COMP 7070 Advanced Topics in AI and ML March 24, 2025

Lecture 8: Kernel Tricks and RKHS

Instructor: Yifan Chen *Scribes: Fu Rao, Kong Chuyi* *Proof reader: Xiong Peng*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

8.1 Kernel Trick

Kernel method or **kernel trick** are widely used in machine learning and are commonly used to solve nonlinearly differentiable problems with data. These methods rely on **kernel function**, which allows us to generalize the original input space to higher latitude spaces under certain conditions. Under certain conditions, we can accomplish this by using the **kernel matrix** instead of the inner product of pairs of data in the original space. The **kernel method** is expressed as:

$$\langle X, Z \rangle \Rightarrow K(x, z)$$

Here, $K(x, z)$ is the kernel function that computes the inner product in the feature space.

8.1.1 PSD Kernel

The most important type of kernel is the **Positive Semi-Definite (PSD) Kernel**. PSD kernels ensure that the optimization problems solved by learning algorithms have a global minimum, which is crucial for the convergence and stability of the algorithm.

8.1.1.1 Kernel Matrix (Gram Matrix)

The **kernel matrix** (or **Gram matrix**) is a matrix where each element represents the kernel function applied to a pair of samples. Mathematically, the ij-th element of the Gram matrix G is defined as:

$$G_{ij} = K(x_i, x_j) \quad \forall i, j \in \{1, 2, \ldots, n\}$$

A matrix is **positive semi-definite** if for any vector $z \in \mathbb{R}^n$, the following holds:

$$z^T G z \geq 0$$

A kernel is considered **PSD** if its corresponding kernel matrix is positive semi-definite for any possible set of input data points.

8.1.1.2 Linear Kernel

Definition: The linear kernel is defined as the dot product between two vectors:

$$K(x, y) = x^T y$$

Proof of PSD Property: For any finite set of points $\{x_1, x_2, \ldots, x_n\}$ and any $z \in \mathbb{R}^n$, the inner

product (dot product) of the vectors is a bilinear form. Since the dot product of any vector with itself is always non-negative, the linear kernel satisfies the PSD property.

8.1.1.3 Polynomial Kernel

Definition: For degree-d polynomials, the polynomial kernel is defined as:

$$K(x, y) = (x^T y + r)^d, \quad x, y \in \mathbb{R}^d, \quad r \geq 0, \quad d \geq 1$$

This can also be expanded as:

$$K(x, y) = \sum_{k=0}^{d} \binom{d}{k} \langle x, y \rangle^k r^{d-k}$$

The polynomial kernel introduces non-linearity by raising the dot product to a power, allowing the model to capture more complex relationships in the data.

8.1.2 Mercer's Theorem

Mercer's Theorem is a fundamental result in functional analysis, particularly in the context of kernel methods in machine learning. It provides conditions under which a kernel function can be expressed as an infinite series expansion in terms of orthogonal basis functions.

8.1.2.1 Orthonormal Basis

An **orthonormal basis** in a Hilbert space (such as $L^2(X)$, the space of square-integrable functions) is a set of basis vectors that are mutually orthogonal and of unit length. For any two basis vectors ϕ_i and ϕ_j:

1. **Orthogonality**: $\langle \phi_i, \phi_j \rangle = 0$ for $i \neq j$.
2. **Normalization**: $\langle \phi_i, \phi_i \rangle = 1$.

Any function f in $L^2(X)$ can be represented as a linear combination of these basis functions:

$$f \in L^2(X) : f = \sum_{i=1}^{\infty} \beta_i \cdot \phi_i$$

8.1.2.2 Non-negative Eigenvalues

In the context of Mercer's theorem, the **eigenvalues** $\{\mu_i\}$ correspond to the weights associated with each orthogonal basis function ϕ_i in the expansion of the kernel function. These eigenvalues are non-negative due to the positive semi-definiteness of the kernel $K(x, y)$.
The integration form of Mercer's theorem is:

$$\int_X K(x, z) \cdot \phi_j(z)\, dz = \mu_j \cdot \phi_j(x)$$

The inner product form is:

$$\langle K(x, \cdot), \phi_j(\cdot) \rangle = \mu_j \cdot \phi_j(x)$$

8.1.2.3 Kernel Function Expansion

Combining the above properties, the kernel function $K(x, y)$ can be expressed as:

$$K(x, y) = \sum_{i=1}^{\infty} \mu_i \phi_i(x) \phi_i(y)$$

This is the essence of **Mercer's Theorem**, which states that any PSD kernel can be expanded into an infinite series of orthogonal basis functions with non-negative eigenvalues.

8.1.3 RKHS

Reproducing kernel Hilbert space (RKHS) is the most important concept in kernel method. Each PSD kernel function will implicitly map the data to a unique RKHS. Therefore, studying the properties of RKHS is very important for understanding the principles of nuclear methods.

8.1.3.1 Hilbert Space and RKHS

RKHS is essentially a specific case of **Hilbert space (HS)**. A Hilbert space is a real or complex inner product space that is also a complete metric space with respect to the distance function induced by the inner product.

Generally speaking, the elements in the Hilbert space can be real numbers or complex numbers, and there is an finite inner product between every two elements. At the same time, we can calculate the distance between elements through the inner product. For example, for a Hilbert space consisting of functions, the inner products between functions are of the following form:

$$\langle f, g \rangle = \int f(x) g(x) \, dx \leq \infty$$

If a Hilbert space $\mathcal{H}$ consists of functions $f : \mathcal{X} \to \mathbb{R}$, it is called a reproducing kernel Hilbert space if there exists a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that satisfies:

1. For any $x \in \mathcal{X}$, the function $K_x(\cdot) = K(x, \cdot)$ is a element of $\mathcal{H}$
2. **Reproducing Property:** For any $f \in \mathcal{H}$ and $x \in \mathcal{X}$, we have $f(x) = \langle f, K_x \rangle_{\mathcal{H}}$

As the name implies, RKHS is a reproducible Hilbert space, which is a very important property. Due to the reproducibility, all PSD kernels correspond to a unique RKHS, and the PSD kernel we use will project our data into this unique RKHS.

8.1.3.2 The Representer Theorem

The representer theorem is a fundamental result in the theory of kernel methods, which plays a central role in the theoretical analysis and design of machine learning algorithms. This theorem solves the problem of finding the optimal solution in infinite-dimensional RKHS, proving that the optimal solution can be expressed as a linear combination of a finite number of kernel functions, thereby transforming the infinite-dimensional optimization problem into a finite-dimensional problem.

Consider the regularized risk minimization problem in the reproducing kernel Hilbert space $\mathcal{H}_K$:

$$\min_{f \in \mathcal{H}K} \sum i = 1^n L(y_i, f(x_i)) + \lambda |f|^2_{\mathcal{H}_K}$$

where:

- $(x_i, y_i)_{i=1}^n$ is the training sample
- L is the loss function
- $\lambda > 0$ is the regularization parameter
- $||f||_{\mathcal{H}_K}$ is the norm of function (f) in RKHS

Conclusion of the representation theorem: Any optimal solution f^* to the above optimization problem can be represented as:

$$f^*(x) = \sum_{i=1}^{n} \alpha_i K(x, x_i)$$

where $\alpha_i \in \mathbb{R}$ are fixed coefficients, and K is the PSD kernel.

8.1.3.3 Kernel Trick Revisited

8.1.3.3.1 Example: Kernel Ridge Regression

For a ridge regression problem with L^2 Regularization:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \cdot [Y - f(x)]^T \cdot [Y - f(x)] + \lambda \|f\|_{\mathcal{H}}^2.$$

we can place all function f by using the representer theorem, $f = \sum_{i=1}^n \alpha_i K(x_i, \cdot)$. Now we have:

$$\begin{aligned} f(x_j) = \langle f, K(x_j, \cdot) \rangle_{\mathcal{H}} &= \sum_{i=1}^{n} \alpha_i \cdot K(x_i, x_j) \\ \Rightarrow f(X) &= K\alpha \\ \Rightarrow \|f\|_{\mathcal{H}}^2 = \langle f, f \rangle_{\mathcal{H}} &= \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j K(x_i, x_j) = \alpha^T K \alpha \end{aligned}$$

Then, for the original problem, we can use Lagrange multiplier method:

$$\begin{aligned} \min_{\alpha} \frac{1}{n} (Y - K\alpha)^T (Y - K\alpha) + \lambda \cdot \alpha^T K \alpha \\ \Rightarrow \frac{2}{n} \cdot (-K)(Y - K\alpha) + 2\lambda K\alpha = 0 \\ \Rightarrow (K + n\lambda I)\alpha = Y \\ \Rightarrow \alpha^* = (K + n\lambda I)^{-1} Y \end{aligned}$$

Finally, we can get

$$\begin{aligned} \hat{f}(X) &= K \cdot \alpha^* = K(K + n\lambda I)^{-1} Y \\ \hat{f}(Z) &= \sum_{i=1}^{n} \alpha_i^* \cdot K(x_i, Z). \end{aligned}$$

8.1.3.3.2 Example: Neural Tangent Kernel

The Neural Tangent Kernel (NTK) is a theoretical tool in deep learning for analyzing the behavior of neural networks in the limit of infinite-width layers.

The basic idea of NTK is to perform linear analysis during the training process of the neural network.

Considering the neural network parameters $\theta(t)$ at time t, we can approximate the network function by a first-order Taylor expansion:

$$f(\theta(t), x_i) \approx f(\theta(0), x_i) + \Phi^T(x_i)(\theta(t) - \theta(0))$$

Thus, we can define NTK as follows:

$$NTK(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle = \left\langle \frac{\partial f(\theta(t), x_i)}{\partial \theta}, \frac{\partial f(\theta(t), x_j)}{\partial \theta} \right\rangle$$

It should be noted that NTK is only meaningful under certain conditions. When using NTK for discussion, the neural network must be very wide, even infinitely wide.

8.1.3.3.3 Example: Attention as Kernel Estimator

Considering the formula of the attention mechanism, we can combine it with Nadaraya-Watson (NW) kernel estimator to get the following formula:

$$\begin{aligned} f(q_i) &= \frac{\exp(\langle q_i, k_j \rangle) \cdot v_i}{\sum_{j=1}^{n} \exp(\langle q_i, k_j \rangle)} \\ &= \frac{K(q_i, k_j) \cdot v_j}{\sum_{j=1}^{n} K(q_i, k_j)} \Leftrightarrow \text{NW estimator in kernel methods.} \\ f(Q) &= D^{-1} \tilde{K}(Q, K) \cdot V \text{ where } D = diag(\tilde{K}(Q, K) \cdot \mathbf{1}_n). \end{aligned}$$

This connection reveals an important property of the attention mechanism: it is essentially a data-based nonparametric learning system. It does not rely on a fixed parameter structure, but dynamically calculates based on the relationship between the inputs.