As in the previous lecture, consider the classification setting. Let $\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \{+1, -1\}$, and

$$\mathcal{H} = \{\psi x + b,\ \psi \in \mathbb{R}^d,\ b \in \mathbb{R}\}$$

where $|\psi| = 1$.

We would like to maximize over the choice of hyperplanes the minimal distance from the data to the hyperplane:

$$\max_H \min_i d(x_i, H),$$

where

$$d(x_i, H) = y_i(\psi x_i + b).$$

Hence, the problem is formulated as maximizing the margin:

$$\max_{\psi,b} \underbrace{\min_i y_i(\psi x_i + b)}_{m \text{ (margin)}}.$$

Rewriting,

$$y_i(\psi' x_i + b') = \frac{y_i(\psi x_i + b)}{m} \geq 1,$$

$\psi' = \psi/m$, $b' = b/m$, $|\psi'| = |\psi|/m = 1/m$. Maximizing m is therefore minimizing $|\psi'|$. Rename $\psi' \to \psi$, we have the following formulation:

$$\min |\psi| \quad \text{such that} \quad y_i(\psi x_i + b) \geq 1$$

Equivalently,

$$\min \frac{1}{2}\psi \cdot \psi \quad \text{such that} \quad y_i(\psi x_i + b) \geq 1$$

Introducing Lagrange multipliers:

$$\phi = \frac{1}{2}\psi \cdot \psi - \sum \alpha_i (y_i(\psi x_i + b) - 1),\ \ \alpha_i \geq 0$$

Take derivatives:

$$\frac{\partial \phi}{\partial \psi} = \psi - \sum \alpha_i y_i x_i = 0$$

$$\frac{\partial \phi}{\partial b} = -\sum \alpha_i y_i = 0$$

Hence,

$$\psi = \sum \alpha_i y_i x_i$$

and

$$\sum \alpha_i y_i = 0.$$

Substituting these into ϕ,

$$
\begin{aligned}
\phi &= \frac{1}{2}\left(\sum \alpha_i y_i x_i\right)^2 - \sum_{i=1}^{n} \alpha_i \left(y_i \left(\sum_{j=1}^{n} \alpha_j y_j x_j x_i + b \right) - 1 \right) \\
&= \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j - \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j - b \sum \alpha_i y_i + \sum \alpha_i \\
&= \sum \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j y_i y_j x_i x_j
\end{aligned}
$$

The above expression has to be maximized this with respect to α_i, $\alpha_i \geq 0$, which is a Quadratic Programming problem.

Hence, we have $\psi = \sum_{i=1}^{n} \alpha_i y_i x_i$.

Kuhn-Tucker condition:

$$\alpha_i \neq 0 \Leftrightarrow y_i(\psi x_i + b) - 1 = 0.$$

Throwing out non-support vectors x_i does not affect hyperplane $\Rightarrow \alpha_i = 0$.

The mapping ϕ is a *feature* mapping:

$$x \in \mathbb{R}^d \longrightarrow \phi(x) = (\phi_1(x), \phi_2(x), ...) \in \mathcal{X}'$$

where $\mathcal{X}'$ is called *feature space*.

Support Vector Machines find optimal separating hyperplane in a very high-dimensional space. Let $K(x_i, x_j) = \sum_{k=1}^{\infty} \phi_k(x_i)\phi_k(x_j)$ be a scalar product in $\mathcal{X}'$. Notice that we don't need to know mapping $x \to \phi(x)$. We only need to know $K(x_i, x_j) = \sum_{k=1}^{\infty} \phi_k(x_i)\phi_k(x_j)$, a symmetric positive definite kernel.

Examples:

(1) Polynomial: $K(x_1, x_2) = (x_1 x_2 + 1)^\ell$, $\ell \geq 1$.

(2) Radial Basis: $K(x_1, x_2) = e^{-\gamma |x_1 - x_2|^2}$.

(3) Neural (two-layer): $K(x_1, x_2) = \frac{1}{1+e^{\alpha x_1 x_2 + \beta}}$ for some α, β (for some it's not positive definite).

Once α_i are known, the decision function becomes

$$\text{sign}\left(\sum \alpha_i y_i x_x \cdot x + b\right) = \text{sign}\left(\sum \alpha_i y_i K(x_i, x) + b\right)$$