Explore Dual SVM

The original, **primal SVM**, is a quadratic optimization problem. We can substitute the original feature representation of a data point $\mathbf{x}_i$ with an abstract representation $\phi\left(\mathbf{x}_i\right)$ to get the following primal formulation:

$$\begin{aligned} \min_{\mathbf{w},b} \mathbf{w}^\top \mathbf{w} + C \sum_{i=1}^{n} \xi_i \\ \text{such that } \forall i, y_i \left(\mathbf{w}^\top \phi\left(\mathbf{x}_i\right) + b\right) \geq 1 - \xi_i \\ \forall i, \xi_i \geq 0 \end{aligned}$$

The dual form of this optimization problem is:

$$\begin{aligned} \min_{\alpha_1,\ldots,\alpha_n} \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^\top \phi\left(\mathbf{x}_j\right) - \sum_{i=1}^{n} \alpha_i \\ \text{such that } \forall i, 0 \leq a_i \leq C \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned}$$

Although the weight vector $\mathbf{w}$ from the primal form is never explicitly computed in the dual form, it is known that it is equal to $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \phi\left(\mathbf{x}_i\right)$. The expression for b comes out to be a function of $\alpha_1, \ldots, \alpha_n$ without any $\mathbf{x}_i$ terms. You'll learn more about these equalities in a later video. The final classifier is:

$$\begin{aligned} h\left(\mathbf{x}\right) &= \text{sign}\left(\mathbf{w}^\top \phi(\mathbf{x}) + b\right) \\ &= \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i \phi(\mathbf{x}_i)^\top \phi\left(\mathbf{x}\right) + b\right) \end{aligned}$$

Notice that the final classifier can be expressed as inner products of pairs of data points: $\phi(\mathbf{x}_i)^\top \phi\left(\mathbf{x}\right)$. However, as we have shown earlier, expanding feature vectors to high dimensions explicitly and then computing inner products is computationally expensive. This leads us to the kernel trick.

Essentially, the idea is to define a kernel function $\mathsf{K}\left(\mathbf{x}_i, \mathbf{x}\right)$ that takes fewer computations than computing $\phi(\mathbf{x}_i)^\top \phi\left(\mathbf{x}\right)$.