Kernels & Support Vector Machines Cont.

November 21, 2019

1 Introduction to Support Vector Machines

- SVMs are Maximum Margin Classifiers
- Two class classification problems: $y(\mathbf{x}) = \mathbf{w}^T\phi(\mathbf{x}) + b$
- We cannot directly compute $\phi(\mathbf{x})$ for all mappings, we want to use a kernel trick. So, we have to write up a dual representation of the problem in terms of K matrices
- We will start with the case where the $\phi(x)$ are linearly separable in the kernel space
- Note: the SVM finds a linear decision boundary in the feature space. But since we can do non-linear transformations to get to the feature space, the descision boundary can be non-linear in the feature space.
- We want to find $\mathbf{w}$ and b so that $y(\mathbf{x}) = \mathbf{w}^T\phi(\mathbf{x}) + b$ is $y(\mathbf{x}_n) > 0$ for $t_n = 1$ and $y(\mathbf{x}_n) < 0$ for $t_n = -1$. Or, in other words, we want:

$$t_n y(\mathbf{x}_n) > 0 \forall n \tag{1}$$

- The SVM finds the particular $\mathbf{w}$ and b that maximizes the margin (the distance between the closest point and the decision boundary for each class)
- Note: The SVM is inherently a classification algorithm. It is based on margin - separating two classes.
- We want to maximize the smallest distance between points from both classes. So we need the form for the distance and we can then plug that into our equation for the linear model

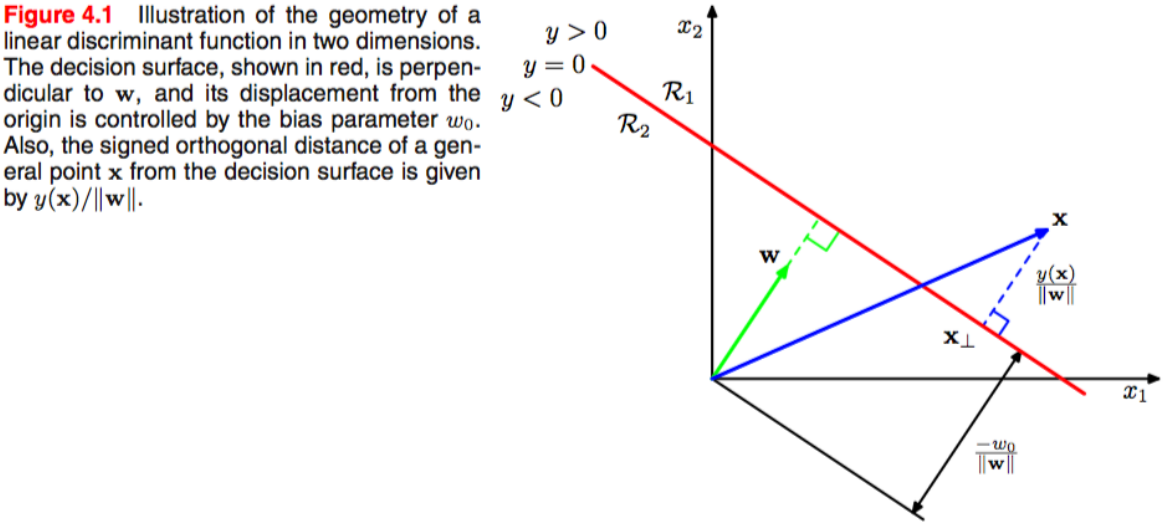


Figure 4.1 Illustration of the geometry of a linear discriminant function in two dimensions. The decision surface, shown in red, is perpendicular to $\mathbf{w}$, and its displacement from the origin is controlled by the bias parameter w_0. Also, the signed orthogonal distance of a general point $\mathbf{x}$ from the decision surface is given by $y(\mathbf{x})/\|\mathbf{w}\|$.

- Let $\mathbf{z}$ be $\phi(\mathbf{x})$

$$
\begin{aligned}
y(\mathbf{z}) &= \mathbf{w}^T\mathbf{z} + b && (2)\\
y(\mathbf{z}) &= \mathbf{w}^T\left(\mathbf{z}_p + r\frac{\mathbf{w}}{\|w\|}\right) + b && (3)\\
y(\mathbf{z}) &= \left(\mathbf{w}^T\mathbf{z}_p + b\right) + r\frac{\mathbf{w}^T\mathbf{w}}{\|w\|} && (4)\\
y(\mathbf{z}) &= 0 + r\frac{\mathbf{w}^T\mathbf{w}}{\|w\|} = r\frac{\|w\|^2}{\|w\|} && (5)\\
y(\mathbf{z}) &= r\,\|w\| && (6)\\
\frac{y(\mathbf{z})}{\|w\|} &= r && (7)
\end{aligned}
$$

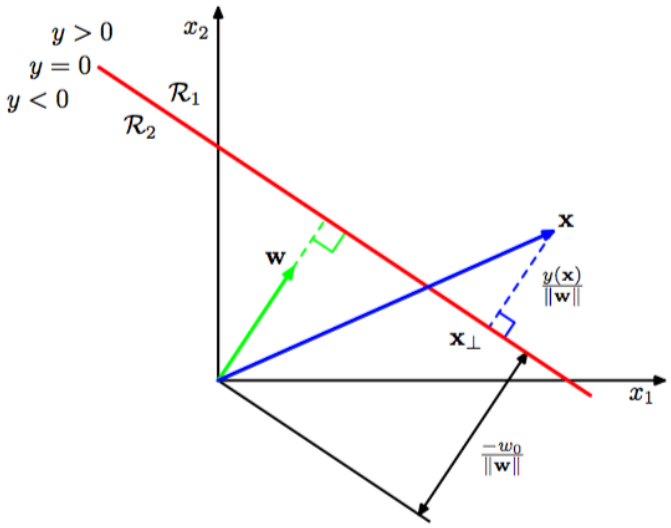
- So, the distance r is $\frac{y(\mathbf{z})}{\|w\|}$

- We want $t_n y(\mathbf{x}_n) > 0 \forall n$:

$$\frac{t_n y(\mathbf{x}_n)}{\|w\|} = \frac{t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b\right)}{\|w\|} > 0 \tag{8}$$

- So, we can define the following objective function:
- Note: the figures below are from Bishop's text.

Figure 4.1 Illustration of the geometry of a linear discriminant function in two dimensions. The decision surface, shown in red, is perpendicular to $\mathbf{w}$, and its displacement from the origin is controlled by the bias parameter w_0. Also, the signed orthogonal distance of a general point $\mathbf{x}$ from the decision surface is given by $y(\mathbf{x})/\|\mathbf{w}\|$.



- So, we can define the following objective function:

$$\arg\max_{w,b} \left\{ \frac{1}{\|w\|} \min_n \left[t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b \right) \right] \right\} \tag{9}$$

- We can rewrite this as a constraint. Find the $\mathbf{w}$ such that $t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b \right) \geq 1$ for $n = 1, \ldots, N$ where $t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b \right) = 1$ for the smallest distances

- Terminology: At equality, a constraint is "active." At > 1, the constraint is "inactive"
- Then, we can say:

$$\arg\min_{w,b} \frac{1}{2} \|\mathbf{w}\|^2 \text{ such that } t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b\right) \geq 1 \forall n \tag{10}$$

- *How do we solve this?* We use Lagrangian Optimization

$$\mathscr{L}(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b\right) - 1\right\} \tag{11}$$

  where $\mathbf{a} = [a_1, \ldots, a_N]^T$ with $a_n \geq 0$

- The KKT (Karush-Kuhn-Tucker conditions):

$$t_n(\mathbf{w}^T \phi(\mathbf{x}_n) + b) - 1 \geq 0 \tag{12}$$
$$a_n \geq 0 \tag{13}$$
$$a_n(t_n(\mathbf{w}^T \phi(\mathbf{x}_n) + b) - 1) = 0 \tag{14}$$

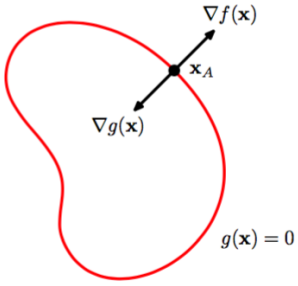


Figure E.1 A geometrical picture of the technique of Lagrange multipliers in which we seek to maximize a function $f(\mathbf{x})$, subject to the constraint $g(\mathbf{x}) = 0$. If $\mathbf{x}$ is D dimensional, the constraint $g(\mathbf{x}) = 0$ corresponds to a subspace of dimensionality $D - 1$, indicated by the red curve. The problem can be solved by optimizing the Lagrangian function $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$.

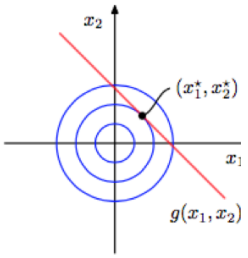
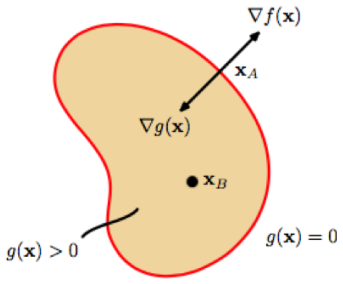


Figure E.2 A simple example of the use of Lagrange multipliers in which the aim is to maximize $f(x_1, x_2) = 1 - x_1^2 - x_2^2$ subject to the constraint $g(x_1, x_2) = 0$ where $g(x_1, x_2) = x_1 + x_2 - 1$. The circles show contours of the function $f(x_1, x_2)$, and the diagonal line shows the constraint surface $g(x_1, x_2) = 0$.

Figure E.3 Illustration of the problem of maximizing $f(\mathbf{x})$ subject to the inequality constraint $g(\mathbf{x}) \geqslant 0$.



- So, we can optimize with respect to $\mathbf{w}$ and b:

$$\frac{\partial \mathscr{L}(\mathbf{w}, b, \mathbf{a})}{\mathbf{w}} = \mathbf{w} - \sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) = 0 \tag{15}$$

$$\mathbf{w} = \sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \tag{16}$$

$$\frac{\partial \mathscr{L}(\mathbf{w}, b, \mathbf{a})}{b} = -\sum_{n=1}^{N} a_n t_n = 0 \tag{17}$$

$$\sum_{n=1}^{N} a_n t_n = 0 \tag{18}$$

- We can plug these into $\mathscr{L}$

$$\begin{aligned}
\mathscr{L}(\mathbf{w}, b, \mathbf{a}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n \left\{ t_n \left(\mathbf{w}^T \phi(\mathbf{x}_n) + b \right) - 1 \right\} && (19)\\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n \left\{ t_n \mathbf{w}^T \phi(\mathbf{x}_n) + t_n b - 1 \right\} && (20)\\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n t_n \mathbf{w}^T \phi(\mathbf{x}_n) - \sum_{n=1}^{N} a_n t_n b + \sum_{n=1}^{N} a_n && (21)\\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n t_n \mathbf{w}^T \phi(\mathbf{x}_n) - b\sum_{n=1}^{N} a_n t_n + \sum_{n=1}^{N} a_n && (22)\\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n t_n \mathbf{w}^T \phi(\mathbf{x}_n) - 0 + \sum_{n=1}^{N} a_n && (23)\\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} a_n t_n \mathbf{w}^T \phi(\mathbf{x}_n) + \sum_{n=1}^{N} a_n && (24)\\
& && (25)
\end{aligned}$$

- Plug in for $\mathbf{w}$:

$$\begin{aligned}
\mathscr{L}(\mathbf{w}, b, \mathbf{a}) &= \frac{1}{2}\left(\sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \right)^T \left(\sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \right) - \sum_{n=1}^{N} a_n t_n \left(\sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \right)^T \phi(\mathbf{x}_n) + \sum_{n=1}^{N} a_n \\
&= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) - \sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) + \sum_{n=1}^{N} a_n \\
&= \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \\
&= \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)
\end{aligned}$$

- This gives us the dual. We want to maximize this wrt a_i:

$$\max_{\mathbf{a}} \mathscr{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) \text{ such that } a_n \geq 0, \sum_{n=1}^{N} a_n t_n = 0 \quad (26)$$

- This is a quadratic programming problem: A quadratic objective with linear constraints
- We can plug into y with out solved $\mathbf{w}$ form: $y(\mathbf{x}) = \sum_{n=1}^{N} a_n t_n \mathbf{K}(\mathbf{x}, \mathbf{x}_n) + b$

- We can look at the KKT conditions for the dual:

$$t_n y(\mathbf{x}_n) - 1 \geq 0 \tag{27}$$
$$a_n \geq 0 \tag{28}$$
$$a_n(t_n y(\mathbf{x}_n) - 1) = 0 \tag{29}$$

- Either $a_n = 0$ or $t_n y(\mathbf{x}_n) = 1$. When $t_n y(\mathbf{x}_n) = 1$, then $\mathbf{x}_n$ is a support vector.
- Using $t_n y(\mathbf{x}_n) = 1$ for support vectors, we can solve for b.

$$t_n \left(\sum_{m \in S} a_m t_m \mathbf{K}(\mathbf{x}_n, \mathbf{x}_m) \right) = 1 \tag{30}$$

- Average over all S.V.s:

$$b = \frac{1}{N_S} \sum_{n \in S} \left(t_n - \sum_{m \in S} a_m t_m \mathbf{K}(\mathbf{x}_n, \mathbf{x}_m) \right) \tag{31}$$

because:

$$t_n \left[t_n \left(\sum_{m \in S} a_m t_m \mathbf{K}(\mathbf{x}_n, \mathbf{x}_m) + b \right) \right] = (1) t_n \tag{32}$$
$$\sum_{m \in S} a_m t_m \mathbf{K}(\mathbf{x}_n, \mathbf{x}_m) + b = t_n \tag{33}$$
$$b = t_n - \sum_{m \in S} a_m t_m \mathbf{K}(\mathbf{x}_n, \mathbf{x}_m) \tag{34}$$