Week 7 : Session 2

# 11. Dual SVM and Kernel SVM

↑ More general formulation of an SVM.

Recall, the problem to determine the SVM classifier with maximum margin is

$$\min \|\bar{a}\|$$

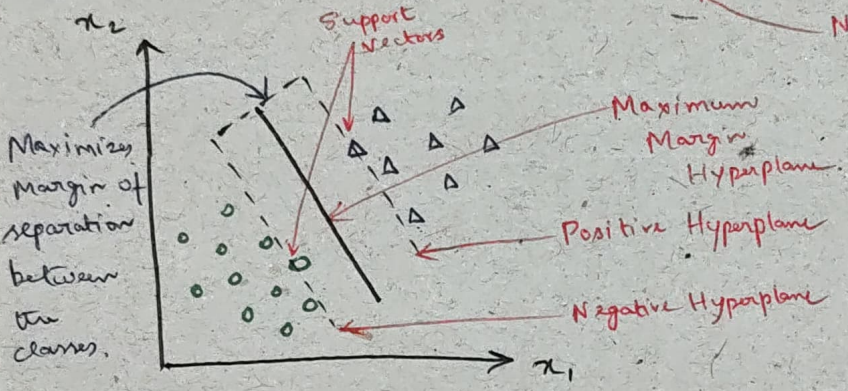
subject to the constraints

Positive Hyperplane

$$c_0 : \bar{a}^T \bar{x}_i + b \geq 1 \quad ; \; i = 1, 2, \ldots, M$$

$$c_1 : \bar{a}^T \bar{x}_i + b \leq -1 \quad ; \; i = M+1, M+2, \ldots, 2M$$

Negative Hyperplane




This is a convex optimization problem that can be solved very easily.

Let the response be defined as

$$c_0 : y_i = 1 \quad ; \; i = 1, 2, \ldots, M$$

$$c_1 : y_i = -1 \quad ; \; i = M+1, M+2, \ldots, 2M$$

The constraints can be expressed as

$$c_0 : 1 \left(\bar{a}^T \bar{x}_i + b\right) \geq 1$$

$$\Rightarrow \boxed{y_i \left(\bar{a}^T \bar{x}_i + b\right) \geq 1}$$

$$c_1 : -1 \left(\bar{a}^T \bar{x}_i + b\right) \geq 1$$

$$\Rightarrow \boxed{y_i \left(\bar{a}^T \bar{x}_i + b\right) \geq 1}$$

Constraints become same for both the classes!

Therefore, the constraints can be combined as

$$y_i\left(\bar{a}^T\bar{x}_i + b\right) \geq 1$$

$$\Rightarrow -y_i\left(\bar{a}^T\bar{x}_i + b\right) \leq -1$$

$$\Rightarrow \boxed{-\left(y_i\left(\bar{a}^T\bar{x}_i + b\right) - 1\right) \leq 0 \;;\; i = 1, 2, \cdots, 2M}$$

This is the Final version of the constraint.

Therefore, finally, the SVM classifier problem can be recast as

$$\min \|\bar{a}\| \equiv \min \frac{1}{2}\|\bar{a}\|^2$$

subject to the constraints

$$-\left(y_i\left(\bar{a}^T\bar{x}_i + b\right) - 1\right) \leq 0 \text{ , for all } i$$

Equivalent SVM optimization problem

Needless to say, this is also convex optimization problem.

---

Now, we approach the Lagrangian for this constrained optimization problem.

The Lagrangian function is

$$\frac{1}{2}\|\bar{a}\|^2 + \sum_{i=1}^{2M} \lambda_i \left(-\left(y_i\left(\bar{a}^T\bar{x}_i + b\right) - 1\right)\right).$$

Remember, to minimize this Lagrangian; we have to take Gradient w.r.t the unknown quantity $\bar{a}$ and set equal to 0

$$\Rightarrow \nabla_{\bar{a}}\left(\frac{1}{2}\bar{a}^T\bar{a} - \sum_{i=1}^{2M}\lambda_i\left(y_i\left(\bar{a}^T\bar{x}_i + b\right) - 1\right)\right) = 0$$

$$\Rightarrow \nabla_{\bar{a}}\left(\frac{1}{2}\bar{a}^T\bar{a}\right) - \sum_{i=1}^{2M}\left(\nabla_{\bar{a}}\left(y_i\lambda_i\bar{a}^T\bar{x}_i + \lambda_i y_i b - 1\right)\right) = 0$$

$$\Rightarrow \nabla_{\bar{a}}\left(\frac{1}{2}2\bar{a}\right) - \sum_{i=1}^{2M}\left(\nabla_{\bar{a}}\left(y_i\lambda_i\bar{a}^T\bar{x}_i\right) + \nabla_{\bar{a}}\left(\lambda_i y_i b\right) - \nabla_{\bar{a}}(1)\right) = 0$$

$$\Rightarrow \bar{a} - \sum_{i=1}^{2M}\lambda_i y_i \bar{x}_i = 0$$

$$\Rightarrow \boxed{\bar{a} = \sum_{i=1}^{2M}\lambda_i y_i \bar{x}_i}$$

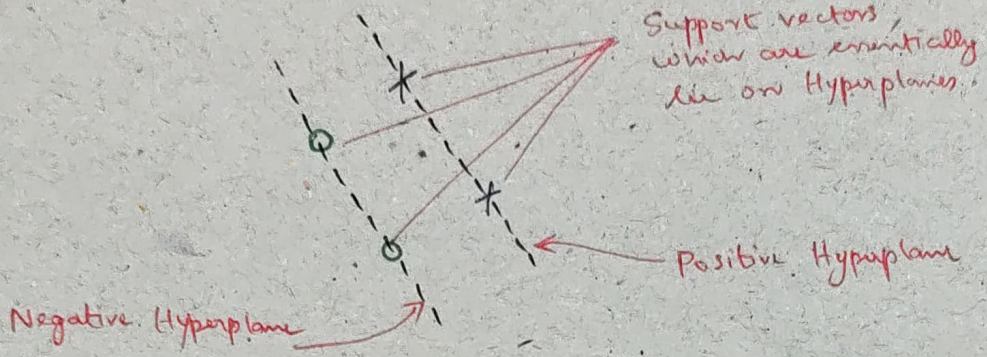
$$\bar{a} = \sum_{i=1}^{2M} \lambda_i y_i \bar{x}_i$$

Linear combination of support vectors, since $a_i \neq 0$.

$$\Rightarrow \bar{a} = \sum_{i=1}^{2M} \alpha_i \bar{x}_i \quad , \text{ where } \alpha_i = \lambda_i y_i$$

Thus, $\bar{a}$ can be expressed as Linear combination of $\bar{x}_i$.

The points for which $\lambda_i \neq 0$ are termed as "Support Vectors"




---

Now, we take Gradient w.r.t. b, and set to 0.

$$(ii)\ \nabla_b \left( \frac{1}{2} \bar{a}^T \bar{a} - \sum_{i=1}^{2M} \lambda_i \left( y_i (\bar{a}^T \bar{x}_i + b) - 1 \right) \right) = 0$$

$$\Rightarrow \nabla_b \left( \frac{1}{2} \bar{a}^T \bar{a} \right)^{0} - \sum_{i=1}^{2M} \left( \nabla_b (\lambda_i y_i \bar{a}^T \bar{x}_i)^{0} + \nabla_b (\lambda_i y_i b) - \nabla_b (i)^{0} \right) = 0$$

$$\Rightarrow \boxed{\sum_{i=1}^{2M} \lambda_i y_i = 0}$$

---

Now, Substitute the expression for $\bar{a}$ in the Lagrangian.

We have, the Lagrangian function,

$$\frac{1}{2} \bar{a}^T \bar{a} - \sum_{i=1}^{2M} \lambda_i \left( y_i (\bar{a}^T \bar{x}_i + b) - 1 \right)$$

$$\Rightarrow \frac{1}{2} \left( \sum_i \lambda_i y_i \bar{x}_i \right)^T \left( \sum_i \lambda_i y_i \bar{x}_i \right) - \sum_i \lambda_i \left( y_i \left[ \left( \sum_i \lambda_i y_i \bar{x}_i \right)^T \bar{x}_i + b \right] - 1 \right)$$

MESSY!

$$\Rightarrow \sum_i \lambda_i - \frac{1}{2} \sum_{i,j=1}^{2M} \lambda_i \lambda_j y_i y_j \bar{x}_i^T \bar{x}_j - b \underbrace{\sum_i \lambda_i y_i}_{0}$$

$$\Rightarrow \boxed{\sum_i \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \bar{x}_i^T \bar{x}_j}$$

This is the Dual objective function, which have to be maximized to formulate Dual Problem.

Remember:

For the Primal problem, we did minimization. Now, For the Dual problem, we have to do maximization.

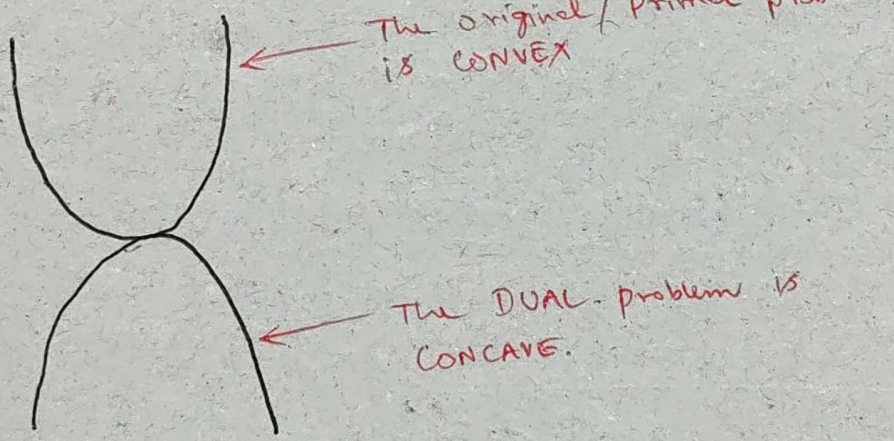
Therefore, the Dual problem can be formulated as

$$\max \sum_{i=1}^{2M} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{2M} \lambda_i \lambda_j y_i y_j \bar{x}_i^T \bar{x}_j$$

subject to constraints

- $\lambda_i \geq 0$ (ii) Lagrange Multipliers cannot be Negative.
- $\sum_{i=1}^{2M} \lambda_i y_i = 0$.

This is the DUAL PROBLEM for SVM, which is equivalent to the Primal problem.




Both have the same optimal solution.

$\Rightarrow$ Duality Gap = 0.

Week 7 : Session 3.

How to calculate b ?

We have, the inequality

$$y_i (\bar{a}^T \bar{x}_i + b) \geq 1$$

When $\lambda_i \neq 0$, (or) $\lambda_i$ is strictly greater than 0, this inequality becomes Equality.

$$\Rightarrow y_i (\bar{a}^T \bar{x}_i + b) = 0$$

This is known as COMPLEMENTARY SLACKNESS property.

By solving this, one can determine b.

Note that, the quantity $\bar{x}_i^T \bar{x}_j$ denotes the Inner Product, which can be represented as $\langle \bar{x}_i, \bar{x}_j \rangle$.

Using this notation, the Dual SVM problem can be defined as

$$\max \sum_{i=1}^{2M} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{2M} \lambda_i \lambda_j y_i y_j \langle \bar{x}_i, \bar{x}_j \rangle$$

The Dual SVM problem depends only on the inner product.

subject to constraints

- $\lambda_i \geq 0$
- $\sum_{i=1}^{2M} \lambda_i y_i = 0$

For any new point $\bar{x}$, y can be calculated as $\bar{a}^T \bar{x} + b$.

$$\Rightarrow \left( \sum_i \lambda_i y_i \bar{x}_i \right)^T \bar{x} + b \qquad \left( \because \bar{a} = \sum_i \lambda_i y_i \bar{x}_i \right)$$

$$\Rightarrow \sum_i \lambda_i y_i \langle \bar{x}_i, \bar{x} \rangle + b$$

The output also depends only on the inner product.

---

One can now replace $\langle \bar{x}_i, \bar{x}_j \rangle$ by a Kernel.

$$K(\bar{x}_i, \bar{x}_j) = \phi^T(\bar{x}_i)\, \phi(\bar{x}_j).$$

$$= (\bar{x}_i^T \bar{x}_j)^2$$

Previously, we were restricted to the linear operation.

Now, using a Kernel, we can go into a non-linear operation..

(ii) We can explore higher order powers of the vectors.

$\bar{x}_i^T \cdot \bar{x}_j \longrightarrow$ Linear

$\phi \longrightarrow$ Non-linear

The SVM that is obtained using a Kernel is called as Kernel SVM.

The quantity $\phi(\bar{x}_i)$ is termed as a Feature Mapping.

This can be used to model non-linear features.

Using this notation, the Kernel SVM problem can be defined as

$$\max \sum_{i=1}^{2M} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{2M} \lambda_i \lambda_j y_i y_j K(\bar{x}_i, \bar{x}_j)$$

subject to constraints

- $\lambda_i \geq 0$
- $\sum_{i=1}^{2M} \lambda_i y_i = 0$.

Kernel SVM use non-linear feature mapping to realize non-linear features. So, it is basically separation in a non-linear space, which can be more efficient and can yield higher accuracy of class picture.

For example, for $N = 3$, (ii) $\bar{x}_i = \begin{bmatrix} x_i(1) \\ x_i(2) \\ x_i(3) \end{bmatrix}$,

The non-linear feature map, $\phi(\bar{x}_i) = \begin{bmatrix} x_i(1)\ x_i(1) \\ x_i(1)\ x_i(2) \\ x_i(1)\ x_i(3) \\ x_i(2)\ x_i(1) \\ x_i(2)\ x_i(2) \\ x_i(2)\ x_i(3) \\ x_i(3)\ x_i(1) \\ x_i(3)\ x_i(2) \\ x_i(3)\ x_i(3) \end{bmatrix}$

For this kernel, we need to know only kernel! Don't even need to know the feature mapping.

Another interesting kernel is the Gaussian kernel defined as,

$$K(\bar{x}_i, \bar{x}_j) = \exp\left(-\frac{\|\bar{x}_i - \bar{x}_j\|^2}{2\sigma^2}\right)$$

Here, we can directly evaluate the kernel!

Advantage of Gaussian Kernel.

Gaussian kernel helps realize an infinite dimensional non-linear feature vector, which comprises of all non-linear (ie) all possible powers of feature vector. So, it maps it to a highly non-linear space, where it is much easier to do classification.

Example. Handwritten digit recognition, from 16 x 16 images.

2715 $\Rightarrow$ 2715

Gaussian kernel SVMs yield very good performance!

---

The Dual SVM is exploited to develop Kernel SVM, which replaces the inner product with a more general kernel (inner product between the feature maps of the original feature).

---