1 Computational Aspects of SVM

We have a dataset containing M points in N dimensions (i.e. each datapoint is described by N floats). After investigating the data and finding optimal hyperparameters, we train a Support Vector Machine (SVM) classifier with an RBF Kernel that results in S support vectors.

A) How many parameters are required to store the trained model?

Solution

To store a trained SVM model, it is necessary to store S support vectors, each of which has a dimensionality of N. Moreover, the corresponding coefficients $\alpha_i y_i$ (one for each support vector) and the scalar parameter b should be kept. Therefore, the total number of parameters to store is

$$S \times N + S + 1 = S(N+1) + 1. \tag{1}$$

Note that, due to the constraint $\sum_i^M \alpha_i y_i = \sum_i^S \alpha_i y_i = 0$, an option is to store one less parameter. However, such choice is rarely used in practice.

■

B) Assume that each parameter has a float data type that takes 8 bytes in memory. What is the required memory to store the model if dataset is 100-dimensional (i.e., $N = 100$) and the number of support vectors is $S = 10,000$?

Solution

Using eq. (1) with $S = 10,000$ and $N = 100$ will amount to

$$10,000 \times (100 + 1) = 1,010,000$$

float-type parameters. Note that the $+1$ term corresponding to b is omitted since it does not make a significant difference. Thus, given that each float number takes up 8B (8 bytes), the required memory is

$$\frac{1,010,000 \times 8\text{B}}{1024 \times 1024} \approx 7.71\text{MB}.$$

■

C) Assume that, in the previous question, only 1% of all datapoints became support vectors, meaning that total number of datapoints is $M = 1,000,000$. The training time complexity for SVM is $O(MN^2)$. Furthermore, assume that training in a smaller problem with $M = 1000$ and $N = 10$ takes 0.1 second. How much time do we approximately need to train the classifier for the initial problem?

Solution

The initial problem with $M = 1,000,000$ has 1000 times more datapoints than the smaller dataset, and 10 times larger dimensionality. As such, training time of the larger dataset will be $1000 \times 10^2 = 100,000$ of that of the smaller dataset, and is equal to

$$100,000 \times 0.1\text{s} = 10,000\text{s} = 166\frac{2}{3}\text{min} \approx 2.78\text{h} \approx 2\text{h}47\text{min}.$$

■

D) Average modern laptop CPU requires 50W of power under full load. Using the time from the previous question, how much energy (in Wh) does one need to train such SVM? For comparison, note that a regular kettle draws 1500W, and it takes 5 minutes to boil approximately 2 liters of water. Training this SVM model is equivalent to boiling how many liters of water?

Solution

If CPU draws 50W, then 2.78h of training requires 2.78h $\times$ 50W = 139Wh of energy. On the other hand, boiling the regular kettle of water for 5 minutes demands $\frac{5}{60}$h $\times$ 1500W = 125Wh. Hence, training this SVM model is roughly equivalent to boiling 2 liters of water.

2 Classification with SVM

A) Consider a 2-dimensional classification problem with only 2 datapoints, including $\boldsymbol{x}^1 = [0.5, 0.5]^\top$ and $\boldsymbol{x}^2 = [-0.5, -0.5]^\top$ with $+1$ and -1 class labels, respectively (see fig. 1). Compute the coefficients α_i and the bias term b for a SVM classifier run on this problem with an RBF kernel ϕ, where $k(\boldsymbol{x}^1, \boldsymbol{x}^2) = 0.5$. Moreover, draw the isolines of the classifier function and the classifier hyperplane.

Hint: Recall that the SVM classifier function is given by

$$f(\boldsymbol{x}) = \text{sign}\left(\sum_i \alpha_i y_i k(\boldsymbol{x}, \boldsymbol{x}^i) + b\right). \tag{2}$$

Furthermore, the necessary conditions for optimality are provided below.

$$\begin{cases} \boldsymbol{w} = \sum_i \alpha_i y_i \phi(\boldsymbol{x}^i) \\ \sum_i \alpha_i y_i = 0 \quad \text{(appearing in the dual problem)} \\ y_i \left(\sum_j \alpha_j y_j k(\boldsymbol{x}^j, \boldsymbol{x}^i) + b\right) \geq 1, \quad \forall i = 1, \dots, M \quad \text{(primal feasibility)} \\ \alpha_i \geq 0, \quad \forall i = 1, \dots, M \quad \text{(dual feasibility)} \\ \alpha_i \left(y_i \left(\sum_j \alpha_j y_j k(\boldsymbol{x}^j, \boldsymbol{x}^i) + b\right) - 1\right) = 0, \quad \forall i = 1, \dots, M \quad \text{(KKT condition)} \end{cases}. \tag{3}$$

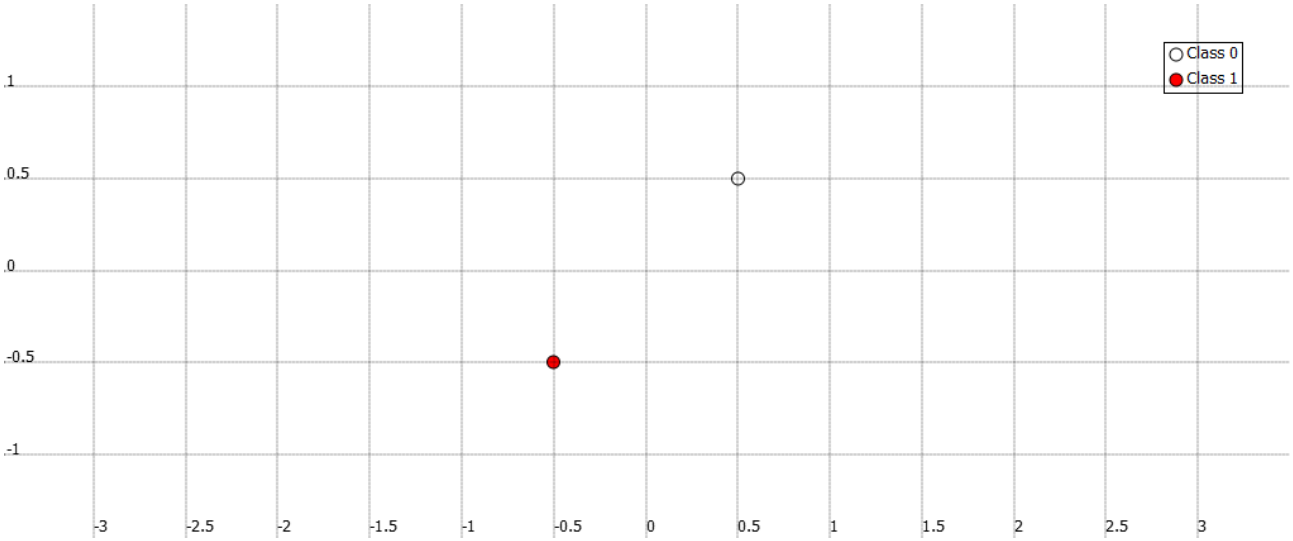


Figure 1: Question 2.A

Solution

Since there are only two data points, both datapoints must be support vectors in order to satisfy the constraint $\sum_i \alpha_i y_i = 0$ mentioned in eq. (3). Each point is located exactly on either side of the margin. Hence, the value of the classifier function at each support vector $\boldsymbol{x}^i$ is equal to ± 1 depending on its label y_i. In other words, it follows from eq. (2) that

$$\begin{cases} \sum_{i=1}^{M} \alpha_i y_i k(\boldsymbol{x}^1, \boldsymbol{x}^i) + b = 1 \\ \sum_{i=1}^{M} \alpha_i y_i k(\boldsymbol{x}^2, \boldsymbol{x}^i) + b = -1 \end{cases}. \tag{4}$$

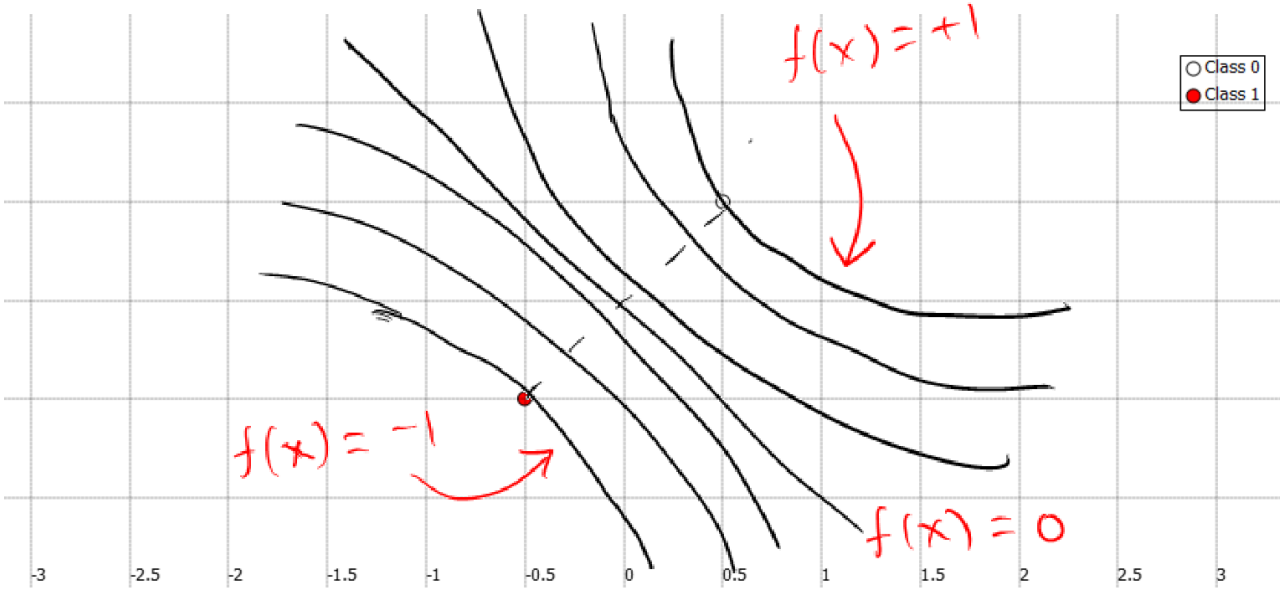


Figure 2: Question 2.A - Solution

The constraint $\sum_i \alpha_i y_i = 0$ reduces to $\alpha_1 y_1 + \alpha_2 y_2 = 0$. As such, with $y_1 = +1$ and $y_2 = -1$, we obtain $\alpha_1 = \alpha_2$. Combining this with $k(\boldsymbol{x}^1, \boldsymbol{x}^1) = k(\boldsymbol{x}^2, \boldsymbol{x}^2) = 1$ and $k(\boldsymbol{x}^1, \boldsymbol{x}^2) = k(\boldsymbol{x}^2, \boldsymbol{x}^1) = 0.5$, eq. (4) will be simplified into:

$$\begin{cases} 0.5\alpha_1 + b = 1 \\ -0.5\alpha_1 + b = -1 \end{cases} . \tag{5}$$

Summing the above equations will result in $b = 0$. Putting the value of b back into one of the equations listed in eq. (5) will yield $\alpha_1 = 2$. Therefore, the parameters of this SVM classifier are $\alpha_1 = \alpha_2 = 2$ and $b = 0$. The separating hyperplane and the isolines are illustrated in fig. 2.

■

B) Two more points are added to this dataset in different ways as illustrated in figs. 3 and 4. How would the α_i and b parameters change in each case? Draw the support vectors and the classification boundary for each case.

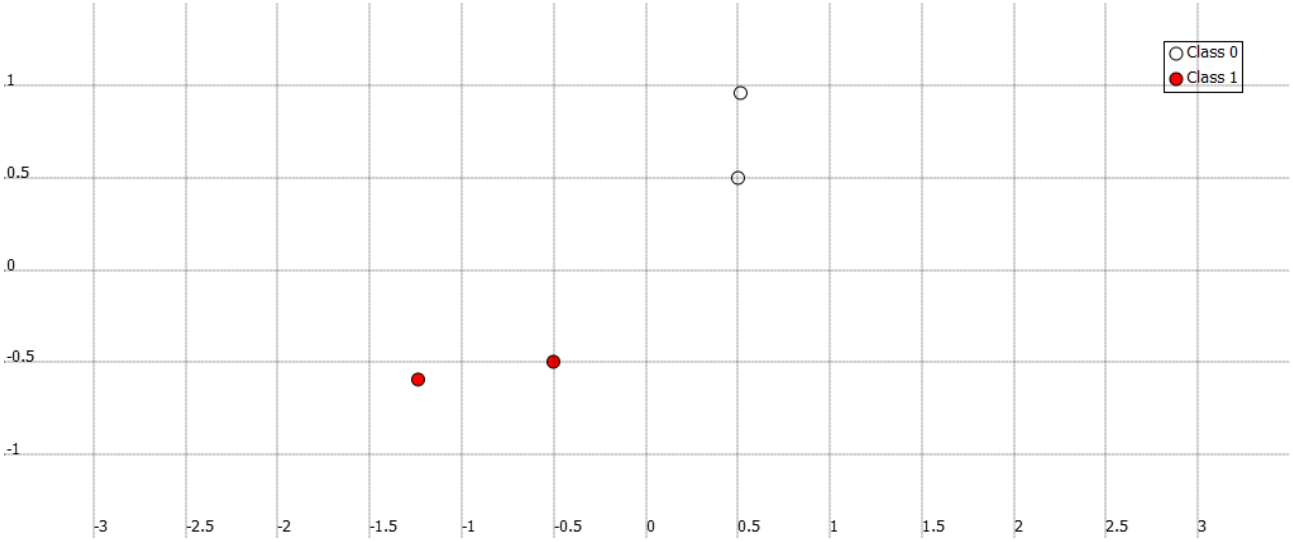


Figure 3: Question 2.B - Case (1)

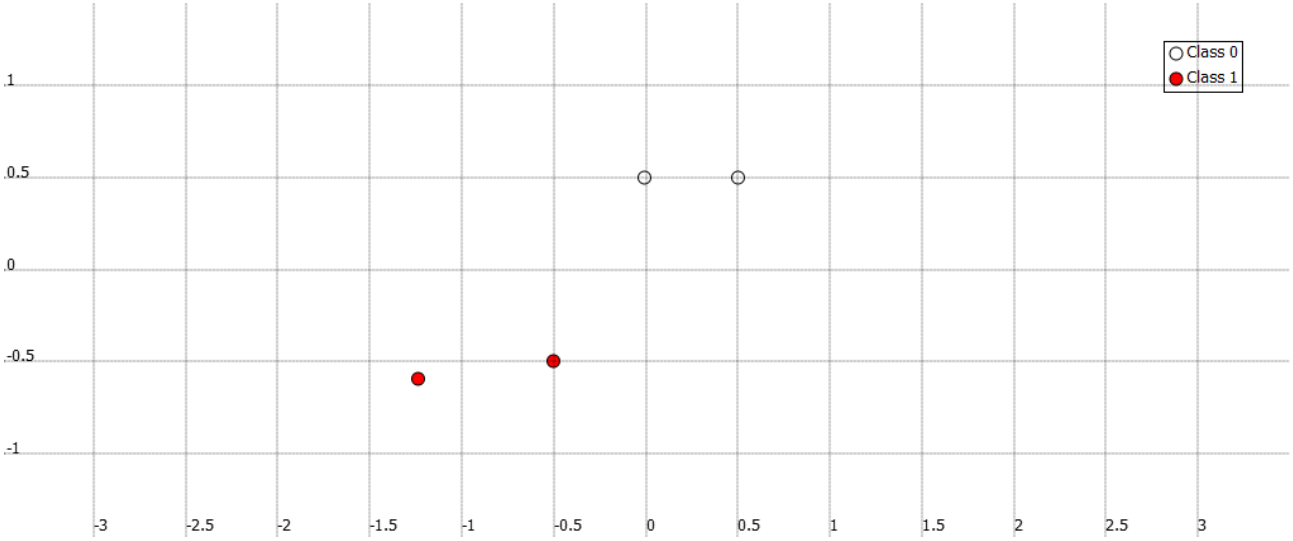


Figure 4: Question 2.B - Case (2)

Solution

Case (1): Since the new points lay outside the margin, the separating hyperplane and the support vectors will remain unchanged (see fig. 5).

Case (2): The point added to the white class is inside the original margin; hence, it becomes a support vector instead the original point in the white class (see fig. 6).

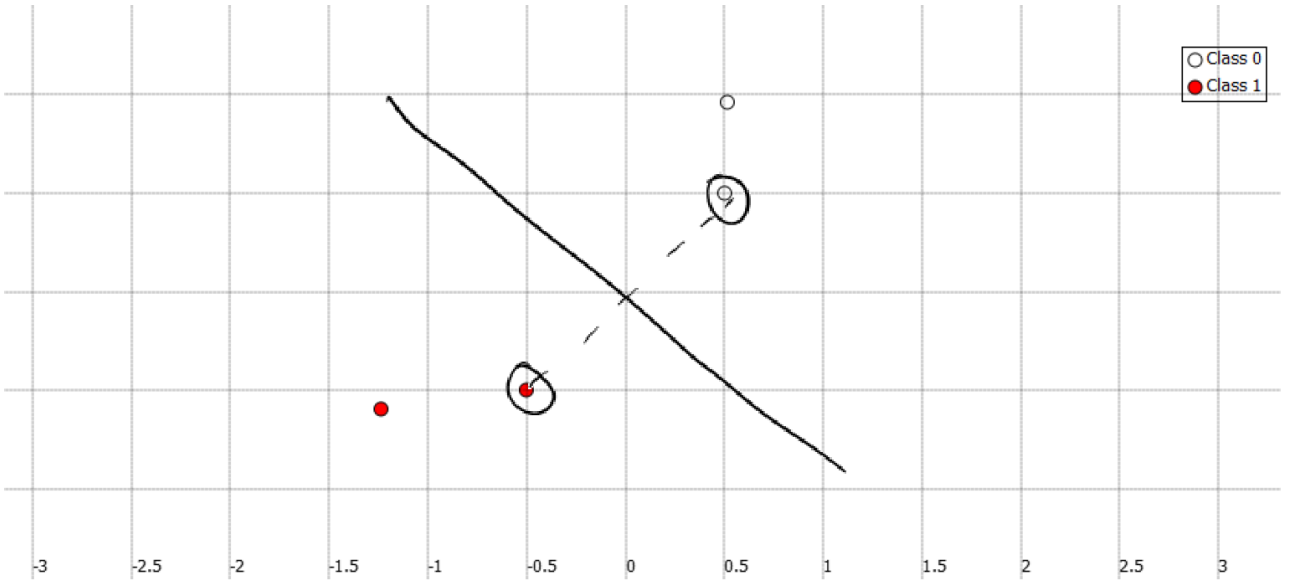


Figure 5: Question 2.B - Case (1) - Solution

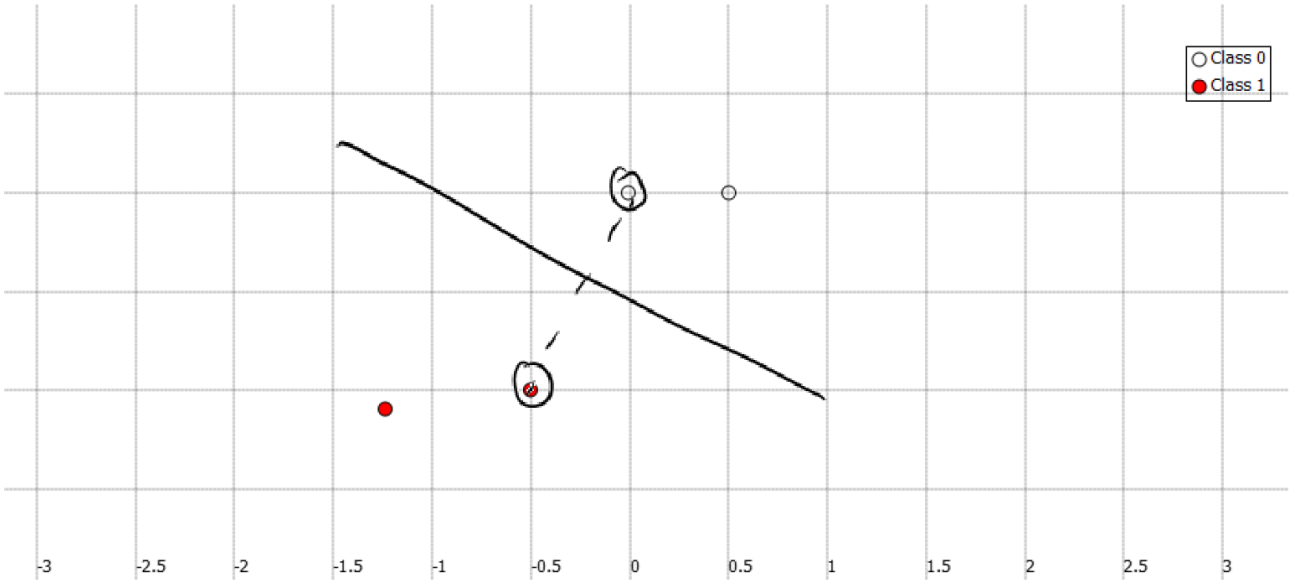
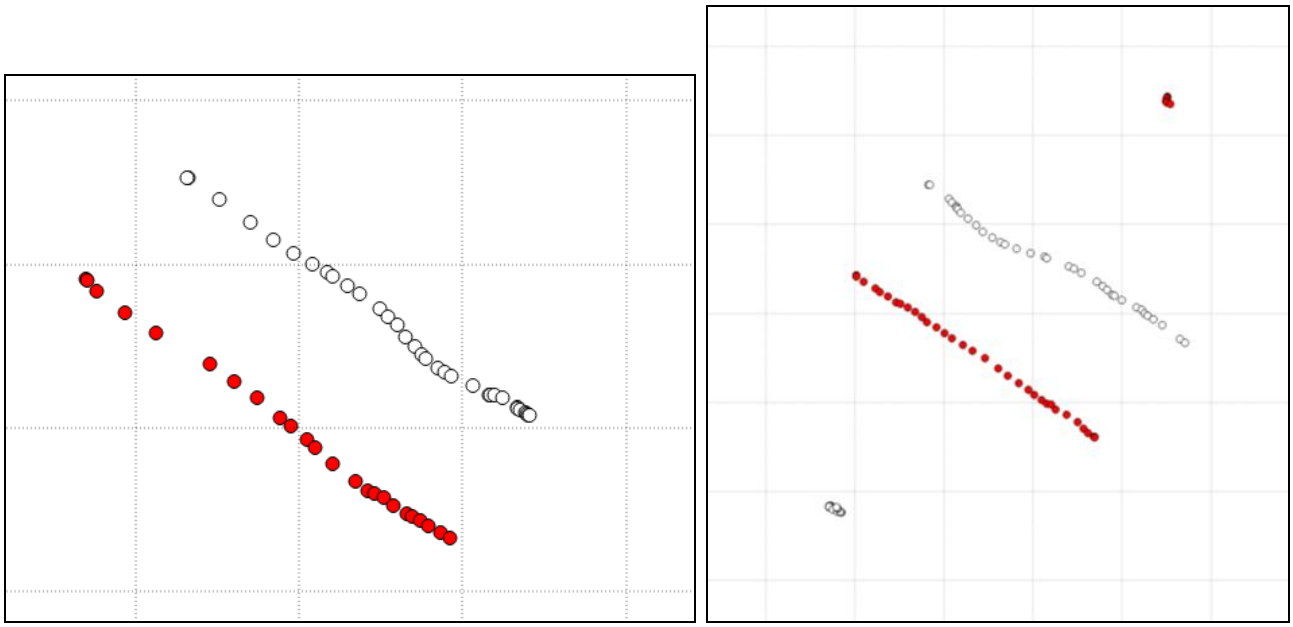


Figure 6: Question 2.B - Case (2) - Solution

C) Consider the binary classification problem among red and white classes shown in fig. 7. For the case of SVM with an RBF kernel, draw the separating line in each case. Do not compute it nor run MLDemos; instead, infer what the line would look like from your intuition. Discuss how this line changes as a function of the penalty factor C and the kernel width σ.



(i) **(ii)**

Figure 7: Question 2.C

Solution

Case (i): The separating line is unaffected by the value of the penalty C since both classes are perfectly separable. The separating line is a straight line passing in-between the two classes. The kernel width affects only the number of support vectors. More specifically, the smaller the kernel width, the more support vectors (see fig. 8).

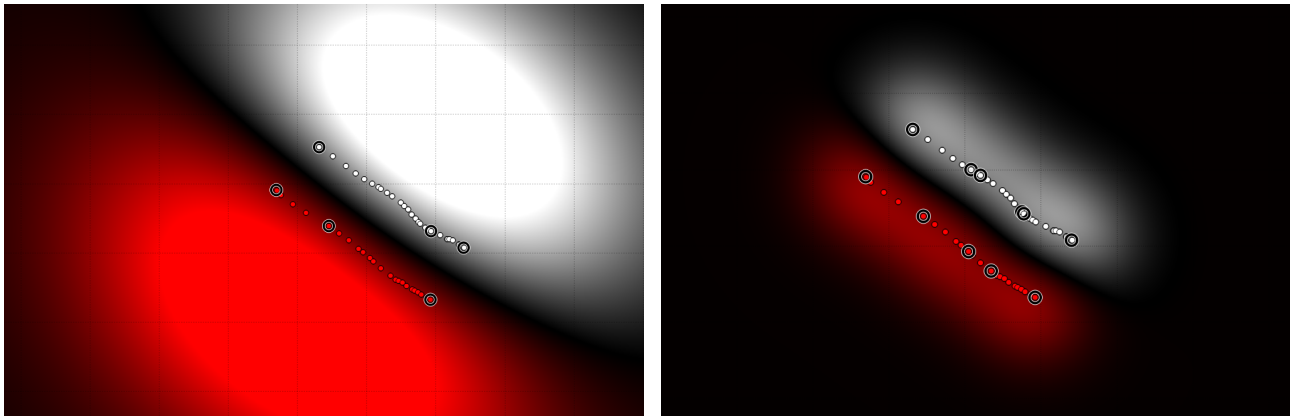


Figure 8: Solution found for case (i) with $\sigma = 0.1$ (left) and $\sigma = 0.01$ (right).

Case (ii): Here, the separating line changes as a function of both the penalty factor C and kernel width σ as presented in figs. 9 and 10.

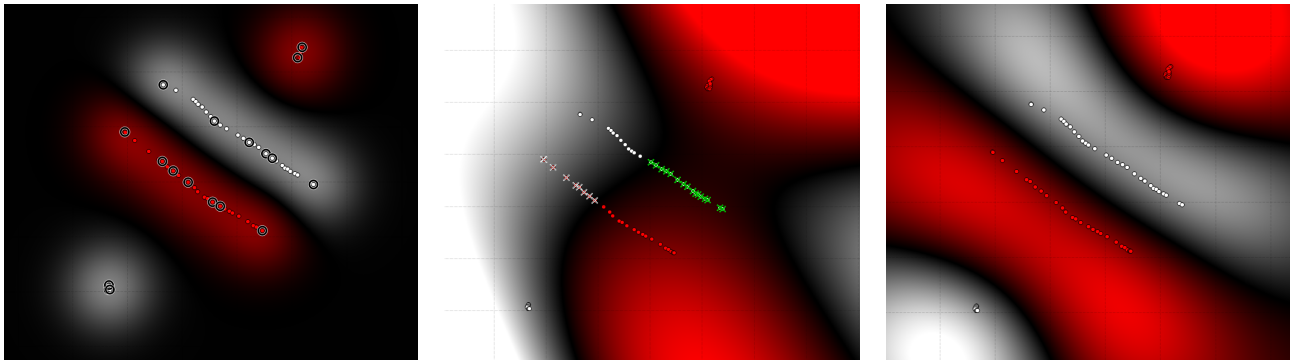
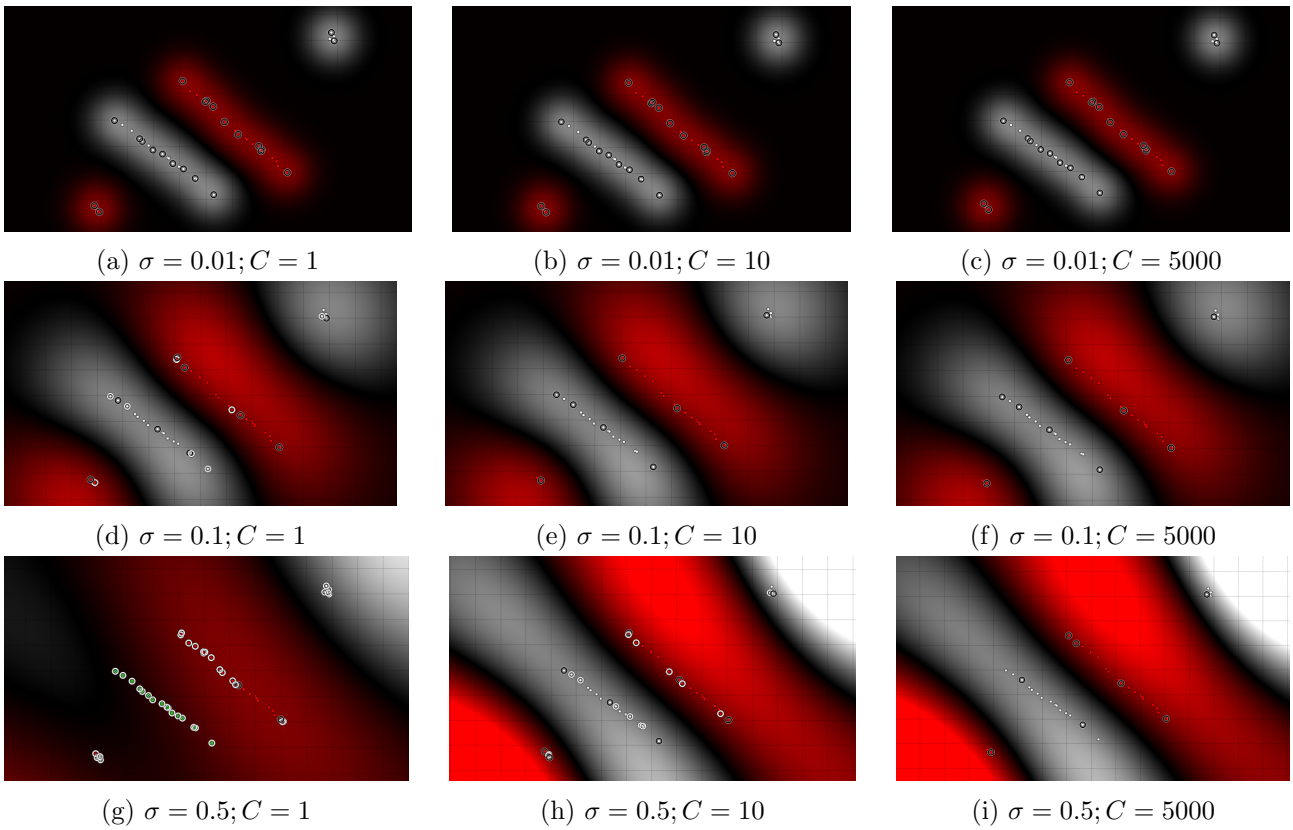


Figure 9: Solution found for case (ii) with (left) small kernel width ($\sigma = 0.01$) and large penalty ($C = 5000$) leads to perfect classification; however, this can also be viewed as overfitting. (Middle) Large kernel width ($\sigma = 0.5$) and small penalty ($C = 10.0$) yields incorrect classification. (Right) Correct values of kernel width ($\sigma = 0.1$) and penalty ($C = 1000$) bringing about a good classification with no overfitting.



(a) $\sigma = 0.01; C = 1$ (b) $\sigma = 0.01; C = 10$ (c) $\sigma = 0.01; C = 5000$

(d) $\sigma = 0.1; C = 1$ (e) $\sigma = 0.1; C = 10$ (f) $\sigma = 0.1; C = 5000$

(g) $\sigma = 0.5; C = 1$ (h) $\sigma = 0.5; C = 10$ (i) $\sigma = 0.5; C = 5000$

Figure 10: Effect of having different values of kernel width σ and penalty factor C in case (ii).