Introduction to Support Vector Machines

Support Vector Machines (SVM)

Linear separation of a feature space

This is a variation of the perceptron learning algorithm. Consider a hyper plane in an n-dimensional (n-D) feature space:

$$f(X) = X^T W + b = \sum_{i=1}^{n} x_i w_i + b = 0$$

where the weight vector W is normal to the plane, and $b/|W|$ is the distance from the origin to the plane. The n-D space is partitioned by the plane into two regions. We further define a mapping function $y = sign(f(X)) \in \{1, -1\}$, i.e.,

$$f(X) = X^T W + b = \begin{cases} > 0, & y = sign(f(X)) = 1,\ X \in P \\ < 0, & y = sign(f(X)) = -1,\ X \in N \end{cases}$$

Any point $X \in P$ on the positive side of the plane is mapped to 1, while any point $X \in N$ on the negative side is mapped to -1. A point X of unknown class will be classified to P if $f(X) > 0$, or N if $f(X) < 0$.

The learning problem

Given a set m training samples from two linearly separable classes P and N:

$$\{(X_i, y_i), i = 1, \cdots, m\}$$

where $y_i \in \{1, -1\}$ labels X_i to belong to either of the two classes. we want to find a hyper-plane in terms of W and b, that linearly separates the two classes.

Before W is properly trained, the actual output $y' = sign(f(X))$ may not be the same as the desired output y. There are four possible cases:

	Input (X,y)	Output $y' = sign(f(X))$	result
1	$(X, y = 1)$	$y' = 1 = y$	corrrect
2	$(X, y = -1)$	$y' = 1 \neq y$	incorrect
3	$(X, y = 1)$	$y' = -1 \neq y$	incorrect
4	$(X, y = -1)$	$y' = -1 = y$	corrrect

The weight vector W is updated whenever the result is incorrect (mistake driven):

- If $(X, y = -1)$ but $y' = 1 \neq y$ (case 2 above), then

$$W^{new} = W^{old} + \eta y X = W^{old} - \eta X$$

  When the same X is presented again, we have

$$f(X) = X^T W^{new} + b = X^T W^{old} - \eta X^T X + b < X^T W^{old} + b$$

  The output $y' = sign(f(X))$ is more likely to be $y = -1$ as desired. Here $0 < \eta < 1$ is the learning rate.

- If $(X, y = 1)$ but $y' = -1 \neq y$ (case 3 above), then

$$W^{new} = W^{old} + \eta y X = W^{old} + \eta X$$

  When the same X is presented again, we have

$$f(X) = X^T W^{new} + b = X^T W^{old} + \eta X^T X + b > X^T W^{old} + b$$

  The output $y' = sign(f(X))$ is more likely to be $y = 1$ as desired.

Summarizing the two cases, we get the learning law:

$$\text{if } \; y f(X) = y(X^T W^{old} + b) < 0, \;\; \text{then } \; W^{new} = W^{old} + \eta y X$$

The two correct cases can also be summarized as

$$y f(X) = y(X^T W + b) \geq 0$$

which is the condition a successful classifier should satisfy.

We assume initially $W = 0$, and the m training samples are presented repeatedly, the learning law during training will yield eventually:

$$W = \sum_{j=1}^{m} \alpha_j y_j X_j$$

where $\alpha_i > 0$. Note that W is expressed as a linear combination of the training samples. After receiving a new sample (X_i, y_i), vector W is updated by

$$\text{if } \; y_i f(X_i) = y_i(X_i^T W^{old} + b) = y_i(\sum_{j=1}^{m} \alpha_j y_j (X_i^T X_j) + b) < 0,$$

$$\text{then } \; W^{new} = W^{old} + \eta y_i X_i = \sum_{j=1}^{m} \alpha_j y_j X_j + \eta y_i X_i, \quad \text{i.e. } \; \alpha_i^{new} = \alpha_i^{old} + \eta$$

Now both the decision function

$$f(X) = X^T W + b = \sum_{j=1}^{m} \alpha_j y_j (X^T X_j) + b$$

and the learning law

$$\text{if } \; y_i(\sum_{j=1}^{m} \alpha_j y_j (X_i^T X_j) + b) < 0, \quad \text{then } \; \alpha_i^{new} = \alpha_i^{old} + \eta$$

are expressed in terms of the inner production of input vectors.

Support Vector Machine

For a decision hyper-plane $X^T W + b = 0$ to separate the two classes P $(X_i, 1)$ and N $(X_i, -1)$, it has to satisfy

$$y_i(X_i^T W + b) \geq 0$$

for both $X_i \in P$ and $X_i \in N$. Among all such planes satisfying this condition, we want to find the optimal one that separates the two classes with the maximal margin (the distance between the decision plane and the closest sample points).

The optimal plane should be in the middle of the two classes, so that the distance from the plane to the closest point on either side is the same. We define two additional planes H_+ and H_- that are parallel to H_0 and go through the point closest to the plane on either side:

$$X^TW + b = 1, \quad \text{and} \quad X^TW + b = -1$$

All points $X_i \in P$ on the positive side should satisfy

$$X_i^TW + b \geq 1, \quad y_i = 1$$

and all points $X_i \in N$ on the negative side should satisfy

$$X_i^TW + b \leq -1, \quad y_i = -1$$

These can be combined into one inequality:

$$y_i(X_i^TW + b) \geq 1, \quad (i = 1, \cdots, m)$$

The equality holds for those points on the planes H_+ or H_-. Such points are called *support vectors*, for which

$$X_i^TW + b = y_i$$

i.e., the following holds for all support vectors:

$$b = y_i - X_i^TW = y_i - \sum_{j=1}^{m} \alpha_j y_j (X_i^T X_j)$$

Moreover, the distances from the origin to the three planes H_-, H_0 and H_+ are, respectively, $|b-1|/||w||$, $|b|/||w||$, and $|b+1|/||w||$, and the distances between planes H_- and H_+ is $2/||W||$, which is to be maximized. Now the problem of finding the optimal decision plane in terms of W and b can be formulated as:

$$\begin{array}{rl} \text{minimize} & \frac{1}{2}W^TW = \frac{1}{2}||W||^2 \qquad \text{(objective function)} \\ \text{subject to} & y_i(X_i^TW + b) \geq 1, \;\; \text{or} \;\; 1 - y_i(X_i^TW + b) \leq 0, \quad (i = 1, \cdots, m) \end{array}$$

Since the objective function is quadratic, this constrained optimization problem is called a quadratic program (QP) problem. (If the objective function is linear, the problem is a linear program (LP) problem). This QP problem can be solved by Lagrange multipliers method to minimize

$$L_p(W, b, \alpha) = \frac{1}{2}||W||^2 + \sum_{i=1}^{m} \alpha_i (1 - y_i(X_i^T W + b))$$

with respect to W, b and the Lagrange coefficients $\alpha_i \geq 0$ $(i = 1, \cdots, \alpha_m)$. We let

$$\frac{\partial}{\partial W} L_p(W, b) = 0, \quad \frac{\partial}{\partial b} L_p(W, b) = 0$$

These lead, respectively, to

$$W = \sum_{j=1}^{m} \alpha_j y_j X_j, \quad \text{and} \quad \sum_{i=1}^{m} \alpha_i y_i = 0$$

Substituting these two equations back into the expression of $L(W, b)$, we get the *dual problem* (with respect to α_i) of the above *primal problem*:

$$\begin{aligned} \text{maximize} \quad & L_d(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j X_i^T, X_j \\ \text{subject to} \quad & \alpha_i \geq 0, \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned}$$

The dual problem is related to the primal problem by:

$$L_d(\alpha) = inf_{(W,b)} L_p(W, b, \alpha)$$

i.e., L_d is the greatest lower bound (infimum) of L_p for all W and b.

Solving this dual problem (an easier problem than the primal one), we get α_i, from which W of the optimal plane can be found.

Those points X_i on either of the two planes H_+ and H_- (for which the equality $y_i(W^T X_i + b) = 1$ holds) are called *support vectors* and they correspond to positive Lagrange multipliers $\alpha_i > 0$. The training depends only on the support vectors, while all other samples away from the planes H_+ and H_- are not important.

For a support vector X_i (on the H_- or H_+ plane), the constraining condition is

$$y_i(X_i^T W + b) = 1 \quad (i \in sv)$$

here sv is a set of all indices of support vectors X_i (corresponding to $\alpha_i > 0$. Substituting

$$W = \sum_{j=1}^{m} \alpha_j y_j X_j = \sum_{j \in sv} \alpha_j y_j X_j$$

we get

$$y_i(\sum_{j \in sv} \alpha_j y_j X_i^T X_j + b) = 1$$

Note that the summation only contains terms corresponding to those support vectors X_j with $\alpha_j > 0$, i.e.

$$y_i \sum_{j \in sv} \alpha_j y_j X_i^T X_j = 1 - y_i b$$

For the optimal weight vector W and optimal b, we have:

$$\begin{aligned} ||W||^2 &= W^T W = \sum_{i \in sv} \alpha_i y_i X_i^T \sum_{j \in sv} \alpha_j y_j X_j = \sum_{i \in sv} \alpha_i y_i \sum_{j \in sv} \alpha_j y_j X_i^T X_j \\ &= \sum_{i \in sv} \alpha_i (1 - y_i b) = \sum_{i \in sv} \alpha_i - b \sum_{i \in sv} \alpha_i y_i \\ &= \sum_{i \in sv} \alpha_i \end{aligned}$$

The last equality is due to $\sum_{i=1}^m \alpha_i y_i = 0$ shown above. Recall that the distance between the two margin planes H_+ and H_- is $2/||W||$, and the margin, the distance between H_+ (or H_-) and the optimal decision plane H_0, is

$$1/||W|| = (\sum_{i \in sv} \alpha_i)^{-1/2}$$

Soft Margin SVM

When the two classes are not linearly separable (e.g., due to noise), the condition for the optimal hyper-plane can be relaxed by including an extra term:

$$y_i(X_i^T W + b) \geq 1 - \xi_i, \quad (i = 1, \cdots, m)$$

For minimum error, $\xi_i \geq 0$ should be minimized as well as $||W||$, and the objective function becomes:

$$\begin{aligned} &\text{minimize} \quad W^T W + C \sum_{i=1}^{m} \xi_i^k \\ &\text{subject to} \quad y_i(X_i^T W + b) \geq 1 - \xi_i, \quad \text{and} \quad \xi_i \geq 0; \quad (i = 1, \cdots, m) \end{aligned}$$

Here C is a regularization parameter that controls the trade-off between maximizing the margin and minimizing the training error. Small C tends to emphasize the margin while ignoring the outliers in the training data, while large C may tend to overfit the training data.

When $k = 2$, it is called 2-norm soft margin problem:

$$\begin{aligned} \text{minimize} \quad & W^T W + C \sum_{i=1}^{m} \xi_i^2 \\ \text{subject to} \quad & y_i(X_i^T W + b) \geq 1 - \xi_i, \quad (i = 1, \cdots, m) \end{aligned} \tag{1}$$

Note that the condition $\xi_i \geq 0$ is dropped, as if $\xi_i < 0$, we can set it to zero and the objective function is further reduced.) Alternatively, if we let $k = 1$, the problem can be formulated as

$$\begin{aligned} \text{minimize} \quad & W^T W + C \sum_{i=1}^{m} \xi_i \\ \text{subject to} \quad & y_i(X_i^T W + b) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0; \quad (i = 1, \cdots, m) \end{aligned} \tag{2}$$

This is called 1-norm soft margin problem. The algorithm based on 1-norm setup, when compared to 2-norm algorithm, is less sensitive to outliers in training data. When the data is noisy, 1-norm method should be used to ignore the outliers.

2-Norm Soft Margin

The primal Lagrangian for 2-norm problem above is

$$L_p(W, b, \xi, \alpha) = \frac{1}{2} W^T W + \frac{C}{2} \sum_{i=1}^{m} \xi_i^2 - \sum_{i=1}^{m} \alpha_i [y_i(W^T X + b) - 1 + \xi_i]$$

Substituting

$$\frac{\partial L}{\partial W} = W - \sum_{i=1}^{m} y_i \alpha_i X_i = 0; \quad \frac{\partial L}{\partial \xi} = C\xi - \alpha = 0; \quad \frac{\partial L}{\partial b} = \sum_{i=1}^{m} y_i \alpha_i = 0$$

into the primal Lagrangian, we get the dual problem

$$\begin{aligned}
\text{maximize} \quad L_d(\alpha) &= \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m y_i y_j \alpha_i \alpha_j X_j^T X_i - \frac{1}{2C}\sum_{i=1}^m \alpha_i^2 \\
&= \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m y_i y_j \alpha_i \alpha_j (X_j^T X_i + \frac{1}{C}\delta_{ij}) \\
\text{subject to} \quad \alpha_i \geq 0, &\quad \sum_{i=1}^m \alpha_i y_i = 0
\end{aligned}$$

This QP program can be solved for α_i. All support vectors X_i corresponding to $\alpha_i > 0$ satisfy:

$$y_i(X_i^T W + b) = 1 - \xi_i$$

Substituting $W = \sum_{j\in sv} y_j \alpha_j X_j$ into this equation, we get

$$y_i(\sum_{j\in sv} y_j \alpha_j (X_i^T X_j) + b) = 1 - \xi_i, \quad \text{i.e.,} \quad y_i \sum_{j\in sv} y_j \alpha_j (X_i^T X_j) = 1 - \xi_i - y_i b$$

For the optimal weight W, we have

$$\begin{aligned}
||W||^2 &= W^T W = \sum_{i\in sv} \alpha_i y_i X_i^T \sum_{j\in sv} \alpha_j y_j X_j = \sum_{i\in sv} \alpha_i y_i \sum_{j\in sv} \alpha_j y_j X_i^T X_j \\
&= \sum_{i\in sv} \alpha_i (1 - \xi_i - y_i b) = \sum_{i\in sv} \alpha_i - \sum_{i\in sv} \alpha_i \xi_i - b \sum_{i\in sv} y_i \alpha_i \\
&= \sum_{i\in sv} \alpha_i - \sum_{i\in sv} \alpha_i \xi_i = \sum_{i\in sv} \alpha_i - \frac{1}{C}\sum_{i\in sv} \alpha_i^2
\end{aligned}$$

The last equation is due to $\xi_i = \alpha_i / C$. The optimal margin is

$$1/||W|| = (\sum_{i\in sv} \alpha_i - \frac{1}{C}\sum_{i\in sv} \alpha_i^2)^{-1/2}$$

1-Norm Soft Margin

The primal Lagrangian for 1-norm problem above is

$$L_p(W, b, \xi, \alpha, \gamma) = \frac{1}{2} W^T W + C\sum_{i=1}^m \xi_i - \sum_{i=1}^m \alpha_i [y_i(W^T X + b) - 1 + \xi_i] - \sum_{i=1}^m \gamma_i \xi_i$$

with $\alpha_i \geq 0$ and $\gamma_i \geq 0$. Substituting

$$\frac{\partial L}{\partial W} = W - \sum_{i=1}^{m} y_i \alpha_i X_i = 0; \quad \frac{\partial L}{\partial \xi} = C - \alpha_i - \gamma_i = 0; \quad \frac{\partial L}{\partial b} = \sum_{i=1}^{m} y_i \alpha_i = 0$$

into the primal Lagrangian, we get the dual problem

$$\begin{aligned} \text{maximize} \quad L_d(\alpha, \gamma) &= \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j X_j^T X_i - \sum_{i=1}^{m} \alpha_i \xi_i - \sum_{i=1}^{m} \gamma_i \xi_i + C \sum_{i=1}^{m} \xi_i \\ &= \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j X_j^T X_i \\ \text{subject to} \quad 0 \leq \alpha_i \leq C, &\quad \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned}$$

Note that interestingly the objective function of the dual problem is identical to that of the linearly separable problem discussed previously, due to the nice cancellation based on $C = \alpha_i + \gamma_i$. Also, since $\alpha_i \geq 0$ and $\gamma_i \geq 0$, we have $0 \leq \alpha_i \leq C$. Solving this QP problem for α_i, we get the optimal decision plane W and b with the margin

$$\left(\sum_{i \in sv} \sum_{j \in sv} \alpha_i \alpha_j y_i y_j X_i^T X_j\right)^{-1/2}$$

Kernel Mapping

The algorithm above converges only for linearly separable data. If the data set is not linearly separable, we can map the the samples X into a feature space of higher dimensions:

$$X \longrightarrow \phi(X)$$

in which the classes can be linearly separated. The decision function in the new space becomes:

$$f(X) = \phi(X)^T W + b = \sum_{j=1}^{m} \alpha_j y_j (\phi(X)^T \phi(X_j)) + b$$

where

$$W = \sum_{j=1}^{m} \alpha_j y_j \phi(X_j)$$

and b are the parameters of the decision plane in the new space. As the vectors X_i appear only in inner products in both the decision function and the learning law, the mapping function $\phi(X)$ does not need to be explicitly specified. Instead, all we need is the inner product of the vectors in the new space. The function $\phi(X)$ is a kernel-induced *implicit* mapping.

Definition: A kernel is a function that takes two vectors X_i and X_j as arguments and returns the value of the inner product of their images $\phi(X_i)$ and $\phi(X_j)$:

$$K(X_1, X_2) = \phi(X_1)^T \phi(X_2)$$

As only the inner product of the two vectors in the new space is returned, the dimensionality of the new space is not important.

The learning algorithm in the kernel space can be obtained by replacing all inner products in the learning algorithm in the original space with the kernels:

$$f(X) = \phi(X)^T W + b = \sum_{j=1}^{m} \alpha_j y_j K(X, X_j) + b$$

The parameter b can be found from any support vectors X_i:

$$b = y_i - \phi(X_i)^T W = y_i - \sum_{j=1}^{m} \alpha_j y_j (\phi(X_i)^T \phi(X_j)) = y_i - \sum_{j=1}^{m} \alpha_j y_j K(X_i, X_j)$$

Example 0: linear kernel

Assume $X = (x_1, \cdots, x_n)^T$, $Z = (z_1, \cdots, z_n)^T$,

$$K(X, Z) = X^T Z = \sum_{i=1}^{n} x_1 z_1$$

Example 1: polynomial kernels

Assume $X = (x_1, x_2)^T$, $Z = (z_1, z_2)^T$,

$$\begin{aligned} K(X, Z) = (X^T Z)^2 &= (x_1 z_1 + x_2 z_2)^2 = x_1^2 z_1^2 + x_2^2 z_2^2 + 2 x_1 z_1 x_2 z_2 \\ &= < (x_1^2, x_2^2, \sqrt{2} x_1 x_2), (z_1^2, z_2^2, \sqrt{2} z_1 z_2) >= \phi(X)^T \phi(Z) \end{aligned}$$

This is a mapping from a 2-D space to a 3-D space. The order can be changed from 2 to general d.

Example 2:

$$K(X, Z) = e^{-||X-Z||^2/2\sigma^2}$$

Example 3:

$$K(X, Z) = K(X, Z) K(X, X)^{-1/2} K(Z, Z)^{-1/2}$$