Exercise session notes - Week 6

Slater's Condition + Dual SVM and Kernels

This week we concluded the chapter on Duality with another sufficient condition for Strong Duality, namely Slater's condition. Consider a convex program

$$\begin{aligned} \min \quad & f(x) \\ & g_i(x) \leq 0 \quad \forall i \\ & h_j(x) = 0 \quad \forall j \end{aligned}$$

Definition. We say a convex program satisfies **Slater's condition** if there exists a point $x \in \mathbb{R}^n$ in the relative interiour of the feasibility set of the program, i.e.

$$g_i(x) < 0 \ , \ h_j(x) = 0 \qquad \forall i, j$$

Theorem. For convex mathematical programs

$$\text{Slater's Condition holds} \implies \text{Strong Duality holds}$$

Dual Support Vector Machine

Now we compute the dual program of the Support Vector Machine. Recall that we are given points $x_1, \dots, x_m \in \mathbb{R}^n$ and labels $y_1, \dots, y_m \in \{-1, 1\}$ and the goal is to find an hyperplane separating points with positive labels from points with negative labels. We formulated the (primal) SVM as

$$\begin{aligned} & \min \ \|w\|^2 \\ \text{s.t.} \qquad & y_i(w^\top x_i + b) \geq 1 \qquad \forall i \in [m] \\ & w \in \mathbb{R}^n \\ & b \in \mathbb{R} \end{aligned}$$

Dual formulation. The dual formulation what we computed is the following

$$\begin{aligned} \max \ & \sum_{i=1}^m \lambda_i - \frac{1}{4} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \left(x_i^\top x_j\right) \\ \text{s.t} \qquad & \sum_{i=1}^m \lambda_i y_i = 0 \\ & \lambda \in \mathbb{R}^m_{\geq 0} \end{aligned}$$

You can find the explicit calculations in Chapter 3.8.1 of the Lecture Notes.

Returning to primal solution. Assume we solve the dual program and we obtain the optimal solution $(\lambda_i)_{i\in[m]}$. Then we can find the primal solution by computing

$$w = \frac{1}{2}\sum_{i=1}^{m} \lambda_i y_i x_i \quad , \quad b = \begin{cases} \max\left\{+1 - w^\top x_i \;\middle|\; \text{ for } y_i = +1\right\} \\ \min\left\{-1 - w^\top x_i \;\middle|\; \text{ for } y_i = -1\right\} \end{cases}$$

Finally, to classify a new vector $z \in \mathbb{R}^n$ we do the following

$$w^\top z + b \begin{cases} > 0 \implies \text{label } = +1 \\ < 0 \implies \text{label } = -1 \end{cases}$$

Remarks. We first note that if the points are linearly separable, then there exists w, b such that the constraints in the primal formulation are satisfied with strictly inequalities. In this case Slater's condition is satisfied, hence Strong Duality holds. Additionally we can see that

$$\text{primal has } n+1 \text{ variables and } m \text{ constraints}$$
$$\text{dual has } m \text{ variables and } m+1 \text{ constraints}$$

Therefore if $n \gg m$ (the dimension of the space in which the points are, is much larger than the amount of points), then it is faster to solve the dual instead of the primal. This is the case for Image Classification where we want to characterize images in two clusters. An image could be saved as a point in a 500 × 500 × 3 dimensional space (each coordinate corresponding the intensity of a color in a single pixel).

Kernels. In the case where the points are not linearly separable, we solved the problem using feature maps $\Phi : \mathbb{R}^n \to \mathbb{R}^{n'}$ so that by increasing the dimension of the space we can find a separating hyperplane, but usually we end up with $n' \gg n$. Hence solving the primal will be much slower after applying the feature maps. In the dual we instead only have to compute the scalar products between points

$$\Phi(x_i)^\top \Phi(x_j)$$

and this doesn't require an explicit computation of the feature map. This scalar product is called a Kernel $K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, and here we present some popular ones:

- Linear Kernel $K(x, z) = x^\top z$, which corresponds to the indentity function as feature map
- Polynomial Kernel $K(x, z) = (1 + x^\top z)^d$, which contains all polynomial of degree at most d
- Gaussian Kernel $K(x, z) = e^{-\frac{\|x-z\|^2}{\sigma}}$, which is the most powerful Kernel as the feature map corresponding to it is infinite dimensional