1 SVM

A) Question 1: To store a trained SVM model, it is necessary to store S support vectors (each has dimension N), as well as corresponding coefficients α (one for each support vector) and one for the scalar b. Total amount of floats will be: $S \times N + S + 1$. Additionally, as alphas are normalized ($\sum_i \alpha_i = 1$), we can store one less float, but that is rarely used in practice.
Question 2: With $S = 10,000$ and $N = 100$ we need $10,001 \times 100 = 1,000,100$ floats (omitting $+1$ since it's too small). If each float takes 8 bytes, then we need $1,000,100 \times 8/1024/1024 = 7.63$ megabytes to store the model.
Question 3: Large problem (with $M = 1,000,000$) has 1000 times the datapoints than smaller model, and 10 times larger dimensionality, meaning that training will be 100,000 times longer and equal to 10,000 seconds (approx. 2.8h or 2h42min)
Question 4: If CPU draws 50W, then in 2.8h of training the model it requires $2.8 \cdot 50 = 140$Wh of energy. Boiling a full kettle of water is equivalent to $\frac{5}{60} \cdot 1500 = 125$Wh, meaning that training one complex SVM model is roughly equivalent to boiling 2 liters of water.

B) Since there are only two data points, both datapoints must be support vectors in order to satisfy the constraint $\sum_i \alpha_i y_i = 0$ (this constraint follows from the dual, see the class's lecture notes). Each point is located exactly on either side of the margin. Hence, the value of the classifier function at each support vector $\mathbf{x}^i$ is equal to ± 1 depending on the datapoint's label y_i. This can be written as:

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

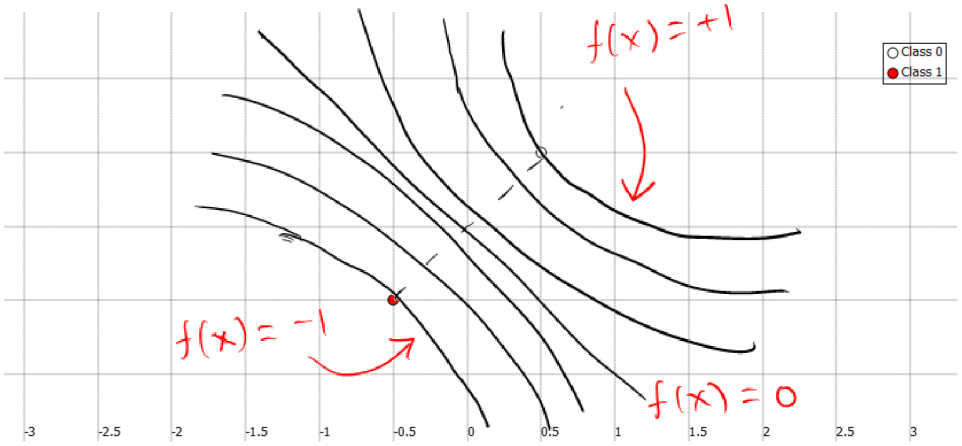


Figure 1:

The constraint $\sum_i \alpha_i y_i = 0$ gives us $\alpha_1 y_1 + \alpha_2 y_2 = 0$. With $y_1 = +1, y_2 = -1$, we obtain $\alpha_2 = \alpha_1$. Combining this with the fact that $k(\mathbf{x}^i, \mathbf{x}^i) = 1$ for $i = 1, 2$ and $k(\mathbf{x}^1, \mathbf{x}^2) = k(\mathbf{x}^2, \mathbf{x}^1) = 0.5$ (given), we can write the System 1 as follows:

$$\begin{cases} \alpha_1 - 0.5\alpha_1 + b = 1 \\ 0.5\alpha_1 - \alpha_1 + b = -1 \end{cases} \tag{2}$$

i.e.

$$\begin{cases} 0.5\alpha_1 + b = 1 \\ -0.5\alpha_1 + b = -1 \end{cases} \tag{3}$$

Summing the above equations we get $b = 0$. Putting value of b back into one of the equations we get $\alpha_1 = 2$. Hence, the parameters of this SVM are $\alpha_1 = \alpha_2 = 2$ and $b = 0$.

C) Case 1: Since the new points lay outside the margin, the separating hyperplane and its margin remain unchanged (see Fig. 2).

Case 2: The point added to "Class 0" is inside the margin, and hence it becomes a support vector instead of the old point. If we recalculate the whole system again with the same value of $k(\mathbf{x}^1, \mathbf{x}^2)$, we will get the same solution for b. However, the α will change as the value of $k(\mathbf{x}^1, \mathbf{x}^2)$ changed because of the new boundary (see Fig. 3).

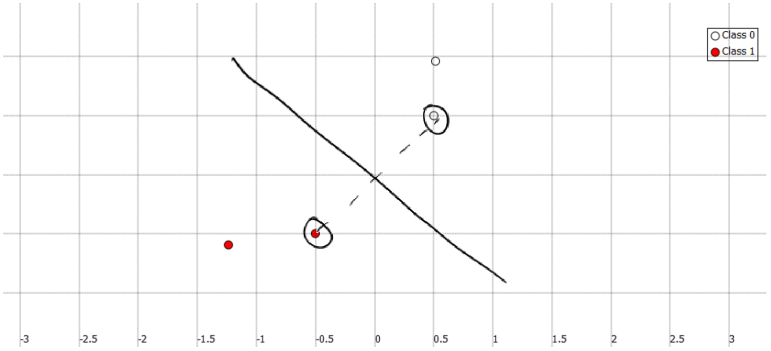


Figure 2: SVM

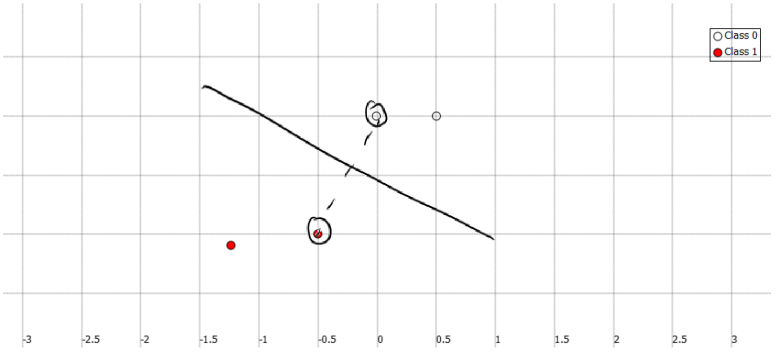


Figure 3: SVM

D)

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. The smaller the kernel width, the more support vectors. This is illustrated in Fig. 4.

ii In this case the separating line changes as a function of C as illustrated by Fig. 5. Fig. 6 shows the effects of the kernel width and C for different values and how they influence the resulting boundary regions.

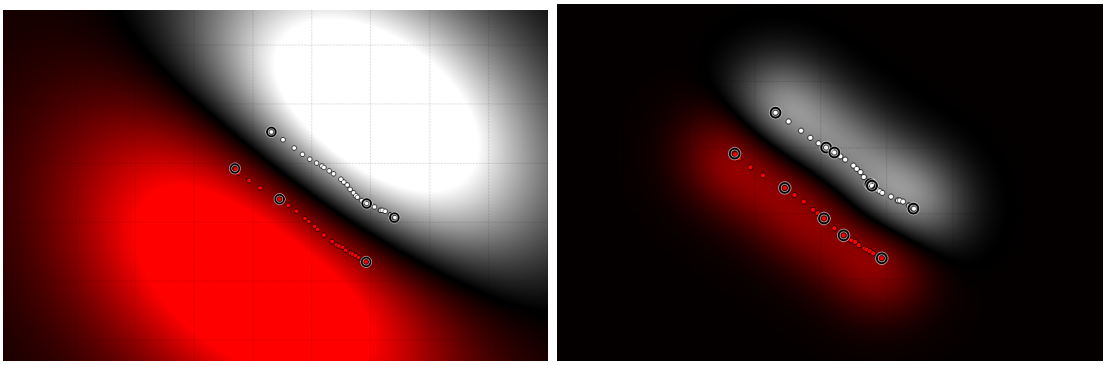


Figure 4: Solution found for example (i) with kernel width 0.1 (Left) and 0.01 (Right).

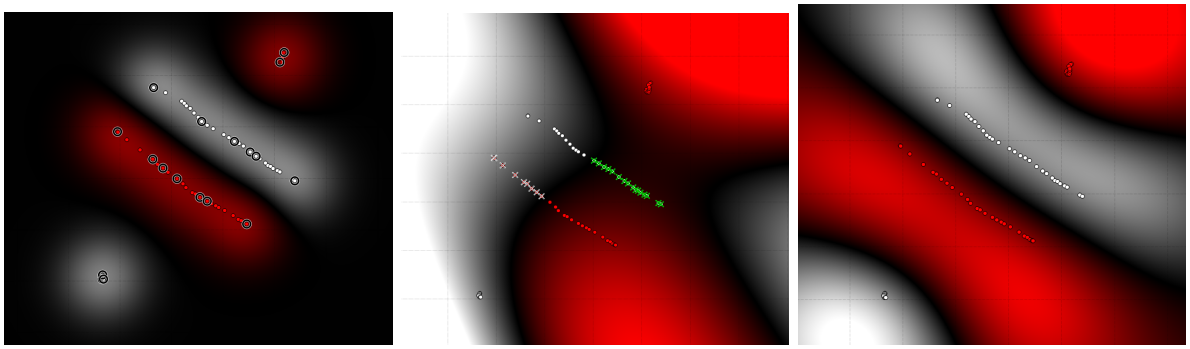
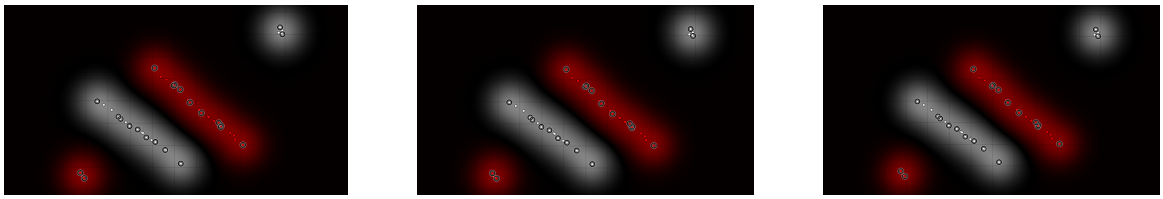
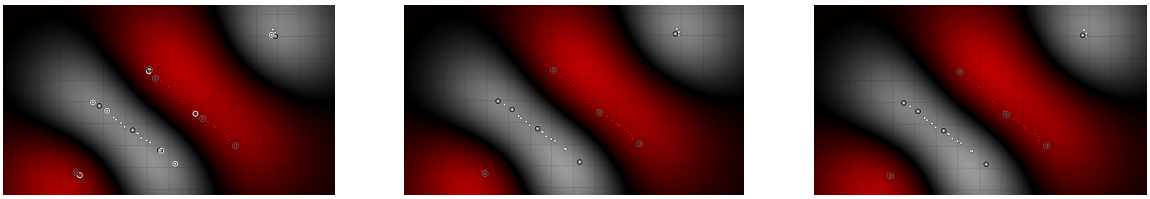


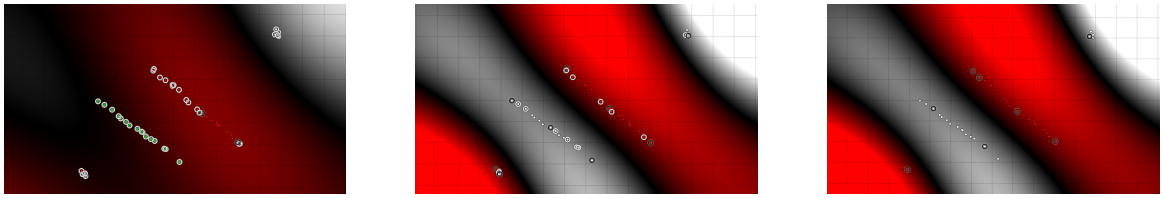
Figure 5: Solution found for example (ii) with (Left) Small kernel width (0.01) and large C (5000) leads to perfect classification but this can also be viewed as overfitting. (Middle) Very large kernel width (0.5) and very small C (10.0) yields incorrect classification. (Right) Correct level of kernel width (0.1) and C (1000) that results in good classification with no overfitting.



(a) kernel width = 0.01; C = 1 (b) kernel width = 0.01; C = 10 (c) kernel width = 0.01; C = 5000



(d) kernel width = 0.1; C = 1 (e) kernel width = 0.1; C = 10 (f) kernel width = 0.1; C = 5000



(g) kernel width = 0.5; C = 1 (h) kernel width = 0.5; C = 10 (i) kernel width = 0.5; C = 5000

Figure 6: SVM applied to the same dataset with varying kernel width and C parameters.

2 Optimization of SVM

A: Convex Optimization: multiplicity of solutions in SVM The variables to the dual SVM optimization are the Lagrange parameters α_i, with one Lagrange parameter per datapoint, i.e. $i = 1...M$. As per the KKT conditions, the Lagrange parameters represent the weight given to each datapoints to construct $w = \sum_i \alpha_i x^i$.

Can we find different sets of α_i that lead to the same optimum?

Let $w = \alpha_1 x^1 + \alpha_2 x^2$ be the optimal w. Since none of the datapoints are collinear, any pair of two points is linearly independent. Hence, each point can be expressed as linear combination of two other points.

We can hence construct $x^2 = \beta_1 x^1 + \beta_2 x^3$ with appropriate scalars β_1, β_2. Replacing x^2 in w, we obtain a new set of α_i for the same optimal w, namely $w = (\alpha_1 + \beta_1)x^1 + \beta_2 x^3$.

B: Margin The KKT condition $\sum_i \alpha_i y_i = 0$ implies that we have at least two support vectors, one in each class. Hence, there exist two points , which we denote as x^1 and x^2 with $y_1 = 1$ and $y_2 = -1$, for which the constraints $y_i(w^T x^i + b) = 1$ are satisfied.

We modify the constraint and set that all support vectors lie on a plane with equation $y_i(w^T x^i + b) = a$, with $a > 0$. We have:

$$\begin{cases} w^T x^1 + b = a \\ w^T x^2 + b = -a \end{cases} \tag{4}$$

Substracting the two lines, we get $w^T(x^2 - x^1) = 2a$. Expanding the inner product, $\|w\| = \frac{2a}{\|(x^2 - x^1)\| \cos(\theta)}$. θ is the angle between w and the vector $x^2 - x^1$. We see that the factor a only scales the norm of the vector w, but does not affect the choice of Support Vectors. It does not change the direction of w and hence does not affect the orientation of the hyperplane.

C: Convexity of the relaxed problem Is $f(w,\xi) = \|w\|^2 + C\sum_i \xi_i$, $\xi_i > 0 \forall i, C \geq 0$ convex?

$f(w,\xi) = \|w\|^2$ is strictly convex and $\sum_i \xi$, $\xi > 0$, $\forall i$ is convex. Since the quadratic term is strictly convex and grows faster than the linear term, the objective function is strictly convex. It hence admits a single global optimum.

The addition of the slack variables, however, can shift the optimum of the objective function to a solution that is not the true optimum (without relaxation of constraints). The relaxed optimization finds an optimal solution that is a tradeoff between augmenting the margin across the two classes (reducing the first term of the cost function) and reducing the cost of violating one or more constraints (reducing the second term of the cost function).

The penalty associated to the violation of the constraint is conveyed through the choice of the constant C. A large C will tend to force the optimization to find a solution close to the unrelaxed problem. This is illustrated in Figure 7. When applying a small penalty, $C = 5$, for a violation of the constraints, the optimization finds a separating hyperplane with a larger margin than with a hight penalty, $C = 100$.

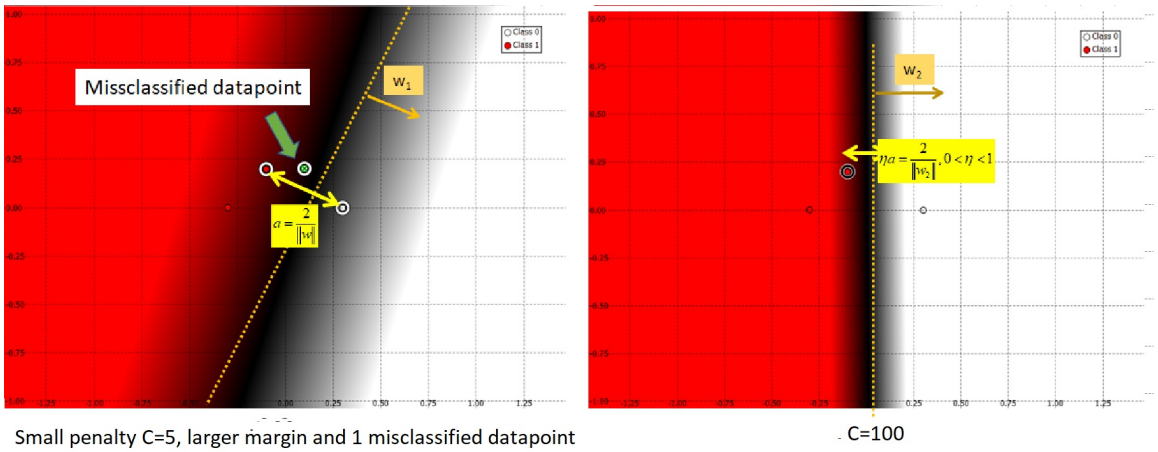


Figure 7: Optimal solution of the relaxed SVM optimization when using a low penalty on slacks $C = 5$ versus a high penalty, $C = 100$.

D: Optimum of the relaxed problem: The true optimal solution to SVM is obtained for an optimal value to the objective function and satisfaction of all constraints. In the relaxed problem, the objective function is given by: $min_{w,\xi}\|w\|^2 + \frac{C}{M}\sum_i \xi_i$, with $C \geq 0$ a constant penalizing for the introduction of slacks and M the number of datapoints. Observe that the SVM objective function is composed of a quadratic and linear cost, both of which are proportional to the width of the margin, which we denote as a.

Consider the group of four points in Figure 7. The two hyperplanes generated by w_1 amd w_2, both optimal solutions for different values of C.

The first hyperplane defined by w_1 has a margin equal to $\|w_1\|^2 = \frac{2}{a^2}$. One of the two points from the white class is missclassfied. The costs associated to the constraint's violation for this point is entailed in the associated slack ξ. We show next that the slack is proportional to the distance to the hyperplane.

Without loss of generality, we can assume $b = 0$ (shift of the origin). The constraints are satisfied at equality for the two datapoints on the margin and for the point inside the margin with slack ξ. For the latter, we have:

$$\begin{cases} w_1^T x^i = 1 + x, \\ \xi = \|w_1\|\|x\| - 1. \end{cases} \tag{5}$$

The second hyperplane w_2 satisfies all constraints, hence $\xi = 0, \forall i$ and is solution to $\|w_2\|^2 + C\sum_i \xi_i = \|w\|^2 = \frac{2}{(\eta a)^2}$, $0 < \eta \leq 1$.

To determine if a solution with slack can lead to a value on the objective function that is equal or better than the solution without slack, one must hence check whether $\|w_1\|^2 + C\xi = \frac{2}{a^2} + C\frac{1}{\eta a} \leq \|w_2\|^2 = \frac{2}{(\eta a)^2}$. Many cases will arise depending on the values of C and η. Observe that the associated cost on the objective function to enlarging the margin is privileged over violating constraints, as the former grows quadratically with the margin whereas the latter grows linearly. The solver will hence tend to privilege solutions with small violation of constraints if these lead to an increase in the margin. The shift of the optimum is illustrated in Figure 8.

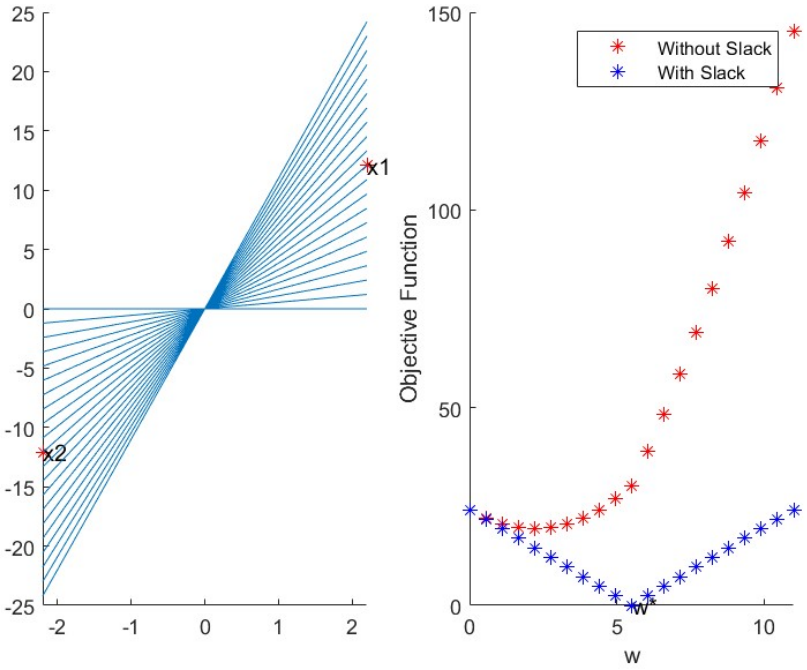


Figure 8: (Left) distribution of separating hyperplanes across a pair of datapoint. (Right) evolution of optimum on SVM objective function for the distribution of hyperplane with and without slack.