MODULE 13
Support Vector Machines

LESSON 29
Training Support Vector Machines

Keywords: Maximum Margin, Training SVM, Classification, Kernel Trick

Maximum Margin Classification using SVM

- **Margin:**
  Consider the linearly separable problem described towards the end of lesson 28. In such a case, the distance between the two support lines is given by $\frac{2}{||w||}$ where $w = (\frac{1}{2}, -\frac{1}{2})$. So, the margin in this specific example is $2\sqrt{2}$. It is possible to maximize the distance (margin) by minimizing a monotone function of $||w||$. Further, each pattern imposes a constraint on the value of w because for each pattern X from class 'X', we want $w^t X + b \leq -1$ and for every pattern from the positive class (class 'O') we need $w^t X + b \geq 1$. A popular formulation of the optimization problem is

$$Minimize \frac{|| w ||^2}{2} \tag{1}$$

$$such\ that\ w^t X + b \leq -1\ \forall X \in X\ and \tag{2}$$

$$w^t X + b \geq 1\ \forall X \in O \tag{3}$$

We illustrate the solution to the linearly separable problem using an example.

Example 1

Consider a one-dimensional data set of 3 points shown in Table 1.

Pattern No.	x	class
1	1	'X'
2	2	'X'
3	6	'O'

Table 1: SVM on one-dimensional data

Here, the required w is a scalar and so the criterion function to be minimized is $\frac{w^2}{2}$ and the three constraints are (one per pattern):

$w + b \leq -1$, $2w + b \leq -1$, and $6w + b \geq 1$. Such a constrained optimization problem is solved by forming the Lagrangian given by

$$MinimizeJ(w) = \frac{w^2}{2} - \alpha_1(-w - b - 1) - \alpha_2(-2w - b - 1) - \alpha_3(6w + b - 1) \quad (4)$$

where $\alpha_i s$ are the Lagrange variables, one per each constraint. By differentiating with respect to w and b and equating the partial derivatives to 0, we get

$$\frac{\delta J}{\delta w} = 0 \Rightarrow w = -\alpha_1 - 2\alpha_2 + 6\alpha_3 \quad (5)$$

$$\frac{\delta J}{\delta b} = 0 \Rightarrow 0 = \alpha_1 + \alpha_2 - \alpha_3 \quad (6)$$

Similarly, by differentiating with αs we get

$$-w - b - 1 = 0 \quad (7)$$

$$-2w - b - 1 = 0 \quad (8)$$

$$6w + b - 1 = 0 \quad (9)$$

Note that equations (7) and (8) lead to the values of $w = 0$ and b = -1. These values do not satisfy equation (9). So, using (8) and (9), we get $w = \frac{1}{2}$ and b = -2. Also, for the optimal solution, we require either $\alpha_1 = 0$ or $-w - b - 1 = 0$. For the values selected (w = $\frac{1}{2}$ and b = -2), $-w - b - 1 \neq 0$; so, $\alpha_1 = 0$. So, from equation (6), we get $\alpha_2 = \alpha_3$. Now from equation (5), we get $\frac{1}{2} = 4\alpha_3$. So, $\alpha_2 = \alpha_3 = \frac{1}{8}$. Note that the decision boundary is specified by $wx + b = 0$. In this case $w = \frac{1}{2}$ and $b = -2$. So, the decision boundary here is

$$\frac{1}{2}x - 2 = 0 \Rightarrow x - 4 = 0 \; or \; x = 4.$$

Note that $x = 4$ separates the points 1 and 2 (Class 'X') from the point 6 (Class 'O').

Even though the data in the above example is one-dimensional, the solution procedure is general enough. For example, we will see that $w = \sum_{i=1}^{n} \alpha_i y_i X_i$. In the current one-dimensional example,

$$w = 0 * (-1) * 1 + \frac{1}{8} * (-1) * 2 + \frac{1}{8} * (1) * 6 = -\frac{2}{8} + \frac{6}{8} = \frac{1}{2}$$

We illustrate it further with a two-dimensional example.

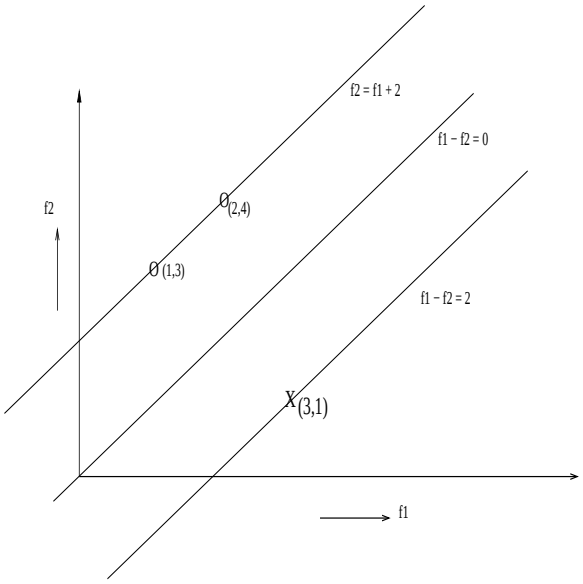

Figure 1: Illustration of Support Vector Machines

Example 2

Consider the three two-dimensional data points shown in Figure 1 where (1,3), and (2,4) are from class 'X' and (3,1) is from 'O'. Here w is a two-dimensional vector and so the objective function and the three constraints are

$$Minimize \ \frac{|| \ w \ ||^2}{2} \tag{10}$$

$$such \ that \ w^t X + b \ \leq \ -1 \ \forall X \in \text{`}X' \ and \tag{11}$$

$$w^t X \ + \ b \ \geq \ 1 \ \forall X \ \in \ \text{`}O' \tag{12}$$

Note that the first constraint is $w^t X + b \leq -1$, where $X = (1,3)^t$. So, we get equivalently $w_1 \ + \ 3w_2 \ + b \ \leq \ -1$ By writing the constraints in terms of the two components of w, namely w_1 and w_2 and the corresponding components of X, we have the Lagrangian to be

$$J(w) = \frac{|| \ w \ ||^2}{2} - \alpha_1(-w_1 - 3w_2 - b - 1) - \alpha_2(-2w_1 - 4w_2 - b - 1)$$
$$-\alpha_3(3w_1 + w_2 + b - 1)$$

Note that by differentiating $J(w)$ with respect to w, we get

$$w + \alpha_1(1,3)^t + \alpha_2(2,4)^t - \alpha_3(3,1)^t = 0 \tag{13}$$

The above equation leads to

$$w_1 = -\alpha_1 - 2\alpha_2 + 3\alpha_3 \tag{14}$$
$$w_2 = -3\alpha_1 - 4\alpha_2 + \alpha_3 \tag{15}$$

By differentiating $J(w)$ with respect to b and equating to 0, we have

$$\alpha_1 + \alpha_2 - \alpha_3 = 0 \tag{16}$$

Also, by differentiating with respect to αs and equating to 0, we get

$$-w_1 - 3w_2 - b - 1 = 0 \tag{17}$$
$$-2w_1 - 4w_2 - b - 1 = 0 \tag{18}$$
$$3w_1 + w_2 + b - 1 = 0 \tag{19}$$

From equations (17) and (19), we get

$$2w_1 - 2w_2 = 2 \;\Rightarrow\; w_1 - w_2 = 1. \tag{20}$$

Similarly, from (18) and (19), we get

$$w_1 - 3w_2 - 2 = 0 \;\Rightarrow\; w_1 - 3w_2 = 2 \tag{21}$$

From (20) and (21), we get $2w_2 = -1 \Rightarrow w_2 = -\frac{1}{2}$. So, from (20), we get $w_1 = \frac{1}{2}$. Now, from (17), we get $b = 0$. So, the solution is $b = 0$ and

$$w = \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix}$$

Note that in this case, all the three vectors are support vectors. It is possible to solve for αs using equations (14), (15), and (16). From (16), we get

$$\alpha_3 = \alpha_1 + \alpha_2 \tag{22}$$

Using this equality in (14) and (15) along with $w_1 = \frac{1}{2}$ and $w_2 = -\frac{1}{2}$, we get

$$2\alpha_1 + \alpha_2 = \frac{1}{2} \tag{23}$$

$$-2\alpha_1 - 3\alpha_2 = -\frac{1}{2} \tag{24}$$

From (23) and (24), we get $\alpha_2 = 0$ and $\alpha_1 = \frac{1}{4}$. Now from (22), we get $\alpha_3 = \alpha_1 = \frac{1}{4}$.

We will be able to show, in general, that

$$w = \sum_i \alpha_i y_i X_i \tag{25}$$

where y_i is +1 for positive samples (patterns from 'O') and is -1 for negative patterns (patterns from 'X'). In this example,

$$w = \frac{1}{4} * (-1) * (1,3)^t + 0 * (-1) * (2,4)^t + \frac{1}{4} * 1 * (3,1)^t = (\frac{1}{2}, -\frac{1}{2})^t.$$

- **Maximizing Margin**

We have observed earlier that the margin is $\frac{2}{||w||}$. So, maximizing margin is equivalent to minimizing $\frac{||w||}{2}$. Instead for the sake of simplicity in calculus, we consider minimizing $\frac{||w||^2}{2}$. Further, we need to satisfy constraints, one per sample, of the form $w^t X + b \geq 1$ for $X \in$ 'X' and $w^t X + b \leq -1$ for $X \in$ 'O'.
The Lagrangian is

$$L(w, b, \alpha) = \frac{|| w ||^2}{2} - \sum_{i=1}^{n} \alpha_i [y_i (w^t X_i + b) - 1] \tag{26}$$

Note that α_is are Lagrange variables and they are non-negative.

We can get the Karush-Kuhn-Tucker (KKT) conditions which are necessary and sufficient for optimizing the convex problem in this case.

These are given by

$$\frac{\partial L(w,b,\alpha)}{\partial w} \;=\; \frac{2w}{2} \;-\; \sum_{i=1}^{n} y_i \alpha_i X_i \;=\; 0 \Rightarrow\; w \;=\; \sum_{i=1}^{n} y_i \alpha_i X_i \qquad (27)$$

Similarly, by differentiating with respect to b, we have

$$\frac{\partial L(w,b,\alpha)}{\partial b} \;=\; \sum_{i=1}^{n} y_i \alpha_i \;=\; 0 \qquad (28)$$

By using the value of w from equation (27) in (26), we get

$$L(w,b,\alpha) \;=\; \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j X_i^t X_j \;-\; \sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j X_i^t X_j \;-\; b\sum_{i=1}^{n} \alpha_i y_i \;+\; \sum_{i=1}^{n} \alpha_i \qquad (29)$$

The third term in the above equation vanishes because $\sum_{i=1}^{n} \alpha_i y_i \;=\; 0$ from equation (28). So, we get

$$L(w,b,\alpha) \;=\; \sum_{i=1}^{n} \alpha_i \;-\; \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j X_i^t X_j \qquad (30)$$

This is the dual function and the dual parameters (αs) are obtained by solving the following dual problem.

$$maximize \;\sum_{i=1}^{n} \alpha_i \;-\; \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j X_i^t X_j \qquad (31)$$

$$such\ that \;\sum_{i=1}^{n} \alpha_i y_i \;=\; 0,\ and\ \alpha_i \;\geq\; 0,\ for\ all\ i \;=\; 1, \cdots, n \qquad (32)$$

An advantage with the dual formulation is that it can be naturally extended to deal with the nonseparable case.
For more details on the KKT conditions and the optimization problem, refer to the tutorial paper by Burges[1].

- **Linearly Nonseparable Case**

In the previous subsection, we have seen a mechanism for dealing with classification of patterns when the classes are linearly separable. However, it is possible that the classes may not be linearly separable in some cases. We illustrate with a simple example.

Example 3

Consider a one-dimensional collection of 4 labelled patterns: -3, and 3 from class 'O' and -1 and 1 from class 'X'. They are not linearly separable. However, if we can map these points using a function of the form $f(x) = x^2$, we get 9 and 9 for patterns in O and 1 and 1 for patterns in 'X' and these modified patterns are linearly separable.

Another possibility is to convert the one-dimensional patterns into two-dimensional patterns is given in the following example.

Example 4

$f(x) = (x, x^2)^t$ maps the points of class 'O' to $(-3, 9)^t$ and $(3, 9)^t$ and points of class 'X' to $(-1, 1)^t$ and $(1, 1)^t$ and these are also linearly separable.

- **Kernel Trick**

We define the dot product between two vectors $(p_1, p_2)^t$ and $(q_1, q_2)^t$ as $p_1q_1 + p_2q_2$. In general, it is possible to map points in a d-dimensional space to some D-dimensional space ($D > d$) to explore the possibility of linear separability. This may be illustrated using the following example.

Example 5

Consider the function f shown in Table 2. If we consider the output values 0 and 1 to represent classes 'X' and 'O' respectively, then these two classes are not linearly separable. This may be shown using a simple argument. Let us consider that, on the contrary, that the two classes are linearly separable. This means that there exists a line of the form $\beta_1 f_1 + \beta_2 f_2 + c = 0$ which separates points of class 'X' from

f_1	f_2	$f(f_1, f_2)$
0	0	1
0	1	0
1	0	0
1	1	1

Table 2: Truth Table for $f(f_1, f_2)$

f_1	f_2	$f_1 \wedge f_2$	$f(f_1, f_2, f_1 \wedge f_2)$
0	0	0	1
0	1	0	0
1	0	0	0
1	1	1	1

Table 3: Truth Table for $f(f_1, f_2, f_1 \wedge f_2)$

those of class 'O'. More specifically, a point (f_1, f_2) labelled 'X' will satisfy the inequality $\beta_1 f_1 + \beta_2 f_2 + c < 0$ and similarly a point (f_1, f_2) labelled '0' will satisfy the inequality $\beta_1 f_1 + \beta_2 f_2 + c > 0$. Note that the first and the fourth rows in the table correspond to class '0' and the remaining two rows (second and the third) correspond to class 'X'. So, we get the following inequalities:
From the first row: $c > 0$
from the fourth row: $\beta_1 + \beta_2 + c > 0$
from the second row: $\beta_2 + c < 0$
and from the third row: $\beta_1 + c < 0$.

These four inequalities lead to a contradiction as by adding the first two inequalities, we get $\beta_1 + \beta_2 + 2c > 0$ and addition of the last two inequalities gives us $\beta_1 + \beta_2 + 2c < 0$ contradicting it. This clearly shows that the two classes are not linearly separable. However, by using the feature $f_1 \wedge f_2$ also, we get a higher dimensional representation (the additional feature makes the 2-dimensional space a 3-dimensional space). The corresponding truth table of f is shown in Table 3. Note that in the three-dimensional space, the classes are linearly separable. In this case, we need to consider a plane that separates points of class

'0' from those of class 'X'. For example, consider $P = (f_1, f_2, f_1 \wedge f_2)^t$ and let $w = (-1, -1, 2)^t$ and $b = 1$, then verify that $w^t P + b = 0$ forms the decision boundary.

In addition, in SVMs it is done such that dot product computations in the D-dimensional space are carried out by using dot product computations in the d-dimensional space. For example, consider mapping from a 2-dimensional space to a 3-dimensional space such that a point $(p_1, p_2)^t$ in the 2-d space is mapped to $(p_1^2, p_2^2, \sqrt{2}p_1p_2)^t$ in the 3-d space. Now the dot product between two points in the 3-d space can be characterized using a function of the dot product of the corresponding vectors in the 2-d space. This may be illustrated as follows.

Let $p = (p_1, p_2)^t$ and $q = (q_1, q_2)^t$ be two points in the two-dimensional space. Let ϕ be a mapping that maps a two-dimensional point to a point in the three-dimensional space so that

$\phi(p) = (p_1^2, p_2^2, \sqrt{2}p_1p_2)^t$ and
$\phi(q) = (q_1^2, q_2^2, \sqrt{2}q_1q_2)^t$ and

We define a kernel function, $K(p, q)$ as
$K(p, q) = \phi(p)^t\phi(q) = \phi(p).\phi(q)$ So,

$$K(p, q) = \phi(p).\phi(q) = (p_1, p_2, \sqrt{2}p_1p_2).(q_1, q_2, \sqrt{2}q_1q_2) \quad (33)$$
$$= p_1q_1 + p_2q_2 + 2p_1p_2q_1q_2 = (p_1q_1 + p_2q_2)^2 = (p.q)^2 \quad (34)$$

Note that we can compute the dot product in the high-dimensional (kernel) space by performing computations in the low-dimensional input (feature) space. For example, in the above computation, K(p,q) can be computed by using $(p.q)^2$. There are necessary and sufficient conditions for a function to be an appropriate kernel function. In the above example, we considered a function ϕ that maps two-dimensional patterns to a three-dimensional space; in theory, it is possible to consider a function that can map to an infinite dimensional space. Integration of such a kernel trick in the SVM optimization problem leads to the following dual problem.

$$Maximize \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_iy_j\alpha_i\alpha_jK(X_i, X_j) \quad (35)$$

$$such\ that\ \sum_{i=1}^{n} y_i\alpha_i\ =\ 0,\ 0 \leq \alpha_i \leq C,\ i\ =\ 1,\ \cdots,\ n \tag{36}$$

Note that, here, instead of using $X_i.X_j$ as in the linear case, we have used the kernel function $K(X_i, X_j)$ which helps in the computation. Also, the values of the Lagrange variables are bounded by a parameter C; this parameter is tuned to permit a proportion of the training patterns to fall on the wrong side of the decision boundary between the two classes. There are efficient software libraries to solve the SVM problem to maximize the margin. However, for most of the large-scale classification problems, linear formulation (that means without the kernel trick) is popularly used for the sake of scalability.

- **Assignment**

1. Consider a binary classifier which assigns pattern X to class 'O' (positive class) if $f(x) > 0$ and to class 'X' (negative class) if $f(X) < 0$, where

$$f(x) \quad = a + bx + cx^2 \tag{37}$$

Observe, based on the discussion above, that equivalently we can assign X to 'O' if $z^t X' > 0$ and to class 'X' if $z^t X' < 0$, where

$$z = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

$$X' = \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix}$$

Let us consider a set of labeled patterns that are not linearly separable. Specifically, let us consider the one-dimensional data set shown in the following table. Normalize and transform the data and learn the weight vector using perceptron learning algorithm.

Pattern No.	x	class
1	1	'O'
2	-1	'O'
3	2	'O'
4	-2	'O'
5	3	'X'
6	4	'X'
7	-3	'X'
8	-4	'X'

2. Consider a two-class two-dimensional dataset given by $Class1 : (1,2)^t,\ (1,1)^t,\ (2,2)^t$ and $Class2 : (2,0)^t,\ (1,-2)^t$.

   (a) Obtain the support vectors.
   (b) What is the Decision Boundary?
   (c) What is the width of the margin?
   (d) What happens to the resulting classifier if we add points $(1,3)^t,\ (2,3)^t$ from Class1 and $(2,-1)^t,\ (1,-3)^t$ from Class2?

3. Consider a one-dimensional data set of 3 points shown in the following table.

Pattern No.	x	class
1	1	'X'
2	2	'X'
3	4	'O'

   (a) What is the Criterion function to be minimized in the case of SVM?
   (b) Identify the constraints associated with the problem.
   (c) What are the corresponding α values?
   (d) Obtain w and b of the SVM in this case.
   (e) What is the corresponding decision boundary?

- **References**

1. **Burges C.J.C**, (1998) A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, 2:121-167.
2. **Cristianini N.;Shawe-Taylor J.** (2000) *An Introduction to Support Vector Machines*, Cambridge University Press, UK.
3. **V. S. Devi and M. N. Murty** (2011) *Pattern Recognition: An Introduction*, Universities Press, Hyderabad.
4. In addition, there are software packages **LIBSVM** and **SVM-Light** which are popularly used to learn the SVM classifier.