

CSE 6740 Lecture 14

How Can I Learn Fancier (Nonlinear) Models? (Kernelization)

Alexander Gray
agray@cc.gatech.edu

Georgia Institute of Technology

Today

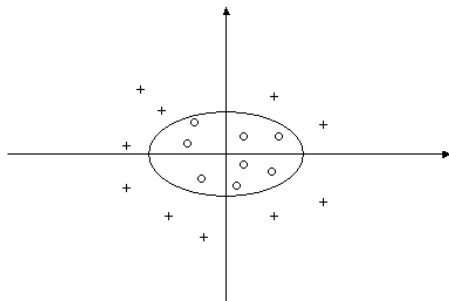
- ① How to make more complex models using kernels
- ② Theory motivating kernels

More Complex Models, Using Kernels

Why kernels, part I. “Because we can get richer models, yet leave the methods the same.”

Generalized Linear Models

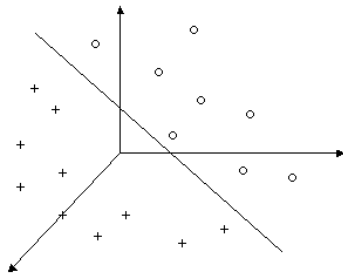
Suppose we have data $\{(x, y)\}_i$ where each $x \in \mathcal{X} = \mathbb{R}^2$ is a vector (x_1, x_2) like the following. Then the classes cannot be separated by a linear decision boundary.



Generalized Linear Models

Now let's make a transformed dataset $\{(z, y)\}_i$ where

$$z = \phi(x) = (x_1^2, \sqrt{2}x_1x_2, x_2^2). \quad (1)$$



Generalized Linear Models

Thus ϕ is a map from $\mathcal{X} = \mathbb{R}^2$ to $\mathcal{Z} = \mathbb{R}^3$. In the new space, the data are linearly separable.

So here a linear classifier in a higher-dimensional space corresponds to a nonlinear classifier in the original space.

Thus we get to leave our learning method exactly as it was.

Generalized Linear Models

We can do this with any linear model, including for example linear regression, where this is called *generalized linear regression*, to effectively get a more powerful model class.

However, there are drawbacks to this. We don't know in advance which features need to be constructed. Thus we might want to consider all possible products of the features, for example. But even considering all possible quadruplets of features, if $D=256$, yields 183,181,376 features in the transformed space.

Kernel Trick

Now suppose we have a model that can be represented in terms of only dot products between points, $\langle x, \tilde{x} \rangle$. Now notice that the inner product in \mathcal{Z} can be written

$$\langle z, \tilde{z} \rangle = \langle \phi(x), \phi(\tilde{x}) \rangle \quad (2)$$

$$= x_1^2 \tilde{x}_1^2 + 2x_1 \tilde{x}_1 x_2 \tilde{x}_2 + x_2^2 \tilde{x}_2^2 \quad (3)$$

$$= (\langle x, \tilde{x} \rangle)^2 \quad (4)$$

$$\equiv K(x, \tilde{x}). \quad (5)$$

Thus we can compute $\langle z, \tilde{z} \rangle$ without ever actually computing $z_i = \phi(x_i)$.

Kernel Trick

In summary, this so-called *kernel trick* involves finding a mapping $\phi : \mathcal{X} \rightarrow \mathcal{Z}$ and a learning method such that

- \mathcal{Z} has higher dimension than \mathcal{X} , effectively leading to a richer set of models
- The method's algorithm only requires computing inner products, as far as how the data appear in the method
- There is a kernel function K such that $\langle \phi(x), \phi(\tilde{x}) \rangle = K(x, \tilde{x})$.
- Everywhere the term $\langle x, \tilde{x} \rangle$ appears in the algorithm, replace it with $K(x, \tilde{x})$.

This is also called *kernelizing* the original method.

Kernel Trick

In fact, we never need to explicitly construct the mapping ϕ at all. We only need to specify a kernel $K(x, \tilde{x})$ that corresponds to $\langle \phi(x)\phi(\tilde{x}) \rangle$ for some ϕ .

One question of interest, then, is the following: Given a function of two variables $K(x, \tilde{x})$, does there exist a function $\phi(x)$ such that $K(x, \tilde{x}) = \langle \phi(x)\phi(\tilde{x}) \rangle$?

Mercer's Theorem

Mercer's Theorem says, roughly, that if K is positive definite, *i.e.*

$$\int \int K(x, \tilde{x}) g(x) g(\tilde{x}) dx d\tilde{x} \geq 0 \quad (6)$$

for square integrable functions g , *i.e.*

$$\int g^2(x) dx < \infty \quad (7)$$

then such a ϕ exists for K .

Mercer Kernels

Some commonly used kernels having this property, or *Mercer kernels*, are:

$$\text{Gaussian } K(x, \tilde{x}) = \exp \left\{ -\|x - \tilde{x}\|^2 / 2a^2 \right\} \quad (8)$$

$$\text{polynomial } K(x, \tilde{x}) = (\langle x, \tilde{x} \rangle + a)^b \quad (9)$$

$$\text{sigmoid } K(x, \tilde{x}) = \tanh(a\langle x, \tilde{x} \rangle + b) \quad (10)$$

There are also Mercer kernels for comparing non-vector objects, like strings and graphs.

The *kernel matrix* or *Gram matrix* \mathbf{K} is the $N \times N$ matrix having entries $\mathbf{K}_{ij} = K(x_i, x_j)$.

Kernelizing the SVM

The support vector machine is a method whose algorithm contains the data only in terms of their dot products with each other. Thus we can replace the objective function

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{i'=1}^N \alpha_i \alpha_{i'} y_i y_{i'} \langle x_i, x_{i'} \rangle \quad (11)$$

with

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{i'=1}^N \alpha_i \alpha_{i'} y_i y_{i'} K(x_i, x_{i'}). \quad (12)$$

Before the discriminant function was $g(x) = \beta^T x + \beta_0$. Now it is $g(x) = \beta^T \phi(x) + \beta_0$.

Theory Motivating Kernels

Why kernels, part II. “Because everything boils down to kernels.”

Regularization Theory

Consider minimizing a regularized loss function:

$$\min_{f \in \mathcal{F}} \left[\sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f) \right] \quad (13)$$

where $J(f)$ is a penalty functional, and \mathcal{F} is the space of functions on which $J(f)$ is defined.

Regularization Theory

Consider a general class of penalty functionals having the form

$$J(f) = \int_{\mathbb{R}^D} \frac{|\check{f}(s)|^2}{\check{K}(s)} ds \quad (14)$$

where \check{f} denotes the Fourier transform of f , and \check{K} is some positive function which falls off to zero as $\|s\| \rightarrow \infty$. The idea is that $1/\check{K}$ increases the penalty for high-frequency components of f .

Regularization Theory

It can be shown under certain assumptions that the solutions have the form

$$f(x) = \sum_{i=1}^N \alpha_i K(\|x - x_i\|) + \sum_{j=1}^M \beta_j \psi_j(x) \quad (15)$$

where the ψ_j span the null space of the penalty functional J , and K is the inverse Fourier transform of \check{K} .

Interestingly:

- While the minimization is over an infinite space, the solution is finite-dimensional.
- The solution has the form of a weighted sum of kernel functions.

Reproducing Kernel Hilbert Spaces

Consider a subclass of this kind of regularization problem, corresponding to certain kernel functions K . The penalty functional J is defined in terms of the kernel also.

The corresponding space of functions \mathcal{F}_K is called a *reproducing kernel Hilbert space*.

Suppose that K has an eigen-expansion

$$K(x, \tilde{x}) = \sum_{j=1}^{\infty} \gamma_j \psi_j(x) \psi_j(\tilde{x}) \quad (16)$$

with $\gamma_j \geq 0$, $\sum_{j=1}^{\infty} \gamma_j^2 < \infty$.

Reproducing Kernel Hilbert Spaces

Elements of \mathcal{F}_K have an expansion in terms of these eigen-functions,

$$f(x) = \sum_{j=1}^{\infty} \beta_j \psi_j(x) \quad (17)$$

with the constraint that

$$\|f\|_{\mathcal{F}_K}^2 \equiv \sum_{j=1}^{\infty} \beta_j^2 / \gamma_j < \infty \quad (18)$$

where $\|f\|_{\mathcal{F}_K}$ is called the norm induced by K . The penalty functional for the space \mathcal{F}_K is defined to be the squared norm $J(f) = \|f\|_{\mathcal{F}_K}^2$. This penalizes functions with large eigenvalues more.

Reproducing Kernel Hilbert Spaces

Thus we are talking about a special case of regularized loss minimization:

$$\min_{f \in \mathcal{F}} \left[\sum_{i=1}^N L(y_i, f(x_i)) + \lambda \|f\|_{\mathcal{F}_K}^2 \right] \quad (19)$$

$$= \min_{\{\beta_j\}} \left[\sum_{i=1}^N L \left(y_i, \sum_{j=1}^{\infty} \beta_j \psi_j(x) \right) + \lambda \sum_{j=1}^{\infty} \beta_j^2 / \gamma_j \right]. \quad (20)$$

It can be shown that the solution is finite-dimensional, having the form

$$f(x) = \sum_{i=1}^N \alpha_i K(x, x_i). \quad (21)$$

RKHS: Representing and Reproducing

The kernel as a function of its first argument

$$\phi_i(x) \equiv K(x, x_i) \quad (22)$$

is called the *representer of evaluation* at x_i in \mathcal{F}_K , since for $f \in \mathcal{F}_K$, it is easily seen that

$$\langle K(\cdot, x_i), f \rangle_{\mathcal{F}_K} = f(x_i). \quad (23)$$

Similarly,

$$\langle K(\cdot, x_i), K(\cdot, x_{i'}) \rangle_{\mathcal{F}_K} = K(x_i, x_{i'}), \quad (24)$$

which is called the *reproducing* property of \mathcal{F}_K .

RKHS: Minimization Problem

It is a consequence that

$$J(f) = \sum_{i=1}^N \sum_{i'=1}^N K(x_i, x_{i'}) \alpha_i \alpha_{i'} \quad (25)$$

for $f(x) = \sum_{i=1}^N \alpha_i K(x, x_i)$.

Our minimization problem then has the form

$$\min_{\alpha} L(\mathbf{y}, \mathbf{K}\alpha) + \lambda \alpha^T \mathbf{K}\alpha. \quad (26)$$

The support vector machine's optimization problem is equivalent to

$$\min_{\alpha} \left[\sum_{i=1}^N (1 - y_i g(x_i))_+ + \lambda \|\beta\|^2 \right] \quad (27)$$

$$= \min_{\alpha} \left[\sum_{i=1}^N (1 - y_i g(x_i))_+ + \lambda \alpha^T \mathbf{K} \alpha \right]. \quad (28)$$

We can kernelize linear regression, obtaining something called *kriging*, which can be reinterpreted slightly in a Bayesian way as something called *Gaussian process regression*.

Main Things You Should Know

- What the kernel trick (or kernelization) is
- What the point of the kernel trick is

Quiz

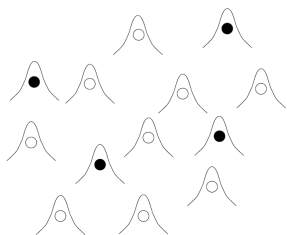
- ① (T/F) The kernel trick can be applied to any machine learning method.
- ② (T/F) Any similarity function is a kernel.

Now that we have kernels, what can we do with these things?

Suppose for each $x \in \mathcal{X}$ we have a feature map ϕ such that $\phi(x) \in \mathcal{F}_K$.

Further, we have

$$\langle \phi(x), \phi(x') \rangle_{\mathcal{F}_K} = k(x, x') \in \mathbb{R}.$$



The feature space for a Gaussian kernel

Statistical Independence

Suppose we have joint observations $(x, y) \in \mathcal{X} \times \mathcal{Y}$.

How might we test statistical independence of the 2 random variables?

- Correlation: $\mathbb{E}[xy] = \mathbb{E}[x]\mathbb{E}[y]$

This is fine for $x, y \sim$ Gaussian but is not a measure of independence.

Note: The principal components induce Gaussian random variables. We only are guaranteed uncorrelatedness, **not statistical independence!**

Statistical independence is much stronger:

- $\mathbb{E}[f(x)g(y)] = \mathbb{E}[f(x)]\mathbb{E}[g(y)] \quad (\text{almost})\forall f, g$

Properties of Measures of Independence

Let $Q(P_{x,y})$ be a measure of statistical independence.

From Rényi, we consider 2 principles:

- ① $Q(P_{x,y})$ is well-defined.
- ② $Q(P_{x,y}) = 0$ if and only if x, y independent.

Can we find a measure satisfying these properties?

Let's revisit a well known property of statistically independent random variables:

$$P_{x,y} = P_x \times P_y \\ \Rightarrow p(x, y) = p(x)p(y) \quad \forall (x, y) \in \mathcal{X} \times \mathcal{Y}$$

This motivates a measure of difference between $P_{x,y}$ and $P_x \times P_y$.
But for this, we need some **distance measure on distributions**.

Mean elements in \mathcal{F}

Kernels can provide us with a suitable distance measure.

Consider the expectation in the RKHS $\mu[P_x] = \mathbb{E}_x[k(x, \cdot)]$.

The empirical quantity is just $\mu[X] = \frac{1}{m} \sum_{i=1}^m k(x_i, \cdot)$.

We can evaluate the inner product of the mean with $f \in \mathcal{F}$ as

$$\langle \mu[P_x], f \rangle = \langle \mathbb{E}_x[k(x, \cdot)], f \rangle = \mathbb{E}_x[f(x)].$$

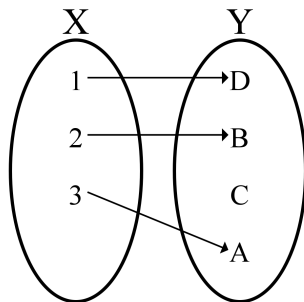
Likewise, for the population statistic, we have

$$\langle \mu[X], f \rangle = \frac{1}{m} \sum_{i=1}^m f(x_i).$$

Universal kernels and injective property

If the kernel k is universal, then the mean map $\mu : P_X \mapsto \mu[P_X]$ is injective.

Different distributions map to different points in the kernel space!



This informs a test of statistical independence:

- Compare the joint distribution of x, y to the product of their marginal distributions.

Mean elements for joints and marginals

The mean element for the joint

$$\mu[P_{xy}] = \mathbb{E}_{x,y}[v((x, y), \cdot)]$$

and the product of the marginals

$$\mu[P_x \times P_y] = \mathbb{E}_x \mathbb{E}_y[v((x, y), \cdot)]$$

Now consider as a measure of dependence:

$$\|\mu[P_{xy}] - \mu[P_x \times P_y]\|$$

Hilbert-Schmidt Independence Criterion

If we let $v((x, y), (x', y')) = k(x, x')l(y, y')$, then

$$\begin{aligned} & \|\mu[P_{xy}] - \mu[P_x \times P_y]\|^2 \\ &= \|\mathbb{E}_{xy}[k(x, \cdot)l(y, \cdot)] - \mathbb{E}_x[k(x, \cdot)]\mathbb{E}_y[l(y, \cdot)]\|^2 \\ &= \mathbb{E}_{xy}\mathbb{E}_{x'}\mathbb{E}_{y'}[k(x, x')l(y, y')] - 2\mathbb{E}_x\mathbb{E}_y\mathbb{E}_{x'y'}[k(x, x')l(y, y')] \\ & \quad + \mathbb{E}_x\mathbb{E}_y\mathbb{E}_{x'}\mathbb{E}_{y'}[k(x, x')l(y, y')]. \end{aligned}$$

This form yields the Hilbert-Schmidt Independence Criterion, which is the Hilbert-Schmidt norm of the cross-covariance operator between RKHSs:

$$\|\mathbb{E}_{x,y}[(\phi(x) - \mu_x) \otimes (\psi(y) - \mu_y)]\|_{\text{HS}}.$$

A cross-covariance derivation for HSIC

The reason that we factored $v((x, y), (x', y'))$ as

$$v((x, y), (x', y')) = k(x, x')l(y, y')$$

may not seem clear. In order to show why this factorization is valid, we need to come at HSIC from a different angle.

This alternate derivation results from considering the Hilbert-Schmidt norm (analogous to the Frobenius norm, but in a Hilbert space) of a cross-covariance operator of 2 kernelized random variables.

We kernelize:

- x as $\phi(x)$, or $k(x, \cdot)$, for $\phi \in \mathcal{F}$.
- y as $\psi(y)$, or $l(y, \cdot)$, for $\psi \in \mathcal{G}$.

Frobenius Norm

A $m \times n$ matrix A is simply a linear operator $A : \mathbb{R}^n \mapsto \mathbb{R}^m$.

$\|A\|_F^2$ is the square of the Frobenius norm of A :

$$\|A\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n A_{i,j}^2 = \sum_{i=1}^m \sum_{j=1}^n \langle Av_i, u_j \rangle^2$$

where

- $\{u_1, u_2, \dots, u_n\}$ is an orthonormal basis of \mathbb{R}^n
- $\{v_1, v_2, \dots, v_m\}$ is an orthonormal basis of \mathbb{R}^m

Hilbert-Schmidt Norm

Similarly, for RKHSs \mathcal{F}, \mathcal{G} a linear operator $C : \mathcal{G} \mapsto \mathcal{F}$ has Hilbert-Schmidt norm

$$\|C\|_{\text{HS}}^2 := \sum_{i,j} \langle Cv_i, u_j \rangle_{\mathcal{F}}^2$$

where

- $\{u_1, u_2, \dots\}$ is an orthonormal basis of \mathcal{F}
- $\{v_1, v_2, \dots\}$ is an orthonormal basis of \mathcal{G}

Tensor Products and Covariance Matrices

The tensor product of two vectors $x \in \mathbb{R}^m$, $y \in \mathbb{R}^n$ is

$$x \otimes y = xy^T = \begin{pmatrix} x_1y_1 & x_1y_2 & \dots & x_1y_n \\ x_2y_1 & x_2y_2 & \dots & x_2y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_my_1 & x_my_2 & \dots & x_my_n \end{pmatrix}$$

This is sometimes referred to as the outer product of two vectors. The name comes from the fact that the inner product is $x^T y$, while the outer product is xy^T .

The tensor product operation should be familiar from the calculation of a covariance matrix.

Tensor Products and Covariance Matrices

Suppose we have m observations of a random vector $X \in \mathbb{R}^n$

The population covariance matrix of X is in fact the expectation of the tensor product of the centered points:

$$\begin{aligned}\mathbb{E}[(x - \mu)(x - \mu)^T] &= \mathbb{E}[(x - \mu) \otimes (x - \mu)] \\ &= \sum_{i=1}^m \sum_{j=1}^m x_i \otimes x_j\end{aligned}$$

Properties of the tensor product

First, a couple of things to note.

The tensor product of $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ is

- A $n \times m$ matrix
- A linear operator $\mathbb{R}^m \mapsto \mathbb{R}^n$.

Also, for $z \in \mathbb{R}^m$, the following is true:

$$(x \otimes y)z = x\langle y, z \rangle$$

Tensor products can be generalized to RKHSs

Now, consider the tensor product of two functions $f \in \mathcal{F}$, $g \in \mathcal{G}$. While we do not have a matrix representation as before, we can say

$$f \otimes g : \mathcal{G} \mapsto \mathcal{F}.$$

We might ask, what is this the Hilbert-Schmidt norm of this operation? It will turn out that for HSIC, we will need to compute this norm, which nicely works out to be

$$\begin{aligned} \|f \otimes g\|_{\text{HS}}^2 &= \langle f \otimes g, f \otimes g \rangle_{\text{HS}}^2 &&= \langle f, (f \otimes g)g \rangle_{\mathcal{F}} \\ &= \langle f, f \rangle_{\mathcal{F}} \langle g, g \rangle_{\mathcal{G}} &&= \|f\|_{\mathcal{F}}^2 \|g\|_{\mathcal{G}}^2. \end{aligned}$$

Back to HSIC

Similar to the familiar covariance operator from before, the cross-covariance operator between random variables X and Y is the cross-covariance of $\phi(X)$ and $\psi(Y)$:

$$C_{x,y} = \mathbb{E}_{x,y}[(\phi(x) - \mu_x) \otimes (\psi(y) - \mu_y)].$$

This becomes

$$\begin{aligned} \|C_{xy}\|_{\text{HS}}^2 &= \mathbb{E}_{x,y,x',y'}[\langle \phi(x) \otimes \psi(y), \phi(x) \otimes \psi(y) \rangle_{\text{HS}}] \\ &\quad - 2\mathbb{E}_{x,y}[\langle \mu_x \otimes \mu_y, \phi(x) \otimes \psi(y) \rangle_{\text{HS}}] \\ &\quad + \langle \mu_x \otimes \mu_y, \mu_x \otimes \mu_y \rangle_{\text{HS}} \end{aligned}$$

Using the equivalence from the last slide and doing some straightforward computations, we can show that the above is precisely HSIC from before:

$$\begin{aligned} \mathbb{E}_{xy} \mathbb{E}_{x'} \mathbb{E}_{y'} [k(x, x') l(y, y')] - 2\mathbb{E}_x \mathbb{E}_y \mathbb{E}_{x'} \mathbb{E}_{y'} [k(x, x') l(y, y')] \\ + \mathbb{E}_x \mathbb{E}_y \mathbb{E}_{x'} \mathbb{E}_{y'} [k(x, x') l(y, y')]. \end{aligned}$$

Empirical HSIC

Now that we have (mostly) shown the derivation for HSIC, we might ask what the population statistic is for real data.

The population statistic for HSIC is

$$HSIC(Z, \mathcal{F}, \mathcal{G}) = \frac{1}{(m-1)^2} \text{tr} KHLH$$

where

- K is the kernel matrix for kernel $k(\cdot, \cdot)$ [$K_{ij} = k(x_i, x_j)$]
- L is the kernel matrix for kernel $l(\cdot, \cdot)$ [$L_{ij} = l(y_i, y_j)$]
- $H_{ij} = \delta_{ij} - \frac{1}{m}$

HSIC Properties and MMD

- Exhibits $O(m^{-1})$ bias \Rightarrow Fast convergence!
- Can be used for Independent Component Analysis
- ICA for structured data

Further, the idea of embedding distributions into Hilbert spaces can be used for the maximum mean discrepancy method:

- Use a *witness function* to identify whether two distributions are different.
- Witness function should maximize discrepancy with means of two sets of samples.