Kernel and RKHS:

From Theory to Applications

Yibin Xiong

December 17, 2021

Table of Contents

Hilbert Space

A **Hilbert space** is a *complete inner product space*

- A natural generalization of $\mathbb{R}^n$ to infinite dimension (think of real-valued functions as infinitely dimensional vectors)
- Inner product provides the geometric notions analogous to those in $\mathbb{R}^n$, for example orthogonality
- We often want a Hilbert space to be *separable*, which means it has an orthonormal basis with *countably* many elements

Hilbert Space

A **Hilbert space** is a *complete inner product space*

- A natural generalization of $\mathbb{R}^n$ to infinite dimension (think of real-valued functions as infinitely dimensional vectors)
- Inner product provides the geometric notions analogous to those in $\mathbb{R}^n$, for example orthogonality
- We often want a Hilbert space to be *separable*, which means it has an orthonormal basis with *countably* many elements

Examples:

- $\mathbb{R}^d$ with $\langle x, y \rangle := \sum_{i=1}^{d} x_i y_i$
  An orthonormal basis is $\{e_1, \ldots, e_d\}$
- $L_2[a, b]$ (square integrable functions) with $\langle f, g \rangle := \int_a^b f(x)g(x)dx$
  An orthonormal basis is $\left\{1, \cos(\frac{2n\pi}{b-a}x), \sin(\frac{2n\pi}{b-a}x)\right\}_{n=1}^{\infty}$

Counter-examples: $C[a, b]$ and $L_1[a, b]$ (MIT slide 15 & UCL Figure 2.1)

Functionals

- ▷ Roughly speaking, a **functional** is a "function of functions".
  - It takes functions as input, rather than scalars or vectors, and outputs scalars.
  - It generalizes the domain of a function from scalar-valued vector spaces to abstract vector spaces.
- ▷ An **operator** further generalizes the codomain to any space.

Functionals

▷ Roughly speaking, a **functional** is a "function of functions".

- It takes functions as input, rather than scalars or vectors, and outputs scalars.
- It generalizes the domain of a function from scalar-valued vector spaces to abstract vector spaces.

▷ An **operator** further generalizes the codomain to any space.

▷ Possible properties of $A : \mathcal{F} \to \mathcal{G}$

- linearity

$$A(\alpha f + \beta g) = \alpha A(f) + \beta A(g) \quad \forall f, g \in \mathcal{H} \ \forall \alpha, \beta \in \mathbb{R}$$

- continuity

$$\forall f \in \mathcal{F} : \|f_n - f\|_{\mathcal{F}} \to 0 \Rightarrow \|A(f_n) - A(f)\|_{\mathcal{G}} \to 0$$

- boundedness

$$\exists M > 0 \text{ s.t. } \forall f \in \mathcal{F} : \|A(f)\|_{\mathcal{G}} \le M \|f\|_{\mathcal{H}}$$

Functionals

The smallest possible M is the operator norm of A, defined as

$$\|A\| := \sup_{f \in \mathcal{F}} \frac{\|A(f)\|_{\mathcal{G}}}{\|f\|_{\mathcal{F}}}$$

Boundedness-Continuity Equivalence (Thm 21 in UCL Notes)

Let $(\mathcal{F}, \|\cdot\|_{\mathcal{F}})$, $(\mathcal{G}, \|\cdot\|_{\mathcal{G}})$ be normed vector spaces. If $L : \mathcal{F} \to \mathcal{G}$ is a *linear* operator, then L is bounded if and only if L is continuous on $\mathcal{F}$.

Functionals

The smallest possible M is the operator norm of A, defined as

$$\|A\| := \sup_{f \in \mathcal{F}} \frac{\|A(f)\|_{\mathcal{G}}}{\|f\|_{\mathcal{F}}}$$

Boundedness-Continuity Equivalence (Thm 21 in UCL Notes)

Let $(\mathcal{F}, \|\cdot\|_{\mathcal{F}})$, $(\mathcal{G}, \|\cdot\|_{\mathcal{G}})$ be normed vector spaces. If $L : \mathcal{F} \to \mathcal{G}$ is a *linear* operator, then L is bounded if and only if L is continuous on $\mathcal{F}$.

$\triangleright$ The (Dirac) **evaluation functional** on a Hilbert space of functions $\mathcal{H}$ is the mapping $\delta_x : \mathcal{H} \to \mathbb{R}$, $f \mapsto f(x)$.

- Evaluation functionals are *linear*, but not necessarily continuous or bounded.

  e.g. evaluation functionals of $L_2[a, b]$ are unbounded (MIT slide 16,17)

Evaluation Functional and Representer

Riesz Representation Theorem

For every *continuous* linear functional L in a Hilbert space $\mathcal{H}$, there exists a *unique* $g_L \in \mathcal{H}$ s.t. $\forall f \in \mathcal{H} : L(f) = \langle g_L, f \rangle$.

- In short, $L(\cdot) \in \mathcal{H} \;\equiv\; \langle g_L, \cdot \rangle \in \mathcal{H}^*$
- More precisely, there is a *isometric* (preserves distance) *isomorphism* (linear bijection) between $\mathcal{H}$ and its topological dual $\mathcal{H}^*$.

  $$\exists \text{ a linear bijective map } U : \mathcal{H} \to \mathcal{H}^* \; s.t. \; \langle h_1, h_2 \rangle_{\mathcal{H}} = \langle Uh_1, Uh_2 \rangle_{\mathcal{H}^*}$$

- We will see that a kernel $k(x_i, \cdot) \in \mathcal{H}$ is a *representer* of the evaluation functional δ_{x_i} on $\mathcal{H} \to \mathbb{R}$ simply because of the reproducing property.

Reproducing Kernel Hilbert Space (RKHS)

Abstract Definition (Def 26 in UCL Notes):

A **reproducing kernel Hilbert space** is a Hilbert space $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$ where the evaluation functionals $\delta_x : \mathcal{H} \to \mathbb{R}$ are *continuous* $\forall x \in \mathcal{X}$.

- Evaluation functionals are continuous provides a nice property of the functions in $\mathcal{H}$: convergence in $\| \cdot \|_{\mathcal{H}}$ implies pointwise convergence.
- To make an RKHS, we impose this property on a Hilbert space: we keep the elements (or say sequences of functions) that satisfy "convergence in norm implies pointwise convergence", and discard the others.

  # For some counterexamples that don't have this property, see UCL Note Example 28, CMU Notes page 12 "Evaluation Functionals"

Reproducing Kernel Hilbert Space (RKHS)

Constructive Definition (using "kernel" in the definition):

Let $\mathcal{H}$ be a Hilbert space of real-valued functions defined on a non-empty set $\mathcal{X}$

$\triangleright$ A **reproducing kernel** of $\mathcal{H}$ is a function $k(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ s.t.
i) $\forall x \in \mathcal{X} : k(x,\cdot) \in \mathcal{H}$ and
ii) $\forall x \in \mathcal{X} : \forall f \in \mathcal{H} : \langle k(x,\cdot), f(\cdot)\rangle_{\mathcal{H}} = f(x)$ (the reproducing property)

$\triangleright$ A **reproducing kernel Hilbert space** is a Hilbert space $\mathcal{H}$ with a reproducing kernel whose linear span is dense in $\mathcal{H}$ (Purdue Notes page 2).

$$\mathrm{cl}\left(\text{span}\left\{k(x,\cdot)\right\}_{x\in\mathcal{X}}\right) = \mathcal{H}$$

[1] For a more rigorous construction, see UCL Notes page 11-18

Reproducing Kernel Hilbert Space (RKHS)

Constructive Definition (using "kernel" in the definition):

Let $\mathcal{H}$ be a Hilbert space of real-valued functions defined on a non-empty set $\mathcal{X}$

$\triangleright$ A **reproducing kernel** of $\mathcal{H}$ is a function $k(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ s.t.
i) $\forall x \in \mathcal{X} : k(x,\cdot) \in \mathcal{H}$ and
ii) $\forall x \in \mathcal{X} : \forall f \in \mathcal{H} : \langle k(x,\cdot), f(\cdot)\rangle_{\mathcal{H}} = f(x)$ (the reproducing property)

$\triangleright$ A **reproducing kernel Hilbert space** is a Hilbert space $\mathcal{H}$ with a reproducing kernel whose linear span is dense in $\mathcal{H}$ (Purdue Notes page 2).

$$\mathrm{cl}\left(\text{span}\left\{k(x,\cdot)\right\}_{x\in\mathcal{X}}\right) = \mathcal{H}$$

- The "abstract" definition and "constructive" definition are *equivalent*. *Riesz representation theorem* shows "abstract" def $\Rightarrow$ "constructive" def. The other direction is easy to show by the reproducing property.
- Every RKHS $\mathcal{H}_k$ has one and only one reproducing kernel $k(\cdot,\cdot)$ (UCL Notes Prop. 30 and Thm. 31).
- Starting with a Hilbert space $\mathcal{H}$, we find the kernel function $k(\cdot,\cdot)$, span the elements into a subspace, and take its closure to get an RKHS $\mathcal{H}_K$.[1]

[1] For a more rigorous construction, see UCL Notes page 11-18

Reproducing Kernels are Mercer Kernels

▷ Indeed, reproducing kernels have nice properties:

- $\langle k(x,\cdot), k(y,\cdot)\rangle_{\mathcal{H}} = k(x,y)$ by the reproducing property[2]
- Symmetry: $k(x,y) = k(y,x)\ \forall x, y \in \mathcal{X}$
- Positive definiteness[3]:

$$\forall (a_1,\dots,a_n) \in \mathbb{R}^n : \forall (x_1,\dots,x_n) \in \mathcal{X}^n : \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j k(x_i, x_j) \geq 0$$

* If for mutually distinct $\{x_1,\dots x_n\}$ we have that the quadratic form $= 0$ if and only if $(a_1,\dots,a_n) = \mathbf{0}$, then $k(\cdot,\cdot)$ is *strictly* positive definite.

Symmetric positive-definite functions are called **Mercer kernels**.

▷ All Mercer kernels have the reproducing property (Moore–Aronszajn theorem), so they are equivalent.

[2] This corresponds to the definition of a kernel, i.e. dot product of the feature maps.

[3] I am curious why people don't use *integral* in this definition

Mercer's Theorem and Kernel Trick

▷ Spectral decomposition

If $\int_{\mathcal{X}} k(x,y)\phi(y)\,dy = \lambda\phi(x)$, then λ is an eigenvalue and $\phi(x)$ is an eigenfunction of the kernel $k(\cdot,\cdot)$.

Mercer's Theorem

1. The eigenvalues of a Mercer kernel $\{\lambda_i\}_{i=1}^{\infty}$ are absolutely summable.
2. $k(x,y) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x)\phi_i(y)$ holds and the series converges absolutely and uniformly.

We define the **feature map** as $\Phi(x) = \left(\sqrt{\lambda_1}\phi_1(x), \sqrt{\lambda_2}\phi_2(x), \dots\right) \in \mathbb{R}^{\infty}$

By Mercer's Theorem, $k(x,y) = \langle \Phi(x), \Phi(y) \rangle_{\ell_2}$

- This means given a kernel function $k(\cdot,\cdot)$, automatically we have an inner product of inherently high-dimensional mappings $\Phi(\cdot)$
- In practice, we do not need to figure out the explicit formula of $\Phi(\cdot)$ but directly use a kernel function, which often associates with a complex, high/infinite-dimensional feature map.

Aside: Another way of Constructing RKHS using Mercer's Theorem

Let $\mathcal{X}$ be a *compact metric space* [4] and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a *continuous kernel* function. Define:

$$\mathcal{H}_K := \left\{ f \;\middle|\; f(s) = \sum_{j \in J} c_j \phi_j(s) \text{ and the sequence } \left\{ \frac{c_j}{\sqrt{\lambda_j}} \right\} \in \ell_2(J)^{5} \right\}$$

and the inner product

$$\left\langle \sum_{j \in J} a_j \phi_j \,, \sum_{j \in J} b_j \phi_j \right\rangle_{\mathcal{H}_K} := \sum_{j \in J} \frac{a_j b_j}{\lambda_j}$$

[4] See UCL Note Thm. 51

[5] J is the index set. It is at most countable. $\ell_2(J)$ is the space of "square summable sequences"

Table of Contents

Example Kernel Functions

- Linear kernel: $k(x, y) = x^T y$ or more generally, $k(x, y) = x^T By$ for some $B \succeq 0$.
- Polynomial kernel: $k(x, y) = (x^T y + c)^d$, where $c \geq 0$ and $d \in \mathbb{N}$
- Gaussian/RBF kernel: $k(x, y) = \exp\left(-\gamma \|x - y\|^2\right) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$
- Laplacian kernel: $k(x, y) = \exp\left(-\alpha \|x - y\|\right)$
- Sigmoid kernel: $k(x, y) = \tanh(\gamma x^T y + c)$
- If k_1, k_2 are kernels [6], then $\forall \alpha \geq 0 : \alpha k_1$ and $k_1 + k_2$ are kernels.
- Hadamard product of 2 kernels $k\Big((x, y), (x', y')\Big) := k_1(x, x')k_2(y, y')$ is a kernel on $(\mathcal{X} \times \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$

  In particular, if $\mathcal{X} = \mathcal{Y}$, then $k(x, x') := k_1(x, x')k_2(x, x')$ is a kernel on $\mathcal{X}$.

[6] Indeed, "kernel", "reproducing kernel", and "Mercer kernel" are the same/equivalent (see UCL Notes "4.6 Summary")

Regularized Problems and Representer Theorem

▷ Intuition: Minimize $\|f\|^2_{\mathcal{H}_K}$ as regularization

- $\|\cdot\|_{\mathcal{H}_K}$ is a measure of *smoothness* of a function. [7]

  Thus, kernel regression controls the *smoothness* of the function to avoid underfitting or overfitting. $\|f\|_{\mathcal{H}_K}$ is small $\Leftrightarrow$ f is smooth.

- Intuitively, $\|f\|^2_{\mathcal{H}_K}$ is like a "modified" Lipschitz constant of f (MIT slide 28).

$$
\begin{aligned}
|f(x) - f(x')| &= |\langle f, k(x,\cdot)\rangle - \langle f, k(x',\cdot)\rangle| \\
&= |\langle f, k(x,\cdot) - k(x',\cdot)\rangle| \\
&\leq \|f\|_{\mathcal{H}} \|k(x,\cdot) - k(x',\cdot)\|_{\mathcal{H}} \\
&= \|f\|_{\mathcal{H}} \sqrt{k(x,x) - 2k(x,x') + k(x',x')}
\end{aligned}
$$

Consider the last term as a kind of *distance* $\tilde{d}(x,x')$, then $\|\cdot\|_{\mathcal{H}_K}$ is like a Lipschitz constant.

[7] See a detailed explanation in MLSS 2015 slide 67-69. It involved Fourier series

Regularized Problems and Representer Theorem

Let $\{(x_i, y_i)\}_{i=1}^n$ be the data points

Representer Theorem

Let ℓ be a loss function and g be an nondecreasing function. If $\hat{f} = \arg\min_f \sum_{i=1}^n \ell\Big(y_i, f(x_i)\Big) + g(\|f\|^2_{\mathcal{H}_K})$, then $\exists \alpha_1, \ldots \alpha_n$ s.t. $\hat{f}(x) = \sum_{i=1}^n \alpha_i k(x_i, x)$

- A quick proof is in MIT slide 37-38. The main properties to derive this theorem are *the reproducing property* and *orthogonality*.
- With this theorem, we reduces an optimization problem from infinitely dimensional function space to $\mathbb{R}^n$ (see this in the following examples)

Examples: Kernel Ridge Regression

Kernel Regularization:

$$\min_f \mathcal{L}(f) = \min_f \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}_K}^2$$

Examples: Kernel Ridge Regression

Kernel Regularization:

$$\min_f \mathcal{L}(f) = \min_f \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2 + \lambda\|f\|_{\mathcal{H}_K}^2$$

- By representer theorem, let $\hat{f}(x) = \sum_{i=1}^{n} \alpha_i k(x_i, x)$. Then

$$\mathcal{L}(\hat{f}) = \frac{1}{n}\|y - \mathbb{K}\alpha\|^2 + \lambda\alpha^T\mathbb{K}\alpha$$

  where $\mathbb{K}_{i,j} = k(x_i, x_j)$ is the kernel matrix. $\mathbb{K} \succeq 0$ is symmetric.
- We minimize this quantity over the parameters $\alpha \in \mathbb{R}^n$

$$\nabla_\alpha L(\hat{f}) = \frac{1}{n}(2\mathbb{K}^T\mathbb{K}\alpha - 2\mathbb{K}^T y) + 2\lambda\mathbb{K}\alpha = 0$$

- If we assume $\mathbb{K} \succ 0$, then we can find the unique minimizer $\hat{\alpha} = (\mathbb{K} + n\lambda I)^{-1}y$ and thus $\hat{f}(x) = \sum_{i=1}^{n} \hat{\alpha}_i k(x_i, x)$.

Examples: SVM (Hinge Loss)

Recall that SVM minimizes the penalized hinge loss

$$\min_{\beta} \sum_{i=1}^{n} \left[1 - y_i(\beta_0 + \beta^T x_i)\right]_+ + \frac{\lambda}{2}\|\beta\|_2^2$$

The dual problem is

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle$$
$$\text{s.t. } 0 \le \alpha_i \le C \quad \forall i = 1, \ldots, n$$

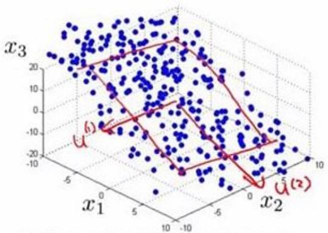
If we change the penalty term from $\|\cdot\|_2$ to $\|\cdot\|_{\mathcal{H}_K}$, then we just need to replace $\langle x_i, x_j\rangle$ with $k(x_i, x_j)$ in the dual problem.

Examples: Kernel PCA

▷ Recall that PCA projects the data to directions that retain most information/variation: $z^{(i)} = \tilde{P}^T x^{(i)}$, where $\tilde{P} \in \mathbb{R}^{n \times d}$ is the first d dominant eigenvectors of the covariance matrix $\Sigma = PDP^T$.

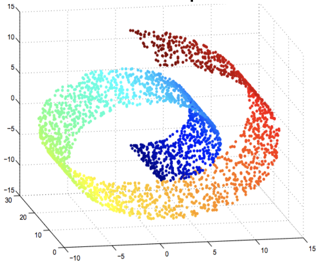
- Dimensionality reduction: $x \in \mathbb{R}^n \to z \in \mathbb{R}^d$
- Linear nature: projection onto a subspace is a *linear transformation*.
  Each "new" feature (i.e. entry in z) is a linear combination of the "old" features: $z_j = \tilde{P}_j^T x$, where $\tilde{P}_j$ is the j-th column of $\tilde{P}$.

▷ But what if the data are NOT distributed "close" to a subspace? [8]



Reduce data from 3D to 2D

V.S.



[8]Andrew Ng, Machine Learning Course; Janu Verma, Manifold Learning Blog

Examples: Kernel PCA

$\triangleright$ Solution: map x to high-dimensional feature space so that the transformed data are distributed close to a subspace, then do projection.

- Consider a feature map $\Phi : \mathbb{R}^n \to \mathbb{R}^d$, where $d \gg n$, d can be infinite.
- The covariance matrix of the transformed data is

$$\Sigma = \mathbb{E}\left[\Phi(x)\,\Phi(x)^T\right], \quad \hat{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}\Phi(x^{(i)})\,\Phi(x^{(i)})^T$$

We assume $\Phi(x^{(i)})$ have 0 mean and unit variance. If not, do $\mathbb{K}' = \mathbb{K} - 2\mathbf{1}_{1/N}\,\mathbb{K} + \mathbf{1}_{1/N}\,\mathbb{K}\,\mathbf{1}_{1/N}$, where $\mathbf{1}_{1/N}$ is filled with $1/N$.

Examples: Kernel PCA

$\triangleright$ Solution: map x to high-dimensional feature space so that the transformed data are distributed close to a subspace, then do projection.

- Consider a feature map $\Phi : \mathbb{R}^n \to \mathbb{R}^d$, where $d \gg n$, d can be infinite.
- The covariance matrix of the transformed data is

$$\Sigma = \mathbb{E}\left[\Phi(x)\,\Phi(x)^T\right], \quad \hat{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}\Phi(x^{(i)})\,\Phi(x^{(i)})^T$$

  We assume $\Phi(x^{(i)})$ have 0 mean and unit variance. If not, do $\mathbb{K}' = \mathbb{K} - 2\mathbf{1}_{1/N}\,\mathbb{K} + \mathbf{1}_{1/N}\,\mathbb{K}\,\mathbf{1}_{1/N}$, where $\mathbf{1}_{1/N}$ is filled with $1/N$.

- Let $\{v_j\}_{j=1}^d$ be the orthonormal eigenbasis. We need to find the projection of $\Phi(x^{(i)})$ onto each eigendirection, i.e. $v_j^T\Phi(x^{(i)})\ \forall\, i, j$
- A key claim: $\forall\, 1 \le j \le d$ s.t. $\lambda_j \neq 0$:

$$\exists\, \alpha_1, \ldots, \alpha_N \text{ s.t. } v_j = \textstyle\sum_{p=1}^N \alpha_p^{(j)}\Phi(x^{(p)})$$

  see Rita Osadchy's slides 12-13, the notations are different from mine

Examples: Kernel PCA

- Then $v_j^T \Phi(x^{(i)}) = \sum_{p=1}^N \alpha_p^{(j)} \Phi(x^{(p)})^T \Phi(x^{(i)}) = \sum_{p=1}^N \alpha_p^{(j)} \mathbb{K}_{p,i} = \mathbb{K}_i^T \alpha^{(j)}$
  Given a kernel matrix $\mathbb{K}$, it remains to find $\alpha^{(j)} = \left[\alpha_1^{(j)}, \ldots, \alpha_N^{(j)}\right]^T$ for the dominant eigendirections v_j, $j = 1, \ldots, \tilde{d}$.
- Write things in matrix form: Let $\Phi(X) = \begin{bmatrix} - & \Phi(x^{(1)})^T & - \\ & \vdots & \\ - & \Phi(x^{(N)})^T & - \end{bmatrix}$.
  We have $\hat{\Sigma} = \frac{1}{N} \Phi(X)^T \Phi(X)$, $\mathbb{K} = \Phi(X) \Phi(X)^T$, and
  $\mathbb{K}^T \alpha^{(j)} = \mathbb{K} \alpha^{(j)} = \Phi(X) v_j$.
- Left multiply $\Phi(X)$ to $\hat{\Sigma} v_j = \lambda_j v_j$, we get

$$\begin{aligned} \Phi(X) \hat{\Sigma} v_j &= \Phi(X) \lambda_j v_j \\ \frac{1}{N} \mathbb{K} \Phi(X) v_j &= \lambda_j \mathbb{K} \alpha^{(j)} \\ \mathbb{K}^2 \alpha^{(j)} &= N \lambda_j \mathbb{K} \alpha^{(j)} \end{aligned}$$

Examples: Kernel PCA

- When $\mathbb{K}$ is *full rank*, we can get $\mathbb{K}\,\alpha^{(j)} = N\,\lambda_j\,\alpha^{(j)}$.
  Note that $\operatorname{rank}(\Phi(X)) = \operatorname{rank}(\hat{\Sigma}) = \operatorname{rank}(\mathbb{K})$. If we ignore v_j's associated with $\lambda_j = 0$, then $\tilde{\Sigma} \in \mathbb{R}^{d\times\tilde{d}}$ is full rank.
- Thus, $\alpha^{(j)}$'s are the eigenvectors of the kernel matrix [9]. We need the dominant ones for projection.

Algorithm 1 Kernel PCA

1: Choose a kernel function $k(\cdot,\cdot)$
2: Evaluate $\mathbb{K}_{i,j} = k(x_i, x_j)$ using the data
3: "Normalize" the transformed data by $\mathbb{K}' = \mathbb{K} - 2\mathbf{1}_{1/N}\,\mathbb{K} + \mathbf{1}_{1/N}\,\mathbb{K}\,\mathbf{1}_{1/N}$
4: Find the first $\tilde{d}$ dominant eigenvectors $\alpha^{(1)}, \ldots \alpha^{(\tilde{d})}$ of $\mathbb{K}'$
5: Project the data onto $\operatorname{span}\{\alpha^{(1)}, \ldots \alpha^{(\tilde{d})}\}$ by calculating $z_j^{(i)} = \alpha_{(j)} x^{(i)}$ to get $z^{(i)} \in \mathbb{R}^{\tilde{d}}$ for dimensionality reduction

[9] See Rita Osadchy's slide 18

Kernel Mean Embedding

Consider a random variable X with PDF/PMF $P : \mathcal{X} \to \mathbb{R}$

▷ Let $x = \{x^{(1)}, \ldots, x^{(m)}\}$ be a random sample of X. The kernel mean embedding is

$$\mu_k(x) := \frac{1}{m} \sum_{i=1}^{m} k(x_i, \cdot) \in \mathcal{H}_K$$

▷ When $m \to \infty$, the kernel mean embedding of the distribution P is

$$\mu_k(P) := \int_{\mathcal{X}} P(x)\, k(x, \cdot)\, dx = \mathbb{E}_{X \sim P}\,[k(X, \cdot)] \in \mathcal{H}_K$$

- By reproducing property and linearity,

$$\forall\, f \in \mathcal{H}_K :\ \langle f, \mu_k(P) \rangle_{\mathcal{H}_K} = \mathbb{E}_{X \sim P}\,[f(X)]$$

- If k is *strictly positive definite* (**characteristic** kernel), then $\mu_k(\cdot)$ is an *injective* function from the space of probability distributions to $\mathcal{H}_K$.
  Namely, $P \neq Q \ \Rightarrow\ \mu_k(P) \neq \mu_k(Q)$. μ_k can be used to represent a distribution.

Maximum Mean Discrepancy (MMD)

$\triangleright$ Inspired by "moment matching", we want to use the following to represent how "close" 2 distributions are to each other.

$$\text{MMD} := \sup_{\|f\|_{\mathcal{H}_K}=1} \left| \mathbb{E}_{X\sim P}[f(X)] - \mathbb{E}_{Y\sim Q}[f(Y)] \right|$$

$\triangleright$ By reproducing property, the above quantity is

$$\sup_{\|f\|_{\mathcal{H}_K}=1} \left| \langle f, \mu_k(P) - \mu_k(Q) \rangle_{\mathcal{H}_K} \right|$$

$\triangleright$ By Cauchy-Schwarz inequality,

$$\left| \langle f, \mu_k(P) - \mu_k(Q) \rangle \right| \leq \|f\|_{\mathcal{H}_K} \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_K} = \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_K}$$

Equality is attained if and only if $f = \mu_k(P) - \mu_k(Q) / \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_k}$

$\triangleright$ We can squared the norm term to get and inner product and expand:

$$\text{MMD}^2 = \mathbb{E}_{X,X' \overset{i.i.d}{\sim} P}[k(X,X')] + \mathbb{E}_{Y,Y' \overset{i.i.d}{\sim} Q}[k(Y,Y')] - 2\mathbb{E}_{X\sim Q, Y\sim Q}[k(X,Y)]$$

References

▷ Dino Sejdinovic. UCL: Advanced Topics in Machine Learning - Theory of RKHS 2014. Lecture Notes: `http://www.stats.ox.ac.uk/~sejdinov/teaching/atml14/Theory_2014.pdf`

* *See a more recent version from Dino's Oxford course Advanced Topics in Statistical Machine Learning 2019*: `https://www.stats.ox.ac.uk/~sejdinov/teaching/atsml19/19_slides3.pdf`

▷ Sayan Mukherjee. Duke University STA 613 Statistical methods in computational biology 2018. Lecture Notes: `http://www2.stat.duke.edu/~sayan/Sta613/2018/lec/nonlin.pdf`

▷ Andrea Caponnetto. MIT 9.520/6.860 Statistical Learning Theory and Applications 2006. Lecture Slides: `https://www.mit.edu/~9.520/spring06/Classes/class03.pdf`

▷ Jian Zhang. Purdue University STAT 598Y Statistical Learning Theorey. Lecture Notes: `https://www.stat.purdue.edu/~jianzhan/STAT598Y/NOTES/slt12.pdf`

References

▷ Larry Wassermann. CMU 36-708 Statistical Machine Learning. Lecture Notes:
http://www.stat.cmu.edu/~larry/=sml/functionspaces.pdf

▷ Gatsby Unit. MLSS Tubingen 2015 Introduction to RKHS part 1.
http://mlss.tuebingen.mpg.de/2015/slides/gretton/part_1.pdf

▷ Rita Osadchy. University of Haifa, Mount Carmel Unsupervised Learning 2011. Lecture Slides: http://www.cs.haifa.ac.il/~rita/uml_course/lectures/KPCA.pdf