SMAI-S25-L18: SVMs

C. V. Jawahar

IIIT Hyderabad

March 25, 2025

Perceptron Vs SVM

Perceptron finds a valid solution. SVM finds an optimal solution.

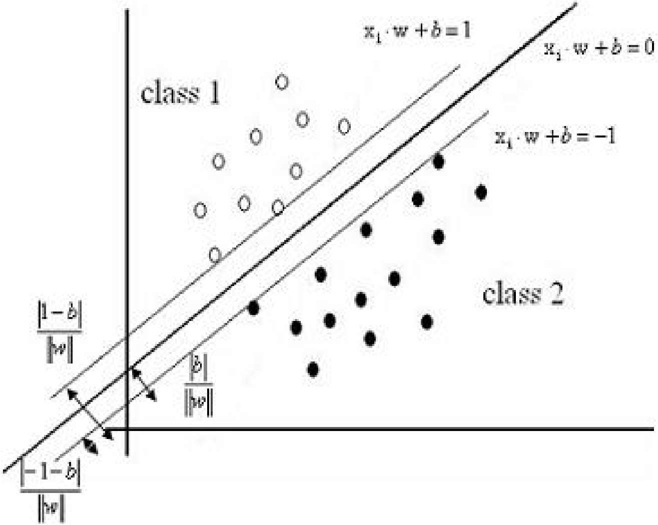
Advantage:

- Avoids overfitting and performs well on test data.
- SVM finds the hyperplane which has the maximum margin.
- Maximization of the margin also corresponds to the higher *generalization*.

Support Vector Machine (SVM)

SVM finds a separating solution(hyperplane) which "maximizes the margin".

SVM:Optimal Hyper Plane



Training Data

- Let there are N m-dimensional training inputs $\mathbf{x_i}(i = 1 \ldots N)$ from two different classes.
- We represent Class 1 and 2 by associating labels. Labels are $y_i = 1$ for Class 1 and -1 for Class 2. Decision function which needs to be determined is

$$f(\mathbf{x_i}) = \mathbf{x_i} \cdot \mathbf{w} + b$$

- Therefore,

$$\mathbf{x_i} \cdot \mathbf{w} + b > 0 \text{ for } y_i = +1$$
$$\mathbf{x_i} \cdot \mathbf{w} + b < 0 \text{ for } y_i = -1$$

Contraints

As training data are linearly separable (let us assume so, at this stage), no training data satisfy $\mathbf{x_i} \cdot \mathbf{w} + b = 0$, to control separability we can consider following inequalities:

$$\mathbf{x_i} \cdot \mathbf{w} + b \geq +1 \text{ for } y_i = +1$$
$$\mathbf{x_i} \cdot \mathbf{w} + b \leq -1 \text{ for } y_i = -1$$

Combining these into one inequality:

$$y_i(\mathbf{x_i} \cdot \mathbf{w} + b) - 1 \geq 0 \qquad \forall i$$

Terminology: Support Vectors

- The vectors which satisfy the equality in the below equation are called *support vectors*.

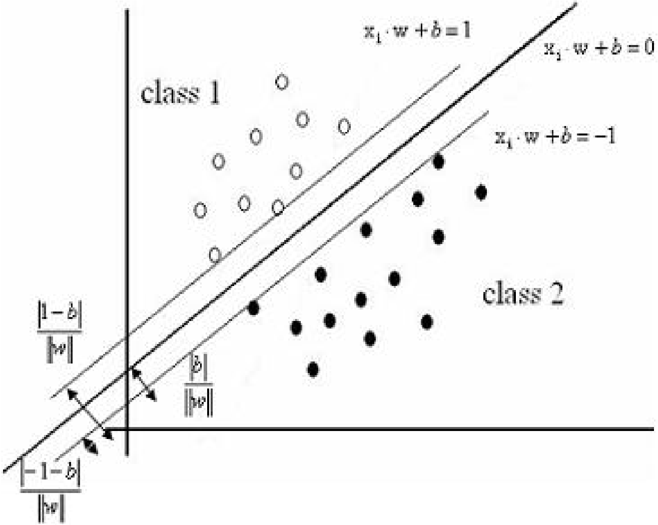
$$y_i(\mathbf{x_i} \cdot \mathbf{w} + b) - 1 \geq 0 \qquad \forall i$$

- They are the points which lie closest to the decision surface and are therefore the most difficult to classify.
- They have a direct bearing on the optimum location of the decision surface. (*In fact nobody else has ...!!*)

Optimal Hyperplane: Formulation

- The points where equality is valid, will lie on the hyperplanes H1 (i.e, $\mathbf{x_i} \cdot \mathbf{w} + b = 1$) and H2 (i.e, $\mathbf{x_i} \cdot \mathbf{w} + b = -1$).
- These two are parallel to the optimal hyperplane $\mathbf{x_i} \cdot \mathbf{w} + b = 0$.
- All these planes are at distance $|1 - b|/||\mathbf{w}||$ (H1), $|-1-b|/||\mathbf{w}||$ (H2) and $|b|/||\mathbf{w}||$ (Optimal) from origin.
- Therefore the *margin*, the distance between H1 and H2 is $\frac{|1-b|}{||\mathbf{w}||} - \frac{|-1-b|}{||\mathbf{w}||} = \frac{2}{||\mathbf{w}||}$.

SVM:Optimal Hyper Plane



Maximization of Margin

- Maximization of the margin $\frac{2}{||\mathbf{w}||} = \frac{2}{\mathbf{w}^T\mathbf{w}}$ is equivalent to minimization of $\mathbf{w^T w}$.
- An unconstrained optimization may result in $\mathbf{w} = \mathbf{0}$. Therefore, we do minimize with the constraints derived above. ($y_i(\mathbf{w} \cdot \mathbf{x_i} + b) - 1 \geq 0 \forall i$)
- Constraint says that all training samples are "correctly classified".
- To make some of the expressions simple, we make the objective function as $\frac{1}{2}\mathbf{w^T w}$

Objective Formulation

Objective is to maximise margin and corresponding mathematical formulation is

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w}$$
$$\textit{subject to} \quad y_i(\mathbf{w^T x_i} + b) - 1 \geq 0 \forall i$$
$$y_i \in \{-1, 1\}$$

Primal to Dual

Primal: SVM problem is that of

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w}$$
$$\textit{subject to} \quad y_i(\mathbf{w^T x_i} + b) - 1 \geq 0 \forall i$$
$$y_i \in \{-1, 1\}$$

Dual: This results in maximization of

$$J_d(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x_i^T x_j}$$

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

Dual variable to Primal ones

Typically the dual function gets solved for α.

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$b = \frac{1}{|S|} \sum_{i \in S} (y_i - \mathbf{w^T x_i})$$

Comments

- Support Vector Machines could be understood as learning machines with maximal margin property.
- The vectors which lie at exactly at margin are the support vectors.
- The error rate of a learning machine on a test data (i.e., generalization error) is bound by the training error rate and a term that depends on the VC dimension of the machine.
- In the case of separable patterns SVMs produce zero for the first term (training error) and minimise the second term.
- Realize that the linear discriminant functions were interested only in minimising the first term.

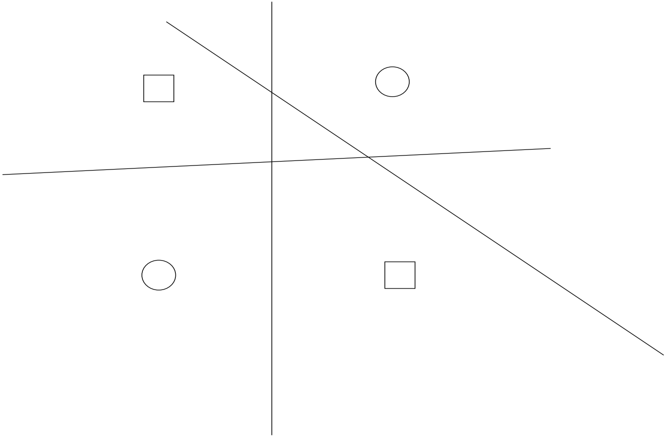
Support Vectors and Importance

- Support vectors are the ones at unit distance from the hyperplane.
- The objective function $J_d(\cdot)$ to be maximised depends *only* on the input patterns in the form of a set of dot products $\{\mathbf{x_i^T x_j}\}$
- From the optimal values of α's, we can compute the weight vector $\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$
- No prior knowledge of the problem.

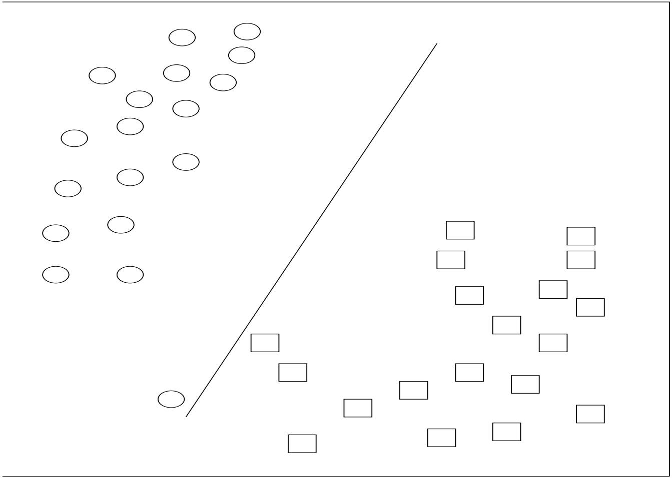
Train and Test

Training: During the training, one computes the SVM from the available data set. (Support vectors and the corresponding α)
Testing: On testing we simply determine on which side of the decision boundary a given test pattern $\mathbf{x}$ lies and assign the corresponding class label. i.e, we take the class of $\mathbf{x}$ to be $sgn(\mathbf{w} \cdot \mathbf{x} + b)$

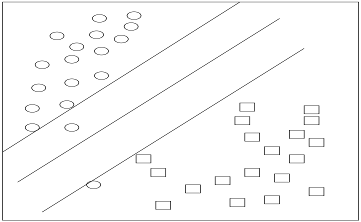
Linearly non-separable problems



Linearly separable problem ?

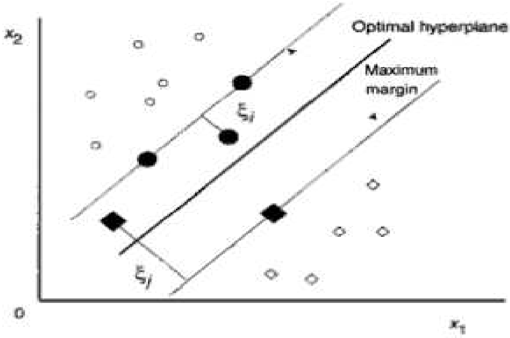


Linearly separable problem ?



Non-Separability

Inseparable case in a two-dimensional space.



Objective Function

The above formulation of separable problem can be extended to a non separable one without much of difficulty, by introducing a set of slack variables $\xi_i \;\; i = 1, \ldots, N$

$$\mathbf{x_i} \cdot \mathbf{w} + b \geq +1 - \xi_i \text{ for } y_i = +1$$

$$\mathbf{x_i} \cdot \mathbf{w} + b \leq -1 + \xi_i \text{ for } y_i = -1$$

$$\xi_i \geq 0 \forall i$$

Thus the problem becomes minimisation of

$$\frac{||\mathbf{w}||}{2} + C \sum_i \xi_i^k$$

instead of $\frac{||\mathbf{w}||}{2}$

Some standard formulations

L1 SVM : Objective is to minimize following function

$$\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i \xi_i$$

L2 SVM : Objective is to minimize following function

$$\frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{2}\sum_i \xi_i^2$$

Formulation: Recap

SVM problem is that of

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w}$$
$$\textit{subject to} \quad y_i(\mathbf{w^T x_i} + b) - 1 \geq 0 \forall i$$
$$y_i \in \{-1, 1\}$$

This results in maximization of

$$J_d(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x_i^T x_j}$$

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

NonLinear SVM's

Interestingly, the vectors appear as only dot product in the formulation. This allows us to solve the problem in a very high dimension (where the data set will well behave) without explicitly bothering about the mapping which converts into higher dimension.
We need only a kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$

$$K(\mathbf{s}_i, \mathbf{x}_i) = \Phi(\mathbf{s}_i) \cdot \Phi(\mathbf{x}_i)$$

Dual form

Dual formation of SVM is to maximise

$$J_d(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

subject to the constraints $\sum_{i=1}^{N} y_i \alpha_i = 0$ $, C \geq \alpha_i \geq 0$.
Kernalizing,

$$J_d(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

While testing,

$$\mathbf{w}^T \phi(\mathbf{x}_{test}) + b = \sum_i y_i \alpha_i \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_{test}) + b \tag{1}$$

$$= \sum_i y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_{test}) + b \tag{2}$$

In terms of kernel matrx

$$\max_{\alpha} \alpha^{\mathsf{T}}\mathbf{1} - \frac{1}{2}\alpha^{\mathsf{T}}\mathbf{K}\alpha$$

Subject to

$$\alpha^{\mathsf{T}}\mathbf{y} = 0$$

$$\alpha \geq \mathbf{0}$$

$$C\mathbf{1} - \alpha \geq \mathbf{0}$$

Adavantages

- **Maximization of generalization ability:**Support vector machine is trained to maximize the margin, the ability to generalization is the objective.
- **No local minima:** Support vector machine is formulated as a quadratic programming problem, there is a global optimum solution.
- **Robustness to outliers :** C controls the rate of missclassification. Outliers can be suppressed by properly setting a value to C.

Disadvantages

- **Extension to multiclass problems :** The extension to multiclass problem is not straightforward, and there are several formulations. Each of the formulation performs better to certain cases.
- **Long training time :** For very large training size solving dual is difficult from both memory and time point of view.
- **Selection of parameters :** In training we have to select appropriate kernel function and its parameters And also we need to fix value of parameter C.

Questions?

Extra Details

Lagrange Multipliers

Consider the optimization problem

$$\text{Maximize} \quad f(x, y)$$

$$\text{Subject to} \quad g(x, y) = b$$

We introduce a new variable (λ), called Langrange Multiplier, and study the Lagrange function defined by:

$$\Lambda(x, y, \lambda) = f(x, y) + \lambda \cdot (g(x, y) - b)$$

If (x', y') is a maximum for the original constrained problem, then there exists a λ such that (x', y', λ) is a stationary point for the Lagrange function.
(Note: Stationary points are those ponts where the partial derivatioves of Λ are zero)

Lagrange Method

The method of obtaining necessary conditions in the problem of determining an extremum of a function $f(x_1, x_2, \ldots, x_n)$ under the constraints

$$g(x_1, \ldots x_n) = b_i, \;\; i = 1, \ldots m$$

consisting of the use of Lagrange multipliers λ_i $i = 1, \ldots, m$ the construction of the Lagrange function

$$\Lambda(\mathbf{x}, \lambda) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i [b_i - g_i(\mathbf{x})]$$

and equating its partial derivatives with respect to x_j and λ_i to zero, is called the **Lagrange method**.
In this method, the optimal value $\mathbf{x}^* = (x_1^*, \ldots x_n^*)$ is found together with the vector of Lagrange multipliers $\lambda^* = (\lambda_1^*, \ldots \lambda_m^*)$ corresponding to it by solving the system of $m + n$ equations.

Optimal Hyperplane: Objective Function

Converting the constrained problem to unconstrained problem we have to minimise

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{N} \alpha_i \left[y_i(\mathbf{w^T x_i} + b) - 1 \right]$$

where $\alpha_i \geq 0$ are the nonnegative Lagrangian multipliers. The optimality conditions are:

$$\frac{\partial J(\mathbf{w}, b, \alpha)}{\partial \mathbf{w}} = \mathbf{0} \text{ and } \frac{\partial J(\mathbf{w}, b, \alpha)}{\partial \mathbf{b}} = \mathbf{0}$$

Optimal Hyperplane: Solution

Thus, minimise

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{N} \alpha_i \left[y_i(\mathbf{w}^\mathbf{T}\mathbf{x_i} + b) - 1 \right]$$

where $\alpha_i \geq 0$ are the nonnegative Lagrangian multipliers. The optimality conditions $\frac{\partial J(\mathbf{w},b,\alpha)}{\partial \mathbf{w}} = \mathbf{0}$ and $\frac{\partial J(\mathbf{w},b,\alpha)}{\partial \mathbf{b}} = \mathbf{0}$ leads to

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

Optimal Hyperplane: Solution(Cont.)

The objective function $J_d(\alpha)$ to be maximised becomes

$$J_d(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^{\mathbf{T}} \mathbf{x_j}$$

Thus find maxima of $J_d(\alpha)$ subject to $\sum_{i=1}^{N} \alpha_i y_i = 0$ and $\alpha_i \geq 0$.

Optimal Hyperplane: Solution(Cont.)

The objective function $J_d(\alpha)$ to be maximised becomes

$$J_d(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^\mathbf{T} \mathbf{x_j}$$

Thus find maxima of $J_d(\alpha)$ subject to $\sum_{i=1}^{N} \alpha_i y_i = 0$ and $\alpha_i \geq 0$.

$\boxed{\text{Minima of } J(w, b, \alpha) \text{ is same as Maxima of } J_d(\alpha).\ \text{Why?}}$

Primal Vs Dual

Consider a problem of minimizing $f(x)$ such that $\mathbf{g}(\mathbf{x}) \geq \mathbf{0}$.
The corresponding lagrangian function is

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda^{\mathbf{T}}\mathbf{g}(\mathbf{x})$$

Now,

$$\max_{\lambda \geq 0} L(\mathbf{x}, \lambda) = \begin{cases} \infty & \text{if } g(x) < 0 \\ f(x) & \text{otherwise} \end{cases}$$

$$\textbf{Primal Problem: } \min_{x} \max_{\lambda \geq \mathbf{0}} L(\mathbf{x}, \lambda)$$

$$\textbf{Dual Problem: } \max_{\lambda \geq \mathbf{0}} \min_{x} L(\mathbf{x}, \lambda)$$

Optimal Hyperplane in L1 SVM: Solution

In L1 SVM we have to minimise

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i \xi_i$$

$$\textit{subject to} \quad y_i(w^T x_i + b) \geq 1 - \xi_i \forall i$$

$$\xi_i > 0, y_i \in \{-1, 1\}$$

Or we have to minimise

$$J(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i \xi_i - \sum_{i=1}^{N} \beta_i \xi_i$$

$$-\sum_{i=1}^{N} \alpha_i \left[y_i(\mathbf{w^T x_i} + b) - 1 + \xi_i \right]$$

where $\alpha_i \geq 0$ and $\beta_i \geq 0$ are the nonnegative Langrangian multipliers.

Optimal Hyperplane in L1 SVM: Solution

The optimality conditions $\frac{\partial J(\mathbf{w}, b, \xi, \alpha, \beta)}{\partial \mathbf{w}} = \mathbf{0}$ and $\frac{\partial J(\mathbf{w}, b, \xi, \alpha, \beta)}{\partial \mathbf{b}} = \mathbf{0}$ and $\frac{\partial J(\mathbf{w}, b, \xi, \alpha, \beta)}{\partial \xi} = \mathbf{0}$ leads to

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

$$\alpha_i + \beta_i = C \quad \forall i$$

substituting above there equation in objective function we have following dual problem. Maximise

$$J_d(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

subject to the constraints $\sum_{i=1}^{N} y_i \alpha_i = 0$,$C \geq \alpha_i \geq 0$

Optimal Hyperplane in L1 SVM: Solution

The only difference betweeen L1 soft-margin support vector mahchines and hard margin support vector machines is that α_i cannot exceed C. Value C decides weight given for rate of missclassification.

Three cases for α_i :

1. $\alpha_i = 0$. Then $\xi_i = 0$. Thus $\mathbf{x_i}$ is correctly classified.
2. $0 < \alpha_i < C$. Then $y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i = 0$ and $\xi_i = 0$. Therefore, $y_i(\mathbf{w}^T\mathbf{x}_i + b) = 1$ and $\mathbf{x}_i$ is a support vector. Especially,we call the support vector with $C > \alpha_i > 0$ an <u>unbounded support vector</u>.
3. $\alpha_i = C$. Then $y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i = 0$ and $\xi_i \geq 0$. Thus ξ_i is a support vector. We call the support vector with $\alpha_i = C$ a <u>a bounded support vector</u>. If $0 \leq \xi_i < 1, \mathbf{x}_i$ is correctly classified. and if $\xi \geq 1 \mathbf{x}_i$ is misclassified.