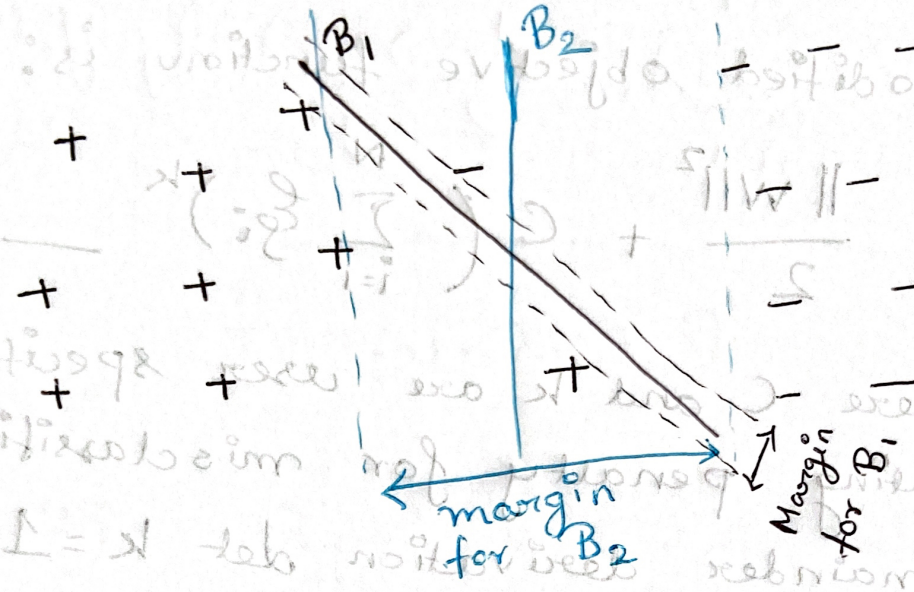
Linear SVM : Non-Separable case.



As it is evident from the above figure, B_2 is a better classifier than B_1 even when it makes small training errors. We thus work towards maximizing this margin, that is tolerable to small training errors.

Our inequality constraints involves addition of slack variables.

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

$$w \cdot x_i + b \leq -1 + \xi_i \quad \text{if } y_i = -1 \qquad \text{---- (1)}$$

where $\forall i : \xi_i > 0$.

ξ is the slack variable that provides estimate of the training errors.

The modified objective function is:

$$f(W) = \frac{\|W\|^2}{2} + C\left(\sum_{i=1}^{N} \xi_i\right)^k \quad \text{— (2)}$$

where C and k are user specified representing penalty for misclassification. For remainder derivation let $k = 1$.

The Lagrangian constrained optimization problem =>

$$L = \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \lambda_i \{ y_i (W \cdot x_i + b) - 1 + \xi_i \} - \sum_{i=1}^{N} \mu_i \xi_i \quad \text{— (3)}$$

The first two terms are the functions to be minimized, third one represents constraint while last term is for non-negativity requirements of ξ_i's.

KKT conditions:

$$\xi_i \geq 0 \quad \lambda_i \geq 0 \quad \mu_i \geq 0 \quad \text{— (4)}$$

$$\lambda_i \{ y_i (w \cdot x_i + b) - 1 + \xi_i \} = 0 \quad \text{— (5)}$$

$$\mu_i \xi_i = 0 \quad \text{— (6)}$$

$$\frac{\partial L}{\partial w_j} = w_j - \sum_{i=1}^{N} \lambda_i y_i x_{ij} = 0$$

$$w_j = \sum_{i=1}^{N} \lambda_i y_i x_{ij} \quad \text{— (7)}$$

$$\frac{\partial L}{\partial b} = - \sum_{i=1}^{N} \lambda_i y_i = 0 \Rightarrow \sum_{i=1}^{N} \lambda_i y_i = 0 \quad \text{— (8)}$$

$$\frac{\partial L}{\partial \xi_i} = C - \lambda_i - \mu_i = 0$$

$$\Rightarrow \lambda_i + \mu_i = C \quad \text{— (9)}$$

Substituting (7), (8), (9) in (3).

$$L_D = \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i \cdot x_j + C \sum_i \xi_i - \sum \lambda_i \{ y_i (\sum_j \lambda_j y_j x_i \cdot x_j + b) - 1 + \xi_i \} - \sum_i (C - \lambda_i) \xi_i$$

Thus, final dual Lagrangian:

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i \cdot x_j$$

Margin for primal and dual problem :

The margin we discussed in the above derivation is **soft margin** as we allow small training errors shown by the slack variables. The **hard margin** on the other hand does not allow any misclassification. In case of the **primal problem** the width of the margin is represented by $\frac{2}{|W|}$ is maximised using the objective function $\frac{1}{2}W^TW + C\sum \zeta_i$ which must be minimized. The penalty parameter C represents the trade-off between margin size and the number of misclassified points. For example, taking C = ∞ requires that all point be correctly classified (this may be infeasible). On the other hand, taking C = 0, tailors to focus on maximizing the margin. C is thus usually chosen using cross-validation to minimize estimated generalization error.

In case of the **dual problem** , the margin that we need to maximize is :
$max(\sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j x_i x_j)$ with respect to λ following the constraints :

$$\sum_{i=1}^{N}\lambda_i y_i = 0 \ , \lambda_i >= 0$$

Benefits of Maximizing the margin:

- In cases discussed above (non-separable linear) we prefer maximizing margin even if that tolerates small training errors because it prevents overfitting.
- A larger margin decision boundary will help more in generalization when predicting samples outside of training dataset.
- The concept is influenced from Bias-Variance tradeoff.

Characterize Support Vectors:

There can 3 scenarios for the training points:

1. $(\lambda_i = 0 \ and \ \zeta_i = 0)$ implies that the data points x_i is correctly classified.
2. When $\zeta_i = 0$ and $y_i(w^Tx_i + b) = 1$ then it implies that x_i is a support vector. The support vectors that satisfy $0 <= \lambda_i < C$ are ***unbounded or free support*** vectors.
3. When $\zeta_i >= 0$ and $y_i(w^Tx_i + b) = 1 - \zeta_i$ then it also implies that x_i is a support vector. These support vectors with $\lambda_i = C$ are called ***bounded vectors*** as they lie inside the margin. They can be further broken down into 2 types => with $0 <= \zeta_i < 1$ means they are correctly classified and $\zeta_i >= 1$ means they are misclassified.

Benefits of Solving Dual rather than Primal Problem:

- Application of kernels to dual problems in SVM is much easier as it does not include mapping each point separately to the higher dimensional plane. Kernel search an optimal separating hyperplane in a higher dimensional space without increasing the computational complexity much. Kernels can simply be applied to algorithms that takes the features in terms of its inner product $x_i^T x_j$. This is exactly what dual problem depends on. We thus classify non-linearly separable data quite easily, with little extra computational effort.

- Most of the dual variables are zero because according to the KKT conditions which states $\lambda_i(y_i(w \,.\, x_i + b) - 1 + \zeta_i) = 0$ for all points not lying on support vectors must have λ to be zero. This helps in solving dual problems.

- Algorithms like Sequential Minimal Optimization that solves the dual problem efficiently.

References:

- Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2005. Introduction to Data Mining, (First Edition). Addison-Wesley Longman Publishing Co., Inc., USA.

- https://www.stat.berkeley.edu/~arturof/Teaching/EE127/Notes/support_vector_machines.pdf

- http://www.cs.umd.edu/class/spring2017/cmsc422/slides0101/lecture20.pdf

③ Multi Class SVM

The derivation is inspired from Crammer and Singer.

Let us assume there are k - classes and l training examples.

Then the primal problem is :

$$\min_{w_m, \xi_i} \frac{1}{2} \sum_{m=1}^{k} w_m^T w_m + C \sum_{i=1}^{l} \xi_i$$

$$w_{y_i}^T \phi(x_i) - w_m^T \phi(x_i) \geq e_i^m - \xi_i \qquad i = 1, \ldots l$$

where $e_i^m \equiv 1 - \delta_{y_i, m}$,

$$\delta_{y_i, m} \equiv \begin{cases} 1 & \text{when } y_i = m \\ 0 & \text{when } y_i \neq m \end{cases}$$

Thus decision function is

$$\operatorname{argmax}_{m=1, \ldots k} \; w_m^T \phi(x)$$

The dual problem is :

$$\min_{\alpha} f(\alpha) = \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} K_{i,j} \bar{\alpha}_i^T \bar{\alpha}_j + \sum_{i=1}^{l} \bar{\alpha}_i^T \bar{e}_i$$

$$\sum_{m=1}^{k} \alpha_i^m = 0, \quad i = 1, \ldots, l$$

$$\alpha_i^m \leq 0 \quad \text{if } y_i \neq m$$

$$\alpha_i^m \leq C \quad \text{if } y_i = m$$

$$i = 1, \ldots, l \qquad m = 1, \ldots k$$

where $K_{i,j} \equiv \phi(x_i)^T \phi(x_j)$

$$\bar{\alpha}_i \equiv [\alpha_i^1, \ldots, \alpha_i^k]^T$$

$$\bar{e}_i \equiv [e_i^1, \ldots, e_i^k]^T$$

Then,

$$w_m = \sum_{i=1}^{l} \alpha_i^m \phi(x_i)$$

If we write,

$$\alpha \equiv [\alpha_1^1, \ldots \alpha_1^k, \ldots, \alpha_l^1, \ldots \alpha_l^k]^T$$

$$e \equiv [e_1^1, \ldots e_1^k, \ldots, e_l^1 \ldots e_l^k]^T$$

then the dual objective can be written as:

$$\frac{1}{2} \alpha^T (K \otimes I) \alpha + e^T \alpha$$

where I is k by k identity matrix $\otimes$ is the Kronecker product.

As K is positive semi-definite, $K \otimes I$, the Hessian of dual objective function is also positive semi-definite.

Thus the decision function is:

$$\underset{m=1,\ldots,k}{\operatorname{argmax}} \sum_{i=1}^{l} \alpha_i^m K(x_i, x).$$

Reference:

① A Comparison of Methods for Multi-class SVM by Chih-Wei Hsu and Chih-Jen Lin.

② On the algorithmic implementation of multiclass kernel-based vector machines by Koby Crammer and Yorum Singer.