# Machine Learning Theory
# Tübingen University, WS 2016/2017
# Lecture 11

Tolstikhin Ilya

**Abstract**

We will introduce the notion of reproducing kernels and associated Reproducing Kernel Hilbert Spaces (RKHS). We will consider couple of easy examples to get some intuition. Next we will motivate an importance of RKHS for machine learning by considering representer theorem, which we will also prove. Finally, we will consider several scenarios where representer theorem actually becomes very useful. Blue colour will be used to highlight parts appearing in the upcoming homework assignments.

## 1 Reproducing kernels and RKHS

Consider any input space $\mathcal{X}$. We will call a function $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *a kernel* or a *reproducing kernel* if it is symmetric $k(x, y) = k(y, x)$ for all $x, y \in \mathcal{X}$ and positive definite, which means

$$\forall n \in \mathbb{N}, \quad \forall \alpha_1, \dots, \alpha_n \in \mathbb{R}, \quad \forall x_1, \dots, x_n \in \mathcal{X}, \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \geq 0.$$

It can be shown that $k$ defines a unique Hilbert space[1] $\mathcal{H}_k$ of real-valued functions on $\mathcal{X}$, such that:

1. Functions $k(X, \cdot)\colon \mathcal{X} \to \mathbb{R}$ for all $X \in \mathcal{X}$ belong to $\mathcal{H}_k$;
2. $f(X) = \langle f, k(X, \cdot) \rangle_{\mathcal{H}_k}$ for any $f \in \mathcal{H}_k$ and $X \in \mathcal{X}$. (*the reproducing property*)

Throughout this lecture we will write $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$ to denote the inner product of $\mathcal{H}_k$ and $\| \cdot \|_{\mathcal{H}_k}$ the norm induced by $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$.

The space $\mathcal{H}_k$ is commonly known as *Reproducing Kernel Hilbert Space* (RKHS). Notice that, because $\mathcal{H}_k$ is a vector space, all the functions of the form

$$\sum_i \alpha_i k(X_i, \cdot)$$

also belong to $\mathcal{H}_k$ for any finite sequence of real coefficients $\alpha_1, \alpha_2, \dots$ and points $X_1, X_2, \dots$ from $\mathcal{X}$.

[1] A Hilbert space is a vector space with an inner product, which is complete with respect to the norm induced by the inner product.

**Feature map** Another way to look at this construction is to say that all the points $X$ of the input space $\mathcal{X}$ are being mapped to the elements $k(X, \cdot)$ of the Hilbert space $\mathcal{H}_k$. Moreover, for any two points $X', X'' \in \mathcal{X}$ the inner product between their images is equal to $\langle k(X', \cdot), k(X'', \cdot)\rangle_{\mathcal{H}_k} = k(X', X'')$. This observation leads to very useful implications. It turns out that, no matter what the input space $\mathcal{X}$ is ($\mathbb{R}^d$, a set of strings, a set of graphs, png pictures, ...), once we come up with a kernel function $k$ defined over $\mathcal{X}$ we simultaneously get a way to embed the whole $\mathcal{X}$ into a Hilbert space $\mathcal{H}_k$. This embedding is very useful, since the RKHS has a very nice geometry: it is a vector space with an inner product, which means we can add its elements with each other and compute distances between them—something which was not necessarily possible for elements of $\mathcal{X}$ (think of a set of graphs).

Next we consider two simple examples of kernels $k$ and corresponding RKHS:

1. **Linear kernel** Consider $\mathcal{X} = \mathbb{R}^d$ and define $k(x, y) := \langle x, y\rangle_{\mathbb{R}^d}$. First of all, let's check that this is indeed a kernel. It is obviously symmetric. Also note that

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j k(x_i, x_j) = \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j\langle x_i, x_j\rangle_{\mathbb{R}^d} = \left\|\sum_{i=1}^{n}\alpha_i x_i\right\|_{\mathbb{R}^d}^2 \geq 0.$$

   Thus, $k$ is indeed a kernel. It is now easy to see that all the homogeneous linear functions of the form

$$f(x) = \langle w, x\rangle_{\mathbb{R}^d}, \quad w \in \mathbb{R}^d \tag{1}$$

   belong to the RKHS $\mathcal{H}_k$. As well as all their finite linear combinations. Actually, it can be shown that $\mathcal{H}_k$ does not contain anything but the functions of the form (1). In this case it is obvious that $\mathcal{H}_k$ is of a finite dimensionality $d$. The inner product in $\mathcal{H}_k$ between its two elements $\langle w, \cdot\rangle_{\mathbb{R}^d} \in \mathcal{H}_k$ and $\langle v, \cdot\rangle_{\mathbb{R}^d} \in \mathcal{H}_k$ (which are two linear functions) is defined by

$$\Big\langle \langle w, \cdot\rangle_{\mathbb{R}^d}, \langle v, \cdot\rangle_{\mathbb{R}^d} \Big\rangle_{\mathcal{H}_k} = \langle w, v\rangle_{\mathbb{R}^d}.$$

2. **Polynomial kernel of a second degree** Consider $\mathcal{X} = \mathbb{R}^2$ and $k(x, y) := (\langle x, y\rangle_{\mathbb{R}^2} + 1)^2$. Expanding the brackets we see:

$$k(x, y) = x_1^2y_1^2 + x_2^2y_2^2 + 2x_1x_2y_1y_2 + 2x_1y_1 + 2x_2y_2 + 1.$$

   First we need to check that it is indeed a kernel. It is symmetric. To check the positive definiteness note that if we define a mapping $\psi\colon \mathcal{X} \to \mathbb{R}^6$ by

$$\psi(x) = (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$$

   we may write

$$k(x, y) = \langle \psi(x), \psi(y)\rangle_{\mathbb{R}^6}, \quad \forall x, y \in \mathcal{X}.$$

   In other words, we showed that $k$ can be expressed as a linear kernel after mapping $\mathcal{X}$ into $\mathbb{R}^6$ using $\psi$. We already showed in the previous example that linear kernel is indeed positive definite. Interestingly notice that the image of $\psi$ is only a subset of $\mathbb{R}^6$, i.e. there are points $z \in \mathbb{R}^6$ such that $z$ can not be expressed as $\psi(x)$ for any $x \in \mathcal{X}$.

   Let us show that $\mathcal{H}_k$ contains all the polynomials up to degree 2, i.e. functions of the form:

$$f(x) = v_1x_1^2 + v_2x_2^2 + v_3x_1x_2 + v_4x_1 + v_5x_2 + v_6, \quad x \in \mathcal{X}, v \in \mathbb{R}^6. \tag{2}$$

First, we know that all the functions of the form $k(x, \cdot)$ belong to $\mathcal{H}_k$ for sure, i.e. all the functions of the form

$$f(x) = w_1^2 x_1^2 + w_2^2 x_2^2 + 2w_1 w_2 x_1 x_2 + 2w_1 x_1 + 2w_2 x_2 + 1, \quad x, w \in \mathcal{X}. \tag{3}$$

These are polynomials with monomials of order up to two. However, we see that coefficients of monomials are interdependent, and they are all defined by setting only two coefficients $w_1$ and $w_2$. This is quite different from (2), where we are free to choose any coefficients of monomials. However, recall that RKHS is a vector space, thus it contains all the linear combinations of its elements. Now, do we get all the functions of the form (2) if we take all the linear combinations of the functions of the form (3)? It turns out that if we take the linear span of the vectors of the form

$$\{(w_1^2, w_2^2, 2w_1 w_2, 2w_1, 2w_2, 1) \colon w_1, w_2 \in \mathbb{R}\} \subset \mathbb{R}^6$$

we will get the whole $\mathbb{R}^6$ (HW). This shows that indeed $\mathcal{H}_k$ contains all the polynomials up to degree 2. It can be also shown that no other functions are contained in $\mathcal{H}_k$.

Two examples above showed that RKHS can be of a finite dimension, which may or may not be larger than the dimensionality of $\mathcal{X}$. At this point it is important to say that actually RKHS can be even infinite dimensional. This is the case, for instance, for the so-called *Gaussian kernel* $k(x, y) = e^{-(x-y)^2/\sigma^2}$.

## 2 Representer theorem

Why are RKHS and kernels so important for machine learning? In all the previous lectures we studied problems of binary classification and also shortly mentioned regression problems. But what type of predictors did we actually see? It turns out that the main focus was on *linear* predictors. These functions (classifiers) are a good start, but of course they are not too flexible. We also saw an example of nonlinear methods, such as KNN. Note, however, that KNN can't be considered as a learning algorithm which chooses a predictor $\hat{h}_n$ from a fixed set of predictors $\mathcal{H}$. Finally, we saw the AdaBoost algorithm, which outputs a complex composition of base classifiers. This composition is of course not a linear classifier (even if the base classifiers were linear).

Kernels and RKHS provide a very convenient way to define classes $\mathcal{H}$ consisting of nonlinear functions. As we saw, it is enough to specify one kernel function $k$ to implicitly get the whole RKHS $\mathcal{H}_k$. Now, assume we would like to choose our predictors from $\mathcal{H}_k$. How do we do that? Next result shows that often this problem can be solved quite efficiently.

**Theorem 1** (Representer theorem)**.** *Assume $k$ is a kernel defined over any $\mathcal{X}$ and $\mathcal{H}_k$ is a corresponding RKHS. Take any points $X_1, \dots, X_n \in \mathcal{X}$. Consider the following optimization problem:*

$$\min_{f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} \ell_i\big(f(X_i)\big) + Q(\|f\|_{\mathcal{H}_k}), \tag{4}$$

*where $\ell_i \colon \mathbb{R} \to \mathbb{R}$, $i = 1, \dots, n$ are any functions and $Q \colon \mathbb{R}^+ \to \mathbb{R}$ is a nondecreasing. Then there exist $\alpha_1, \dots, \alpha_n \in \mathbb{R}$ such that*

$$f = \sum_{i=1}^{n} \alpha_i k(X_i, \cdot) \in \mathcal{H}_k$$

*solves* (4)*.*

*Proof.* Assume there is $f^* \in \mathcal{H}_k$ solving (4). Because $\mathcal{H}_k$ is a Hilbert space we may write

$$f^* = \sum_{i=1}^{n} \beta_i k(X_i, \cdot) + u,$$

where $u \in \mathcal{H}_k$, and $\langle u, k(X_i, \cdot)\rangle_{\mathcal{H}_k} = 0$ for all $i = 1, \ldots, n$. We used the fact that any vector (function) in a Hilbert space can be uniquely expressed as a sum of its orthogonal projection onto the linear subspace and a complement, which is orthogonal to that subspace. It is also easy to check that

$$\|f^*\|^2_{\mathcal{H}_k} = \left\| \sum_{i=1}^{n} \beta_i k(X_i, \cdot) \right\|^2_{\mathcal{H}_k} + \|u\|^2_{\mathcal{H}_k}$$

and thus

$$\|f^*\|_{\mathcal{H}_k} \geq \|f_X\|_{\mathcal{H}_k},$$

where we denoted

$$f_X := \sum_{i=1}^{n} \beta_i k(X_i, \cdot).$$

Because $Q$ is nondecreasing we conclude that $Q(\|f^*\|_{\mathcal{H}_k}) \geq Q(\|f_X\|_{\mathcal{H}_k})$. Now note that because of the reproducing property

$$\ell_i\big(f^*(X_i)\big) = \ell_i\big(\langle f^*, k(X_i, \cdot)\rangle_{\mathcal{H}_k}\big) = \ell_i\left(\langle f_X + u, k(X_i, \cdot)\rangle_{\mathcal{H}_k}\right) = \ell_i\left(\langle f_X, k(X_i, \cdot)\rangle_{\mathcal{H}_k}\right) = \ell_i\big(f_X(X_i)\big).$$

In other words we shoed that

$$\frac{1}{n}\sum_{i=1}^{n} \ell_i\big(f^*(X_i)\big) = \frac{1}{n}\sum_{i=1}^{n} \ell_i\big(f_X(X_i)\big).$$

Thus, the value of the objective functional (4) at $f_X$ is not larger than for $f^*$, which shows that $f_X$ also solves the optimization problem. □

In order to motivate representer theorem we will first consider two concrete examples of Problem 4.

**Binary classification** Can we use the real-valued functions from $\mathcal{H}_k$ for a binary classification with $\mathcal{Y} = \{-1, +1\}$? Of course! We just need to take the sign of $f \in \mathcal{H}_k$, which gives us a binary-valued function. Consider a training sample $S_n = \{(X_i, Y_i)\}_{i=1}^n$ with $X_i \in \mathcal{X}$ for any input space $\mathcal{X}$ and $Y_i \in \mathcal{Y}$. Take any kernel $k$ on $\mathcal{X}$. Finally, set $\ell_i(z) := 1\{Y_i \cdot z \leq 0\}$. In this case

$$\frac{1}{n}\sum_{i=1}^{n} \ell_i\big(f(X_i)\big) = \frac{1}{n}\sum_{i=1}^{n} 1\{Y_i \cdot f(X_i) \leq 0\}$$

is just an empirical binary loss associated with a classifier $\operatorname{sgn} f(x)$. Setting $Q(z) = 0$ we see that (4) corresponds to the empirical risk minimization of a binary loss over $\mathcal{H}_k$.

**Squared loss regression** We may also use elements of $\mathcal{H}_k$ for predicting real-valued outputs. Set $\mathcal{Y} = \mathbb{R}$ and $\ell_i(z) = (Y_i - z)^2$. In this case

$$\frac{1}{n}\sum_{i=1}^{n} \ell_i\big(f(X_i)\big) = \frac{1}{n}\sum_{i=1}^{n}\big(Y_i - f(X_i)\big)^2$$

is just an empirical squared loss and thus, setting $Q(z) = 0$ we get the empirical squared loss minimization over $\mathcal{H}_k$.

What is the importance of Theorem 1? A surprising message is the following. Originally, (4) is an optimization with respect to elements of $\mathcal{H}_k$, which are high-dimensional objects and potentially even infinite-dimensional. In other words, solving (4) requires choosing $m$ real numbers if $\mathcal{H}_k$ is $m$-dimensional (with $m$ potentially huge) or choosing *a function*, which can not be described by any finite number of parameters if $\mathcal{H}_k$ is infinite-dimensional. Still, Theorem 1 tells us that in any case this problem may be reduced to choosing only $n$ real-valued parameters. This gives a huge boost in efficiency if $\dim(\mathcal{H}_k) \gg n$, and especially if $\mathcal{H}_k$ is infinite-dimensional.

Using representer theorem and reproducing property we may restate the Problem 4 in the following form:

$$\begin{aligned}
&\min_{\alpha_1,\ldots,\alpha_n\in\mathbb{R}} \frac{1}{n}\sum_{i=1}^{n} \ell_i\left(\sum_{j=1}^{n}\alpha_j k(X_i,X_j)\right) + Q\left(\left\|\sum_{j=1}^{n}\alpha_j k(X_j,\cdot)\right\|_{\mathcal{H}_k}\right)\\
&= \min_{\alpha_1,\ldots,\alpha_n\in\mathbb{R}} \frac{1}{n}\sum_{i=1}^{n} \ell_i\left(\sum_{j=1}^{n}\alpha_j k(X_i,X_j)\right) + Q\left(\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j k(X_i,X_j)}\right).
\end{aligned}$$

We see that this optimization problem depends on $X_i$ and $k$ only through the *kernel matrix* $K_X \in \mathbb{R}^{n\times n}$ with $(i,j)$-th element being $k(X_i, X_j)$.