CPSC 532D — 10. KERNELS

Danica J. Sutherland

University of British Columbia, Vancouver

Fall 2023

We've mentioned a couple times the idea of implementing a polynomial classifier as a special case of a linear one: in $\mathbb{R}$, a cubic classifier might look like

$$h(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

where we have four parameters in w. Notice that we can also write this as

$$h(x) = w \cdot \phi(x), \qquad w \in \mathbb{R}^4,\ \phi(x) = (1, x, x^2, x^3).$$

As we saw last time, the SVM problem in particular allows us to solve problems of this form in two ways:

- by finding the coefficients w directly, operating on the data $\phi(x)$;
- by finding the dual variables α, looking only at inner products between data points given by

$$\phi(x) \cdot \phi(x') = 1 + xx' + (xx')^2 + (xx')^3.$$

Now, consider the set of all cubic functions

$$\mathcal{F}' = \{x \mapsto w \cdot \phi(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 : w \in \mathbb{R}^4\}.$$

We can re-parameterize this set as, for any $c > 0$,

$$\mathcal{F} = \{x \mapsto w \cdot \phi(x) = w_0 \sqrt{c^3} + w_1 \sqrt{3c^2} x + w_2 \sqrt{3c} x^2 + w_3 x^3 : w \in \mathbb{R}^4\};$$

this changes the meaning of w, but the set of available functions is the same. This form is helpful, though, in that for this different ϕ we have

$$\phi(x) \cdot \phi(x') = c^3 + 3c^2 xx' + 3c(xx')^2 + (xx')^3 = (xx' + c)^3.$$

So, computing dot products is now pretty simple, compared to operating in the explicit feature space. In higher dimensions, we can do the same thing; there'll be $\mathcal{O}(d^k)$ terms in the full version, since we'll need all kinds of interaction terms, but we can still parameterize the inner product as $(x \cdot x' + c)^k$.

We're going to call this a *kernel function*, which for general features ϕ will be

$$k(x, x') = \phi(x) \cdot \phi(x').$$

"Kernel" is a super-overloaded word. This is not *the same thing as in kernel density estimation, the kernel of a convolution, the kernel of a probability density, the kernel of a linear map, a CUDA kernel, an operating system kernel...*

1 DEFINING A FUNCTION SPACE

We're going to think of $\mathcal{F}$ as a vector space of functions. Let $f, f' \in \mathcal{F}$ correspond to weight vectors w, w'. Then we can let $f + f'$ be the function with weight vector

For more, visit https://cs.ubc.ca/~dsuth/532D/23w1/.

$w + w'$, and af that with weight vector af. This definition makes it a valid vector space.

Now, we're going to given $\mathcal{F}$ some even stronger structure: making it a (real) Hilbert space. To do this, define an inner product $\langle f, f' \rangle_{\mathcal{F}}$ by $w \cdot w'$, which also induces the norm $\|f\|_{\mathcal{F}} = \|w\|$. This satisfies the necessary linearity conditions and so on; the only thing left is to show that it's complete, meaning that all Cauchy sequences converge in $\mathcal{F}$; this will also be true.

It's worth emphasizing that while the $\mathcal{F}$ for each value of c, and $\mathcal{F}'$, are all the exact same set, $\|w\|$, and hence $\|f\|_{\mathcal{F}}$, is different between them. (Larger c will mean the lower-order coefficients can be smaller in order to express the same function, and so means that $\|f\|_{\mathcal{F}}$ is more determined by the coefficient on x^3.) This will be important when we use algorithms that depend on $\|f\|_{\mathcal{F}}$.

Now, let's do something slightly weird. Recall that

$$\phi(x) = (\sqrt{c^3}, \sqrt{c^2}x, \sqrt{c}x^2, x^3) \in \mathbb{R}^4.$$

Elements of $\mathcal{F}$ are functions corresponding to any $w \in \mathbb{R}^4$. So what happens if we think of the element of $\phi(x)$ as a weight vector for an element in $\mathcal{F}$? This would give us a function of the form

$$\begin{aligned} x' \mapsto & \sqrt{c^3}\sqrt{c^3} + \sqrt{3c^2}x\sqrt{3c^2}x' + \sqrt{3c}x\sqrt{3c}(x')^2 + x^3(x')^3 \\ &= c^3 + 3c^2xx' + 3c(xx')^2 + (xx')^3 \\ &= (xx' + c)^3 = k(x, x'). \end{aligned}$$

That is, if we evaluate the function with weights $\phi(x)$ at a point x', we just get the kernel back. There actually isn't any magic here at all; we defined $\mathcal{F}$ that way in the first place! Letting $\varphi(x) \in \mathcal{F}$ denote the function with weight vector $\phi(x) \in \mathbb{R}^4$, this means that

$$\langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}} = k(x, x').$$

Now, because it's a vector space, we know that $\sum_{i=1}^{n} \alpha_i \varphi(x_i) \in \mathcal{F}$ for any n, $\alpha_i \in \mathbb{R}$, and choice of x_i. By the linearity properties of inner product spaces,

$$\left\langle \sum_{i=1}^{n} \alpha_i \varphi(x_i), \varphi(x) \right\rangle_{\mathcal{F}} = \sum_{i=1}^{n} \alpha_i \langle \varphi(x_i), \varphi(x) \rangle_{\mathcal{F}} = \sum_{i=1}^{n} \alpha_i k(x_i, x).$$

Notice that, thinking of $\varphi(x_i)$ as a function from $\mathcal{X}$ to $\mathbb{R}$, this is the same as taking a linear combination of the functions, in terms of their pointwise evaluations.

So, we could think of $\mathcal{F}$ as having a vector space structure totally independent of w, where $af + f'$ is defined as the function $x \mapsto af(x) + f'(x)$, and where $f(x) = \langle f, \varphi(x) \rangle_{\mathcal{F}}$ (also known as the **reproducing property**) – at least for any f that's a linear combination of $\varphi(x_i)$ for some x_i. This will be the basis for our construction of a *reproducing kernel Hilbert space* (RKHS) for a generic kernel.

2 REPRODUCING KERNELS

Not every function can be a kernel: it needs to be possible to write as an inner product. So:

DEFINITION 1. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive definite kernel if and only if there exists some Hilbert space $\mathcal{F}$ and feature map $\phi : \mathcal{X} \to \mathcal{F}$ such that $k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{F}}$.

Notice that the space, and the map, don't need to be unique (e.g. you could always use $-\phi$ instead of ϕ). Sometimes it's clear what such a map is: for the cubic kernel we considered above, we used $\mathcal{F} = \mathbb{R}^4$ and $\phi(x) = (\sqrt{c^3}, \sqrt{3c^2}x, \sqrt{3c}x^2, x^3)$. Sometimes, though, it's not obvious for a given k whether there is such a map or not.

The definition implies that we need $k(x, x') = k(x', x)$, and that $k(x, x) \geq 0$. But those are only necessary, not sufficient.

THEOREM 2 ([Aro50]). *A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive definite kernel if and only if for all $m \geq 1$ and $x_1, \ldots, x_m \in \mathcal{X}$, the kernel matrix $\begin{bmatrix} k(x_1, x_1) & \ldots & k(x_1, x_m) \\ \vdots & \ddots & \vdots \\ k(x_m, x_1) & \ldots & k(x_m, x_m) \end{bmatrix} \in \mathbb{R}^{m \times m}$ is positive semi-definite.*

Unfortunately people are very inconsistent about terminology around positive definiteness. For matrices, "positive semi-definite" unambiguously means the eigenvalues are nonnegative, and "strictly positive definite" unambiguously means eigenvalues are all positive, but "positive definite" might mean either. Some people get annoyed if you try to say "positive semi-definite kernel function," though.

Recall that a positive semi-definite matrix can be equivalently characterized as:

- For all $\alpha \in \mathbb{R}^m$, $\alpha^\mathsf{T} \mathrm{K} \alpha \geq 0$.
- All eigenvalues of K are nonnegative.
- $\mathrm{K} = \mathrm{L}\mathrm{L}^\mathsf{T}$ for some $\mathrm{L} \in \mathbb{R}^{m \times m}$.

Proof (sketch). One direction is easy: if $k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}}$, then

$$\alpha^\mathsf{T} \mathrm{K} \alpha = \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \langle \phi(x_i), \phi(x_j) \rangle_{\mathcal{F}} \alpha_j = \left\| \sum_{i=1}^{m} \alpha_i \phi(x_i) \right\|_{\mathcal{F}}^2 \geq 0.$$

To show the other direction, given a k satisfying this property, we'll construct a space $\mathcal{F}$: the reproducing kernel Hilbert space.

We'll start by building a "pre-Hilbert space" $\mathcal{F}_0$, containing functions $\mathcal{X} \to \mathbb{R}$. Start by defining the functions $\varphi(x) = [x' \mapsto k(x, x')]$ for all x. Then, let $\mathcal{F}_0$ be the set of all linear combinations of these functions, $\sum_{i=1}^{m} \alpha_i \varphi(x_i)$ for any $m \geq 0$, $x_1, \ldots, x_m \in \mathcal{X}$, $\alpha_1, \ldots, \alpha_m \in \mathbb{R}$. Define an inner product by

$$\left\langle \sum_{i=1}^{m} \alpha_i \varphi(x_i), \sum_{j=1}^{n} \beta_j \varphi(x'_j) \right\rangle_{\mathcal{F}_0} = \sum_{i=1}^{m} \sum_{j=1}^{n} k(x_i, x'_j).$$

This satisfies the required linearity and nonnegativity properties to be an inner product. It also has the reproducing properties that we expect:

$$\langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}_0} = k(x, x') \qquad \langle f, \varphi(x) \rangle_{\mathcal{F}_0} = f(x).$$

Notice also that this is well-defined in the sense that it's representation-independent:

$$\left\langle \sum_{i=1}^{m} \alpha_i \varphi(x_i), f' \right\rangle_{\mathcal{F}_0} = \sum_{i=1}^{m} \alpha_i \langle \varphi(x_i), f' \rangle_{\mathcal{F}_0} = \sum_{i=1}^{m} \alpha_i f'(x_i),$$

which doesn't depend on how we wrote f' as a linear combination, just on its values.

The only thing left is that we need $\mathcal{F}_0$ to be complete: it's conceivable that not all Cauchy sequences have limits in this space. So, we construct the RKHS as the completion of $\mathcal{F}_0$: just add the limits in, defining their inner products as limits of the inner products of the sequence (which is guaranteed to exist since the sequence is Cauchy). So, not all $f \in \mathcal{F}$ can be written as $\sum_{i=1}^{n} \alpha_i \varphi(x_i)$, but you can always get arbitrarily close (in the distance defined by $\|\cdot\|_{\mathcal{F}}$) to f with things of that form.

After checking all the details work out, we've constructed a Hilbert space and a feature map for any k. □

(There are also other ways to define an RKHS; it turns out each RKHS has a unique kernel, and each kernel has a unique RKHS, though there could be more than Hilbert space aligning with the definition.)

2.1 *Special case: linear kernel*

If we use $k(x, x') = x \cdot x'$ for $x \in \mathbb{R}^d$, then $\varphi(x) = [x' \mapsto x' \cdot x]$ is just a linear function with weight x. Also,

$$\left\|\varphi(x)\right\|_{\mathcal{F}} = \sqrt{\langle \varphi(x), \varphi(x) \rangle_{\mathcal{F}}} = \sqrt{k(x,x)} = \|x\|.$$

So everything we've done with linear predictors can be thought of as operating in the RKHS corresponding to a linear kernel. This is often a useful thing to think about if you're looking at some complicated kernel expression: see what it'd be with a linear kernel.

3 OPTIMIZING IN THE RKHS

Theorem 3 (Representer theorem). *If $\mathcal{F}$ is an RKHS with feature map φ, then for any function $\mathrm{L} : \mathbb{R}^m \to \mathbb{R}$ and any nondecreasing function $\mathrm{R} : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$,*

$$\operatorname*{arg\,min}_{f \in \mathcal{F}} \mathrm{L}(f(x_1), \dots, f(x_m)) + \mathrm{R}(\|f\|)$$

contains a solution of the form $f = \sum_{i=1}^{m} \alpha_i \varphi(x_i)$, where $\mathrm{S} = (x_1, \dots, x_m)$. If R is strictly increasing, all solutions are of this form.

Notice that $\operatorname*{arg\,min}_{f : \|f\|_{\mathcal{F}} \le \mathrm{B}} \mathrm{L}_{\mathrm{S}}(f)$ fits this form: use $\mathrm{R}(t) = \begin{cases} 0 & t \le \mathrm{B} \\ \infty & t > \mathrm{B} \end{cases}$.

Proof. Let $\mathcal{F}_{\|}$ be the subspace of $\mathcal{F}$ spanned by $\{\varphi(x_i)\}_{i=1}^m$, and $\mathcal{F}_{\perp}$ its orthogonal complement. Then any element of $\mathcal{F}$ can be uniquely decomposed into $f_{\|} + f_{\perp}$, where $f_{\|} \in \mathcal{F}_{\|}$, $f_{\perp} \in \mathcal{F}_{\perp}$, and $\langle f_{\|}, f_{\perp} \rangle_{\mathcal{F}} = 0$. Now, since

$$f(x_i) = \langle f, \varphi(x_i) \rangle_{\mathcal{F}} = \langle f_{\|} + f_{\perp}, \varphi(x_i) \rangle_{\mathcal{F}} = \langle f_{\|}, \varphi(x_i) \rangle_{\mathcal{F}} + \underbrace{\langle f_{\perp}, \varphi(x_i) \rangle_{\mathcal{F}}}_{0},$$

the L component only depends on $f_{\|}$. Also,

$$\|f\|_{\mathcal{F}}^2 = \left\|f_{\|}\right\|_{\mathcal{F}}^2 + \|f_{\perp}\|_{\mathcal{F}}^2 + 2\underbrace{\langle f_{\|}, f_{\perp} \rangle_{\mathcal{F}}}_{0} = \left\|f_{\|}\right\|_{\mathcal{F}}^2 + \|f_{\perp}\|_{\mathcal{F}}^2.$$

Thus, having a nonzero value of $f_\perp$ does not change L, and cannot help R. If R is strictly increasing, it can only hurt the overall objective. □

This means that the form $w = \sum_i \alpha_i \varphi(x_i)$ that we got from SVM duality wasn't just a coincidence: *any* problem will have a solution of that form. But this allows us to reduce optimization in $\mathcal{F}$ – potentially infinite-dimensional – to optimization over $\alpha \in \mathbb{R}^m$.

This α_i is slightly different than we used in the SVM; there we had $\alpha_i y_i$ with $\alpha_i \geq 0$, whereas here we just have a generic $\alpha_i \in \mathbb{R}$.

3.1 *Example: kernel ridge regression*

Consider the problem

$$\min_{h \in \mathcal{F}} \mathrm{L}_\mathrm{S}^{sq}(h) + \lambda \|h\|_\mathcal{F}^2 \tag{1}$$

for $\lambda > 0$. First off, with a linear kernel, this becomes just plain ridge regression $\min_w \mathrm{L}_\mathrm{S}^{sq}(x \mapsto w \cdot x) + \lambda \|w\|^2$.

We know that all solutions will be of the form $\sum_{i=1}^m \alpha_i \varphi(x_i)$, so (1) is equivalent to

$$\min_{\alpha \in \mathbb{R}^m} \mathrm{L}_\mathrm{S}^{sq}\left(\sum_i \alpha_i \varphi(x_i)\right) + \lambda \left\|\sum_i \alpha_i \varphi(x_i)\right\|_\mathcal{F}^2 . \tag{2}$$

The second term here is just

$$\left\|\sum_i \alpha_i \varphi(x_i)\right\|_\mathcal{F}^2 = \sum_{i,j} \alpha_i k(x_i, x_j) \alpha_j = \alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} \alpha,$$

where $\mathrm{K}|_{\mathrm{S}_x} \in \mathbb{R}^{m \times m}$ is the kernel matrix on S_x, as in Theorem 2. For the first term, notice that

$$\sum_i \alpha_i k(x_i, x_j) = \alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} e_j$$

where $e_j \in \mathbb{R}^m$ is the jth standard basis vector. Then

$$\mathrm{L}_\mathrm{S}^{sq}\left(\sum_i \alpha_i \varphi(x_i)\right) = \frac{1}{m} \sum_i \left(\alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} e_i - y_i\right)^2 = \frac{1}{m} \left\|\mathrm{K}\alpha - y\right\|_{\mathbb{R}^m}^2 .$$

Thus the overall problem is

$$\begin{aligned}\hat{\alpha} &\in \arg\min_\alpha \frac{1}{m} \alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} \mathrm{K}|_{\mathrm{S}_x} \alpha - \frac{2}{m} y^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} \alpha + \frac{1}{m} y^\mathsf{T} y + \lambda \alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} \alpha \\ &= \arg\min_\alpha \alpha^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} (\mathrm{K}|_{\mathrm{S}_x} + m\lambda \mathrm{I}) \alpha - 2 y^\mathsf{T} \mathrm{K}|_{\mathrm{S}_x} \alpha.\end{aligned}$$

Setting the gradient to zero gives that we want

$$\mathrm{K}|_{\mathrm{S}_x} (\mathrm{K}|_{\mathrm{S}_x} + m\lambda \mathrm{I}) \alpha = \mathrm{K}|_{\mathrm{S}_x} y,$$

which is achieved by

$$\hat{\alpha} = (\mathrm{K}|_{\mathrm{S}_x} + m\lambda \mathrm{I})^{-1} y.$$

When $\lambda > 0$ this inverse is guaranteed to exist, since $\mathrm{K}|_{\mathrm{S}_x}$ is positive semidefinite, so $\mathrm{K}|_{\mathrm{S}_x} + m\lambda$ has all eigenvalues at least $m\lambda$.

We can also make predictions on an arbitrary test point with

$$\langle \sum_i \hat{\alpha}_i \varphi(x_i), \varphi(x) \rangle_{\mathcal{F}} = \sum_i \hat{\alpha}_i k(x_i, x) = \hat{\alpha} \cdot \begin{bmatrix} k(x_1, x) \\ \vdots \\ k(x_m, x) \end{bmatrix}.$$

Unlike SVMs, we don't in general expect $\hat{\alpha}$ to be sparse.

It's worth checking for yourself that this agrees with standard ridge regression. (You might have to use the Woodbury matrix identity to line them up, since usual expressions for ridge regression invert a $d \times d$ matrix instead of an $m \times m$ one. In 340, we called this version the "other normal equations.")

People sometimes call this transformed version a dual form, especially e.g. for kernel ridge regression. While "dual" isn't necessarily a strictly defined term, note that it's not *a Lagrange dual.*

3.2 *Other problems*

We often won't be able to solve things in closed form like we can for kernel ridge regression. But the representer theorem will still be helpful for any problem of the right form; we just still might have to run an optimization algorithm like gradient descent on the α variables. This allows you to, for example, run kernel SVMs in the "primal"; you'll still have m variables, but it'll be a minimization of the hinge loss objective instead of a maximization of the Lagrange dual.

4 Other kernels

The most common kernel people use is the Gaussian kernel, also called the "square exponential" or "exponentiated quadratic" by some communities:

$$k(x, x') = \exp\left(-\frac{1}{2\sigma^2} \left\|x - x'\right\|^2\right).$$

My preferred way to prove this is a kernel goes through the following construction:

proposition 4. *Let $k, k_1, k_2, \ldots$ be positive definite kernels on $\mathcal{X}$. Then the following are all also positive definite kernels:*

1. $\gamma k = (x, x') \mapsto \gamma k(x, x')$ *for any* $\gamma > 0$.
2. $k_1 + k_2 = (x, x') \mapsto k_1(x, x') + k_2(x, x')$.
3. $k_1 k_2 = (x, x') \mapsto k_1(x, x') k_2(x, x')$.
4. $k^n = (x, x') \mapsto k(x, x')^n$ *for any nonnegative integer* n.
5. $k_\infty = (x, x') \mapsto \lim_{n\to\infty} k_n(x, x')$, *when the limit always exists.*
6. $e^k = (x, x') \mapsto \exp(k(x, x'))$.
7. $(x, x') \mapsto f(x) k(x, x') f(x')$ *for any function* $f : \mathcal{X} \to \mathbb{R}$.
8. $(x, x') \mapsto k'(f(x), f(x'))$ *for any function* $f : \mathcal{X} \to \mathcal{X}'$ *and* k' *a kernel on* $\mathcal{X}'$.

Proof. Let $\varphi, \varphi_1, \varphi_2, \ldots$ be the feature maps for these kernels, and $\mathrm{K}, \mathrm{K}_1, \mathrm{K}_2, \ldots$ the kernel matrices for arbitrary $(x_1, \ldots, x_m) \in \mathcal{X}^m$.

1. Use the feature map $x \mapsto \sqrt{\gamma}\phi$.
2. $\alpha^\mathsf{T}(\mathrm{K}_1 + \mathrm{K}_2)\alpha = \alpha^\mathsf{T}\mathrm{K}_1\alpha + \alpha^\mathsf{T}\mathrm{K}_2\alpha \geq 0$.

3. This is called the Schur product theorem. Define independent multivariate normal random vectors $V \sim \mathcal{N}(0, K_1)$ and $W \sim \mathcal{N}(0, K_2)$. Let $V \odot W$ be the elementwise product of V and W; this has covariance matrix $K_1 \odot K_2$, and covariances must be psd.
4. Iteratively apply the previous property; also, k^0 has feature map $x \mapsto 1$.
5. $\alpha^T K_\infty \alpha = \alpha^T [\lim_{n\to\infty} K_n]\alpha = \lim_{n\to\infty} \alpha^T K_n \alpha \geq 0$.
6. Use $\exp(k(x, x')) = \lim_{N\to\infty} \sum_{n=0}^{N} \frac{1}{n!} k(x, x')^n$ and the previous properties.
7. Use the feature map $x \mapsto f(x)\varphi(x)$.
8. Use the feature map $x \mapsto \varphi'(f(x))$. □

To get the Gaussian kernel, notice that

$$\exp\left(-\frac{1}{2\sigma^2}\left\|x - x'\right\|^2\right) = \exp\left(-\frac{1}{2\sigma^2}\|x\|^2\right)\exp\left(\frac{1}{\sigma^2}x \cdot x'\right)\exp\left(-\frac{1}{2\sigma^2}\left\|x'\right\|^2\right)$$

and apply the properties above.

The Gaussian is *not* always the best kernel, particularly in high dimensions. Functions in $\mathcal{F}$ for a Gaussian kernel are very smooth; the Matérn kernel is preferred in some settings where rougher functions are expected. Another good general-purpose kernel is the *distance kernel* [SSGF13]

$$k(x, x') = \rho(x, O) + \rho(x', O) - \rho(x, x')$$

where ρ is a (semi)metric, and $O \in \mathcal{X}$ is some arbitrary center point.

If you have a good (e.g. deep) feature extractor ψ, using a kernel of the form $k(\psi(x), \psi(x'))$ can often be a good idea.

4.1 *Some properties*

PROPOSITION 5. *Let $f \in \mathcal{F}$, the RKHS with kernel k. Then*

$$|f(x)| \leq \|f\|_{\mathcal{F}} \sqrt{k(x,x)} \qquad \left|f(x) - f(x')\right| \leq \|f\|_{\mathcal{F}} \sqrt{k(x,x) + k(x',x') - 2k(x,x')}.$$

Proof. We have by the representer property and Cauchy-Schwartz that

$$|f(x)| = \left|\langle f, \varphi(x)\rangle_{\mathcal{F}}\right| \leq \|f\|_{\mathcal{F}} \left\|\varphi(x)\right\|_{\mathcal{F}}.$$

Similarly,

$$\left|f(x) - f(x')\right| = \left|\langle f, \varphi(x) - \varphi(x')\rangle_{\mathcal{F}}\right| \leq \|f\|_{\mathcal{F}} \sqrt{k(x,x) + k(x',x') - 2k(x,x')}.$$

□

Many more properties of this kind are available. For *shift-invariant* kernels, $k(x, x') = \kappa(x - x')$, a lot is available via Fourier properties of κ.

We've only scratched the surface here. We'll touch on kernels again through the rest of the course, but if you want more, Chapter 7 of [Bach23] goes in some more depth, and [SC08] is a classic very deep/mathematically thorough reference. Bayesian-oriented people might also want to see connections to Gaussian Processes [RW06; KHSS18], which are very much "almost the same thing" from a slightly different point of view. There's also the kernels reading group we're starting! :)

REFERENCES

[Aro50] Nachman Aronszajn. "Theory of Reproducing Kernels." *Transactions of the American Mathematical Society* 68.3 (May 1950), pages 337–404.

[Bach23] Francis Bach. *Learning Theory from First Principles*. April 2023 draft.

[KHSS18] Motonobu Kanagawa, Philipp Hennig, Dino Sejdinovic, and Bharath K Sriperumbudur. *Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences*. 2018. arXiv: 1807.02582.

[RW06] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[SC08] Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer, 2008.

[SSGF13] Dino Sejdinovic, Bharath K. Sriperumbudur, Arthur Gretton, and Kenji Fukumizu. "Equivalence of distance-based and RKHS-based statistics in hypothesis testing." *Annals of Statistics* 41.5 (Oct. 2013), pages 2263–2291.