with decision boundaries and false positives and false negatives highlighted:



9.4 - Parameter Optimization

(a)

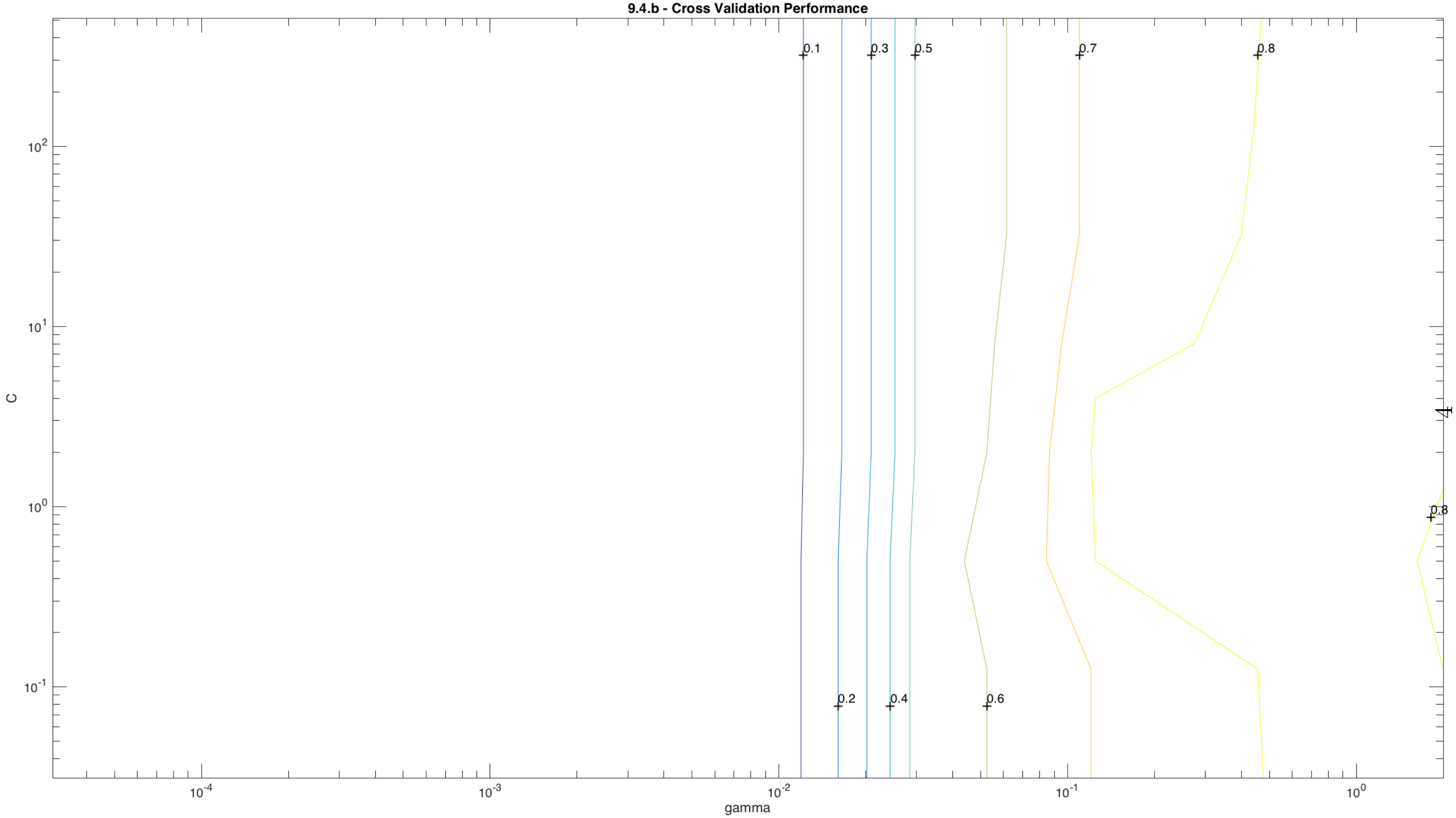
Following the procedure for grid search as described in the guide we performed leave-one-out cross validation (LOOCV) on the training data using

the following parameter range:
$\gamma = [2, 2^{-1}, 2^{-3}, 2^{-5}, 2^{-7}, 2^{-9}, 2^{-13}, 2^{-15}]$, $C = [2^{-5}, 2^{-3}, 2^{-1}, 2, 2^{3}, 2^{5}, 2^{7}, 2^{9}]$

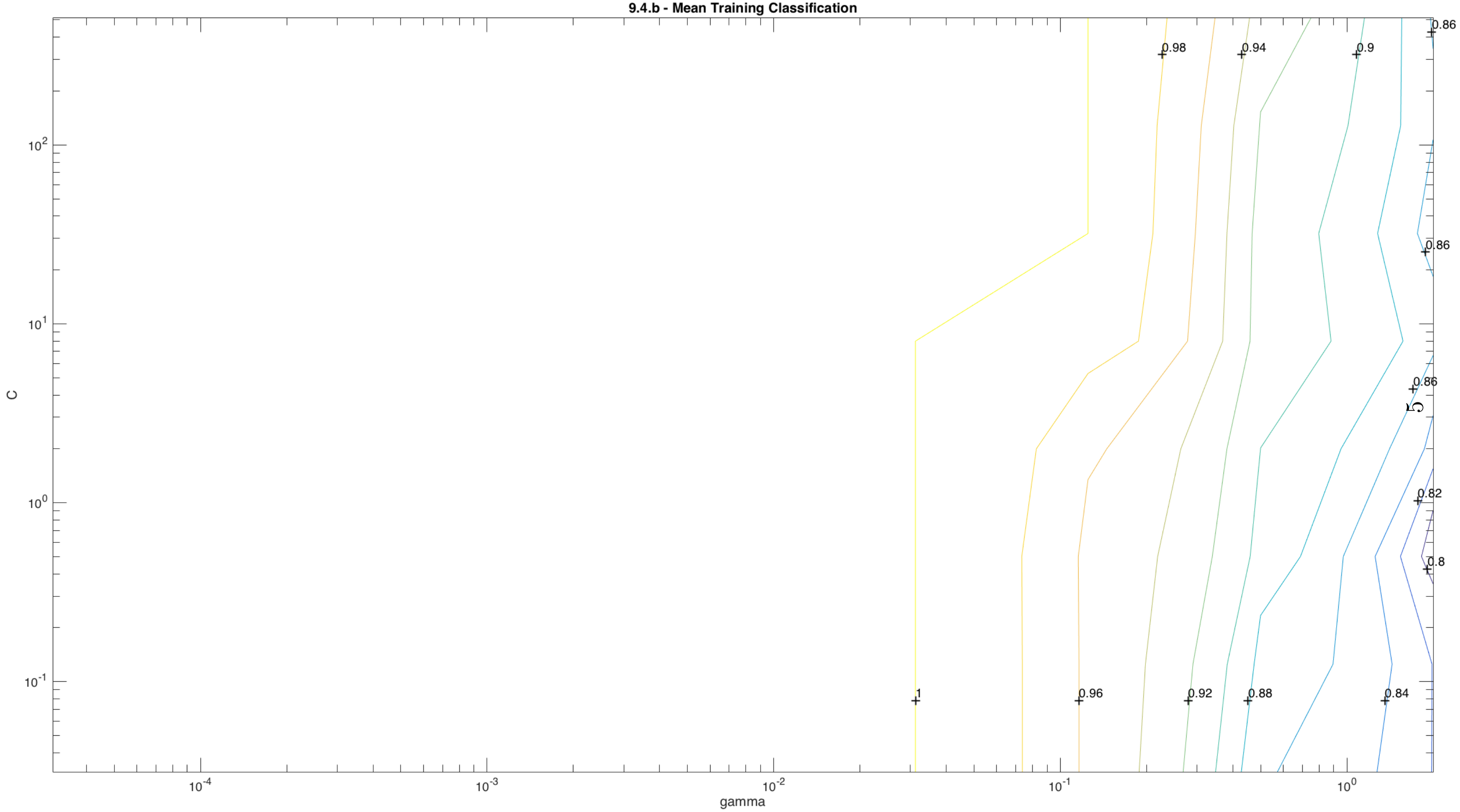
(c)

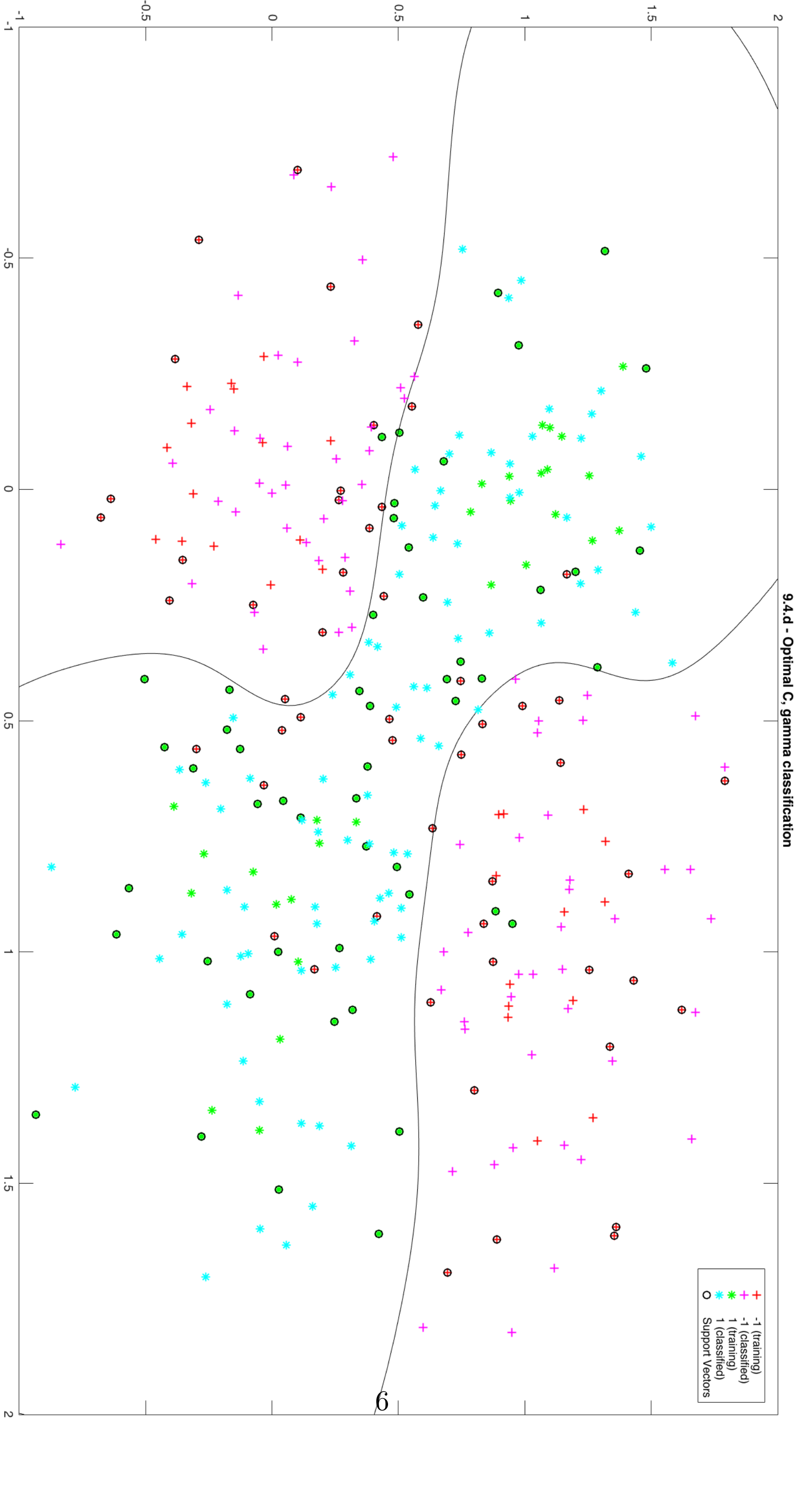
(e)

9.4.b - Cross Validation Performance



9.4.b - Mean Training Classification




9.4.d - Optimal C, gamma classification
-1 (training)
-1 (classified)
1 (training)
1 (classified)
Support Vectors
-1
-0.5
0
0.5
1
1.5
2