SC4/SM8 Advanced Topics in Statistical Machine Learning

Kernel Methods

Dino Sejdinovic
Department of Statistics
Oxford

Slides and other materials available at:
http://www.stats.ox.ac.uk/~sejdinov/atsml19/

Dual C-SVM

$$\text{maximize} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j,$$

subject to the constraints

$$0 \le \alpha_i \le C, \quad \sum_{i=1}^{n} y_i \alpha_i = 0$$

From α, obtain the hyperplane with

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i.$$

Offset b can be obtained from any of the margin SVs (for which $\alpha_i \in (0, C)$): $1 = y_i \left(w^\top x_i + b\right)$.

Dual form and Inner Products

We have stumbled across something quite interesting. Dual program

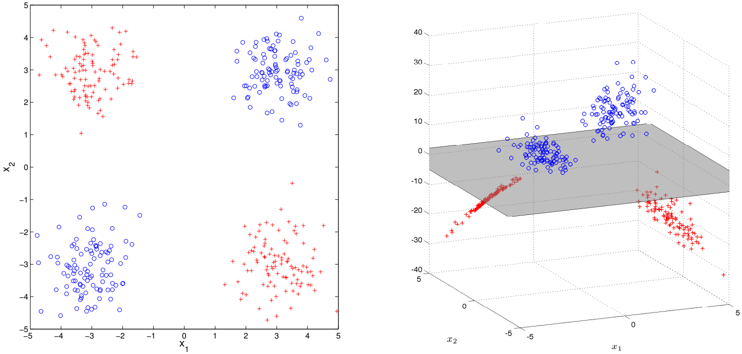
$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j \qquad \text{subject to} \quad \begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 \\ 0 \preceq \alpha \preceq C \end{cases}$$

only depends on inputs x_i through their inner products (similarities) with other inputs.
Decision function

$$f(x) = \text{sign}(w^\top x + b) = \text{sign}(\sum_{i=1}^{n} \alpha_i y_i x_i^\top x + b)$$

also depends only on the similarity of a test point x to the training points x_i.
Thus, we do not need explicit inputs - just their pairwise similarities.
Key property: even if $p > n$, it is still the case that $w \in \text{span}\{x_i : i = 1, \ldots, n\}$ (normal vector of the hyperplane lives in the subspace spanned by the datapoints).

Beyond Linear Classifiers



- No linear classifier separates red from blue.
- Linear separation after mapping to a **higher dimensional feature space**:

$$\mathbb{R}^2 \ni \begin{pmatrix} x^{(1)} & x^{(2)} \end{pmatrix}^\top = x \;\mapsto\; \varphi(x) = \begin{pmatrix} x^{(1)} & x^{(2)} & x^{(1)}x^{(2)} \end{pmatrix}^\top \in \mathbb{R}^3$$

Non-Linear SVM

- Consider the dual C-SVM with explicit non-linear transformation $x \mapsto \varphi(x)$:

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \varphi(x_i)^\top \varphi(x_j) \qquad \text{subject to} \quad \begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 \\ 0 \preceq \alpha \preceq C \end{cases}$$

- Suppose $p = 2$, and we would like to introduce quadratic non-linearities,

$$\varphi(x) = \left(1, \sqrt{2}x^{(1)}, \sqrt{2}x^{(2)}, \sqrt{2}x^{(1)}x^{(2)}, \left(x^{(1)}\right)^2, \left(x^{(2)}\right)^2\right)^\top.$$

  Then

$$\begin{aligned}\varphi(x_i)^\top \varphi(x_j) &= 1 + 2x_i^{(1)}x_j^{(1)} + 2x_i^{(2)}x_j^{(2)} + 2x_i^{(1)}x_i^{(2)}x_j^{(1)}x_j^{(2)} \\ &\quad + \left(x_i^{(1)}\right)^2 \left(x_j^{(1)}\right)^2 + \left(x_i^{(2)}\right)^2 \left(x_j^{(2)}\right)^2 = (1 + x_i^\top x_j)^2\end{aligned}$$

- Since only inner products are needed, non-linear transform need not be computed explicitly - inner product between features can be a simple function (**kernel**) of x_i and x_j: $k(x_i, x_j) = \varphi(x_i)^\top \varphi(x_j) = (1 + x_i^\top x_j)^2$
- d-order interactions can be implemented by $k(x_i, x_j) = (1 + x_i^\top x_j)^d$ (**polynomial kernel**). Never need to compute explicit feature expansion of dimension $\binom{p+d}{d}$ where this inner product happens!

Kernel SVM: Kernel trick

- Kernel SVM with $k(x_i, x_j)$. Non-linear transformation $x \mapsto \varphi(x)$ still present, but **implicit** (coordinates of the vector $\varphi(x)$ are never computed).

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad \text{subject to} \quad \begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 \\ 0 \preceq \alpha \preceq C \end{cases}$$

- Prediction? $f(x) = \text{sign}\left(w^\top \varphi(x) + b\right)$, where $w = \sum_{i=1}^{n} \alpha_i y_i \varphi(x_i)$ and offset b obtained from a margin support vector x_j with $\alpha_j \in (0, C)$.
  - No need to compute w either! Just need

$$w^\top \varphi(x) = \sum_{i=1}^{n} \alpha_i y_i \varphi(x_i)^\top \varphi(x) = \sum_{i=1}^{n} \alpha_i y_i k(x_i, x).$$

  - Get offset from

$$b = y_j - w^\top \varphi(x_j) = y_j - \sum_{i=1}^{n} \alpha_i y_i k(x_i, x_j)$$

    for any margin support-vector x_j ($\alpha_j \in (0, C)$).
- Fitted a separating hyperplane in a high-dimensional feature space without ever mapping explicitly to that space.

Kernel trick in general

- In a learning algorithm, if only inner products $x_i^\top x_j$ are explicitly used, rather than data items x_i, x_j directly, we can replace them with a kernel function $k(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$, where $\varphi(x)$ could be **nonlinear, high- and potentially infinite-dimensional** features of the original data.
  - Kernel ridge regression
  - Kernel logistic regression
  - Kernel PCA, CCA, ICA
  - Kernel K-means

Kernel Methods and Reproducing Kernel Hilbert Spaces

slides based on Arthur Gretton's Reproducing kernel Hilbert spaces in Machine Learning course

Kernel: an inner product between feature maps

Definition (kernel)

Let $\mathcal{X}$ be a non-empty set. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a **kernel** if there exists a **Hilbert space** and a map $\varphi : \mathcal{X} \to \mathcal{H}$ such that $\forall x, x' \in \mathcal{X}$,

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

- Almost no conditions on $\mathcal{X}$ (eg, $\mathcal{X}$ itself need not have an inner product, e.g., documents).
- Think of kernel as a **similarity measure between features**

What are some simple kernels? E.g., for text documents? For images?

- A single kernel can correspond to multiple sets of underlying features.

$$\varphi_1(x) = x \qquad \text{and} \qquad \varphi_2(x) = \begin{pmatrix} x/\sqrt{2} & x/\sqrt{2} \end{pmatrix}^\top$$

Positive semidefinite functions

If we are given a “measure of similarity” with two arguments, $k(x, x')$, how can we determine if it is a valid kernel?

1. Find a feature map?
   - Sometimes not obvious (especially if the feature vector is infinite dimensional)
2. A simpler direct property of the function: positive semidefiniteness.

Positive semidefinite functions

Definition (Positive semidefinite functions)

A symmetric function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is positive semidefinite if $\forall n \geq 1,\ \forall (a_1, \ldots a_n) \in \mathbb{R}^n,\ \forall (x_1, \ldots, x_n) \in \mathcal{X}^n$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j \kappa(x_i, x_j) \geq 0.$$

- Kernel $k(x, y) := \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}}$ for a Hilbert space $\mathcal{H}$ is positive semidefinite.

$$\begin{aligned} \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(x_i, x_j) &= \sum_{i=1}^{n} \sum_{j=1}^{n} \langle a_i \varphi(x_i), a_j \varphi(x_j) \rangle_{\mathcal{H}} \\ &= \left\| \sum_{i=1}^{n} a_i \varphi(x_i) \right\|_{\mathcal{H}}^2 \geq 0. \end{aligned}$$

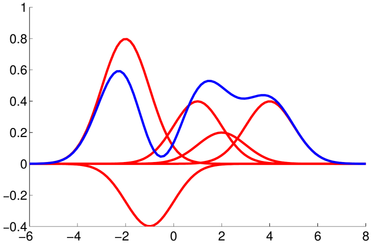
Positive semidefinite functions are kernels

Moore-Aronszajn Theorem

Every positive semidefinite function is a kernel for some Hilbert space $\mathcal{H}$.

- $\mathcal{H}$ is usually thought of as a space of functions (**Reproducing kernel Hilbert space - RKHS**)

Gaussian RBF kernel $k(x, x') = \exp\left(-\frac{1}{2\gamma^2}\|x - x'\|^2\right)$ has an infinite-dimensional $\mathcal{H}$ with elements $h(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x)$ and their pointwise limits.



Reproducing kernel

Definition (Reproducing kernel)

Let $\mathcal{H}$ be a Hilbert space of functions $f : \mathcal{X} \to \mathbb{R}$ defined on a non-empty set $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called **a reproducing kernel** of $\mathcal{H}$ if it satisfies

- $\forall x \in \mathcal{X}, \quad k_x = k(\cdot, x) \in \mathcal{H}$,
- $\forall x \in \mathcal{X}, \forall f \in \mathcal{H}, \quad \langle f, k(\cdot, x) \rangle_{\mathcal{H}} = f(x)$ (the reproducing property).

In particular, for any $x, y \in \mathcal{X}$, $k(x, y) = \langle k(\cdot, y), k(\cdot, x) \rangle_{\mathcal{H}} = \langle k(\cdot, x), k(\cdot, y) \rangle_{\mathcal{H}}$.

Can forget all about $\varphi(x)$ and just treat $k(\cdot, x)$ as a feature of x (it is a perfectly valid Hilbert-space valued feature)!

RKHS

Definition (Reproducing kernel Hilbert space)

A Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \to \mathbb{R}$, defined on a non-empty set $\mathcal{X}$ is said to be a Reproducing Kernel Hilbert Space (RKHS) if evaluation functionals $\delta_x : \mathcal{H} \to \mathbb{R}$, $\delta_x f = f(x)$ are continuous $\forall x \in \mathcal{X}$.

Theorem (Norm convergence implies pointwise convergence)

If $\lim_{n\to\infty} \|f_n - f\|_{\mathcal{H}} = 0$*, then* $\lim_{n\to\infty} f_n(x) = f(x), \forall x \in \mathcal{X}$.

- If two functions $f, g \in \mathcal{H}$ are close in the norm of $\mathcal{H}$, then $f(x)$ and $g(x)$ are close for all $x \in \mathcal{X}$
- This is a property of particularly “nice” functional spaces. For example, does not hold on spaces endowed with L_2 norm: x^n on $[0, 1]$ converges to 0 in L_2 but not pointwise.

Back to SVMs

Maximum margin classifier in RKHS: Looking for a decision function of form $\text{sign}(f(x))$ where $f \in \mathcal{H}_k$. Because we are in an RKHS, $f(x) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}_k}$.

$$\min_{f \in \mathcal{H}_k} \left(\frac{1}{2}\|f\|^2_{\mathcal{H}_k} + C \sum_{i=1}^{n} \left(1 - y_i \langle f, k(\cdot, x_i)\rangle_{\mathcal{H}_k}\right)_+ \right)$$

for the RKHS $\mathcal{H}$ with kernel $k(x, x')$. Maximizing the margin equivalent to minimizing $\|f\|^2_{\mathcal{H}}$: for many RKHSs a smoothness constraint on function f (more about this later).
Why can we solve this infinite-dimensional optimization problem? Because we know that $f \in \text{span}\left\{k(\cdot, x_i) : i = 1, \ldots, n\right\}$ – Representer Theorem.

Representer Theorem

Representer theorem

Standard supervised learning setup: we are given a set of paired observations $(x_1, y_1), \ldots (x_n, y_n)$.
Goal: find the function f^* in the RKHS $\mathcal{H}$ which solves the regularized empirical risk minimization problem.

$$\min_{f \in \mathcal{H}} \hat{R}(f) + \Omega\left(\|f\|_{\mathcal{H}}^2\right),$$

where empirical risk is

$$\hat{R}(f) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i), x_i),$$

and Ω is a non-decreasing function.

- Classification: L could be a hinge loss $L(y, f(x), x) = (1 - yf(x))_+$ or a logistic loss $L(y, f(x), x) = \log(1 + \exp(-yf(x)))$.
- Regression: $L(y, f(x), x) = (y - f(x))^2$.

Representer theorem

Theorem (Representer Theorem)

There is a solution to

$$\min_{f \in \mathcal{H}} \hat{R}(f) + \Omega\left(\|f\|_{\mathcal{H}}^2\right)$$

that takes the form

$$f^* = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i).$$

If Ω is strictly increasing, all solutions have this form.

Representer theorem: proof

Proof: Denote f_s projection of f onto the subspace

$$\mathrm{span}\left\{k(\cdot, x_i) : \ i = 1, \ldots, n\right\}$$

such that

$$f = f_s + f_\perp,$$

where $f_s = \sum_{i=1}^n \alpha_i k(\cdot, x_i)$ and $f_\perp$ is orthogonal to $\mathrm{span}\left\{k(\cdot, x_i) : \ i = 1, \ldots, n\right\}$.
Regularizer:

$$\|f\|_{\mathcal{H}}^2 = \|f_s\|_{\mathcal{H}}^2 + \|f_\perp\|_{\mathcal{H}}^2 \geq \|f_s\|_{\mathcal{H}}^2,$$

then

$$\Omega\left(\|f\|_{\mathcal{H}}^2\right) \geq \Omega\left(\|f_s\|_{\mathcal{H}}^2\right).$$

Representer theorem: proof

Proof (cont.): Individual terms $f(x_i)$ in the loss:

$$f(x_i) = \langle f, k(\cdot, x_i) \rangle_{\mathcal{H}} = \langle f_s + f_\perp, k(\cdot, x_i) \rangle_{\mathcal{H}} = \langle f_s, k(\cdot, x_i) \rangle_{\mathcal{H}} ,$$

so

$$L(y_i, f(x_i), x_i) = L(y_i, f_s(x_i), x_i) \forall i \implies \hat{R}(f) = \hat{R}(f_s).$$

Hence

- The empirical risk only depends on the components of f lying in the subspace spanned by canonical features.
- Regularizer $\Omega(\ldots)$ is minimized when $f = f_s$.
- If Ω is strictly non-decreasing, then $\|f_\perp\|_{\mathcal{H}} = 0$ is required at the minimum.

Kernel Ridge Regression

Regularised Least Squares

We are given n training points $\{x_i\}_{i=1}^n$ in $\mathbb{R}^p$: Define some $\lambda > 0$. Our goal is:

$$\begin{aligned} w^* &= \arg\min_{w \in \mathbb{R}^p} \left(\sum_{i=1}^n (y_i - x_i^\top w)^2 + \lambda \|w\|^2 \right) \\ &= \arg\min_{w \in \mathbb{R}^p} \left(\|\mathbf{y} - \mathbf{X}w\|^2 + \lambda \|w\|^2 \right), \end{aligned}$$

Solution is:

$$w^* = \left(\mathbf{X}^\top \mathbf{X} + \lambda I\right)^{-1} \mathbf{X}^\top \mathbf{y},$$

which is the standard regularised least squares solution.

Kernel ridge regression

Use features $\phi(x_i)$ in the place of x_i:

$$w^* = \arg\min_{w \in \mathcal{H}} \left(\sum_{i=1}^{n} (y_i - \langle w, \phi(x_i) \rangle_{\mathcal{H}})^2 + \lambda \|w\|_{\mathcal{H}}^2 \right).$$

E.g. for finite dimensional feature spaces,

$$\phi_p(x) = \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^\ell \end{bmatrix} \qquad \phi_s(x) = \begin{bmatrix} \sin(x) \\ \cos(x) \\ \sin(2x) \\ \vdots \\ \cos\left(\frac{\ell}{2}x\right) \end{bmatrix}$$

In finite dimensions, w is a vector of length ℓ giving weight to each of these features so that learned function is $f_w(x) = w^\top \phi(x)$. Feature vectors can also have **infinite** length.

Kernel ridge regression

Recall that feature maps ϕ and feature spaces $\mathcal{H}$ are not unique, but RKHS $\mathcal{H}_k$ is. Thus, we can identify w with the function f_w (there is an isometry between w and f_w: $\|w\|_{\mathcal{H}} = \|f_w\|_{\mathcal{H}_k}$ regardless of the choice of the feature space $\mathcal{H}$) and write

$$
\begin{aligned}
f^* &= \arg\min_{f\in\mathcal{H}_k} \left(\sum_{i=1}^{n} \left(y_i - \langle f, k(\cdot, x_i)\rangle_{\mathcal{H}}\right)^2 + \lambda \|f\|^2_{\mathcal{H}_k} \right) \\
&= \arg\min_{f\in\mathcal{H}_k} \left(\sum_{i=1}^{n} \left(y_i - f(x_i)\right)^2 + \lambda \|f\|^2_{\mathcal{H}_k} \right).
\end{aligned}
$$

Kernel ridge regression

Recall the representer theorem: f is a linear combination of feature space mappings of data points

$$f = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i).$$

Then

$$\begin{aligned}\sum_{i=1}^{n} \left(y_i - \langle f, k(\cdot, x_i)\rangle_{\mathcal{H}_k}\right)^2 + \lambda \|f\|^2_{\mathcal{H}_k} &= \|\mathbf{y} - \mathbf{K}\alpha\|^2 + \lambda \alpha^\top \mathbf{K}\alpha \\ &= \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{K}\alpha + \alpha^\top \left(\mathbf{K}^2 + \lambda \mathbf{K}\right)\alpha\end{aligned}$$

Differentiating wrt α and setting this to zero, we get

$$\alpha^* = (\mathbf{K} + \lambda I_n)^{-1} y.$$

Recall: $\frac{\partial \alpha^\top U \alpha}{\partial \alpha} = (U + U^\top)\alpha, \qquad \frac{\partial v^\top \alpha}{\partial \alpha} = \frac{\partial \alpha^\top v}{\partial \alpha} = v$

Parameter selection for KRR

Given the objective

$$f^* = \arg\min_{f \in \mathcal{H}_k} \left(\sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|_{\mathcal{H}_k}^2 \right).$$

How do we choose

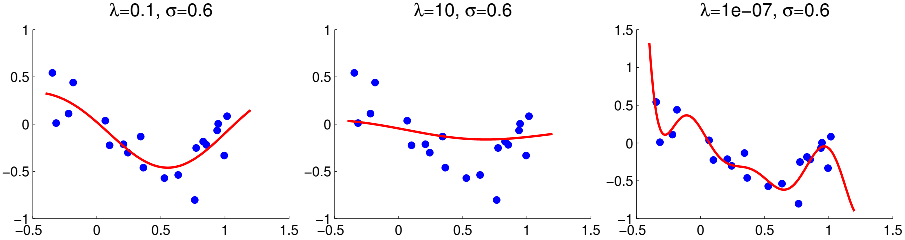
- The regularization parameter λ?
- The kernel parameter: for Gaussian kernel, σ in

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

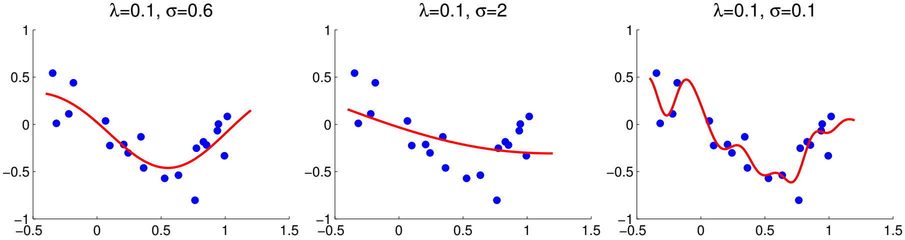
Beware: Gaussian kernel has many different parametrisations in the literature and software packages!

Typically use cross-validation.

Choice of λ



Choice of σ



Kernel families and operations with kernels

Examples of kernels

- **Linear**: $k(x,x') = x^\top x'$.
- **Polynomial**: $k(x,x') = (c + x^\top x')^m$, $c \in \mathbb{R}$, $m \in \mathbb{N}$.
- **Periodic (1d)**: $k(x,x') = \exp\left(-\frac{2\sin^2(\pi|x-x'|/p)}{\gamma^2}\right)$, period p, $\gamma > 0$.
- **Exponential**: $k(x,x') = \exp(\frac{x^\top x'}{\gamma})$, $\gamma > 0$.
- **Gaussian RBF**: $k(x,x') = \exp\left(-\frac{1}{2\gamma^2}\|x - x'\|^2\right)$, $\gamma > 0$.
- **Laplace**: $k(x,x') = \exp\left(-\frac{1}{\gamma}\|x - x'\|\right)$, $\gamma > 0$.
- **Rational quadratic**: $k(x,x') = \left(1 + \frac{\|x-x'\|^2}{2\alpha\gamma^2}\right)^{-\alpha}$, $\alpha, \gamma > 0$.
- **Brownian covariance**: $k(x,x') = \frac{1}{2}\left(\|x\|^\gamma + \|x'\|^\gamma - \|x - x'\|^\gamma\right)$, $\gamma \in [0,2]$.

all norms are 2-norms unless specified otherwise

Matérn Family

$$k(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\gamma} \|x - x'\| \right)^{\nu} K_{\nu} \left(\frac{\sqrt{2\nu}}{\gamma} \|x - x'\| \right), \quad \nu > 0, \gamma > 0,$$

where K_ν is the modified Bessel function of the second kind of order ν.

- $\nu = 1/2$: $k(x,x') = \exp\left(-\frac{1}{\gamma}\|x - x'\|\right)$
- $\nu = 3/2$: $k(x,x') = \left(1 + \frac{\sqrt{3}}{\gamma}\|x - x'\|\right) \exp\left(-\frac{\sqrt{3}}{\gamma}\|x - x'\|\right)$
- $\nu = 5/2$: $k(x,x') = \left(1 + \frac{\sqrt{5}}{\gamma}\|x - x'\| + \frac{5}{3\gamma^2}\|x - x'\|^2\right) \exp\left(-\frac{\sqrt{5}}{\gamma}\|x - x'\|\right)$
- as $\nu \to \infty$, converges to Gaussian RBF $k(x,x') = \exp\left(-\frac{1}{2\gamma^2}\|x - x'\|^2\right)$

Matérn family norms penalize the derivatives of f. In particular, for $\nu = s + 1/2$, it penalizes the derivatives up to order $s + 1$, e.g. for $\nu = 3/2$ and in one dimension:

$$\|f\|^2_{\mathcal{H}_k} \propto \int f''(x)^2 dx + \frac{6}{\gamma^2} \int f'(x)^2 dx + \frac{9}{\gamma^4} \int f(x)^2 dx$$

New kernels from old: sums, transformations

The great majority of useful kernels are built from simpler kernels.

Lemma (Sums of kernels are kernels)

Given $\alpha > 0$ and k, k_1 and k_2 all kernels on $\mathcal{X}$, then αk and $k_1 + k_2$ are kernels on $\mathcal{X}$.

To prove this, just check inner product definition (features get scaled with $\sqrt{\alpha}$ or concatenated). A difference of kernels need not be a kernel (**why?**)

Lemma (Space transformation)

Let $\mathcal{X}$ and $\widetilde{\mathcal{X}}$ be sets, and consider any map $s : \mathcal{X} \to \widetilde{\mathcal{X}}$. Let $\tilde{k}$ be a kernel on $\widetilde{\mathcal{X}}$. Then $k(x, x') = \tilde{k}(s(x), s(x'))$ is a kernel on $\mathcal{X}$.

Proof: if $\tilde{\varphi}$ is a feature map for $\tilde{k}$, then $\varphi = \tilde{\varphi} \circ s$ is a feature map for k.

New kernels from old: products

Lemma (Products of kernels are kernels)

Given k_1 on $\mathcal{X}_1$ and k_2 on $\mathcal{X}_2$, then $k_1 \times k_2$ is a kernel on $\mathcal{X}_1 \times \mathcal{X}_2$.

Proof.

Sketch for finite-dimensional spaces only. Assume $\mathcal{H}_1$ corresponding to k_1 is $\mathbb{R}^m$, and $\mathcal{H}_2$ corresponding to k_2 is $\mathbb{R}^n$. Define:

- $k_1 := u^\top v$ for $u, v \in \mathbb{R}^m$ (e.g.: kernel between two images)
- $k_2 := p^\top q$ for $p, q \in \mathbb{R}^n$ (e.g.: kernel between two captions)

Is the following a kernel?

$$K\left[(u,p);(v,q)\right] = k_1 \times k_2$$

(e.g. kernel between one image-caption pair and another)

□

New kernels from old: products

Proof.

(continued)

$$\begin{aligned} k_1 k_2 &= \left(u^\top v\right)\left(q^\top p\right) \\ &= \operatorname{trace}(u^\top v q^\top p) \\ &= \operatorname{trace}(p u^\top v q^\top) \\ &= \langle A, B \rangle, \end{aligned}$$

where $A := pu^\top$, $B := qv^\top$ (features of image-caption pairs) Thus $k_1 k_2$ is a valid kernel, since inner product between $A, B \in \mathbb{R}^{m \times n}$ is

$$\langle A, B \rangle = \operatorname{trace}(AB^\top).$$

□

Another way: just note that the **Kronecker product of positive definite matrices is positive definite**!

More products and Taylor expansions

Lemma (Products of kernels are kernels)

Given kernels k_1 and k_2 on $\mathcal{X}$ $k_1 \times k_2$ is a kernel on $\mathcal{X}$.

Proof: It is certainly a kernel on $\mathcal{X} \times \mathcal{X}$, so just consider space transformation $s : \mathcal{X} \to \mathcal{X} \times \mathcal{X}$ with $s(x) = (x, x)$.
Another way: just note that the **Hadamard product of positive definite matrices is positive definite**!
As a corollary:

$$k(x, x') = c + \sum_{j=1}^{d} a_j \langle x, x' \rangle^d \tag{1}$$

is certainly a kernel. Readily extends to

$$k(x, x') = g\left(\langle x, x' \rangle\right) \tag{2}$$

for an analytic function g with nonnegative Taylor coefficients, e.g., $\exp$.

Gaussian RBF is a kernel

As a product of an exponential kernel and a kernel with 1-d feature $x \mapsto \exp\left(-\frac{\|x\|^2}{2\gamma^2}\right)$.

$$\begin{aligned} k(x,x') &= \exp\left(-\frac{1}{2\gamma^2}\|x-x'\|^2\right) \\ &= \exp\left(-\frac{\|x\|^2}{2\gamma^2}\right)\exp\left(-\frac{\|x'\|^2}{2\gamma^2}\right)\exp\left(\frac{1}{\gamma^2}\langle x,x'\rangle\right) \end{aligned}$$

All of the proofs above are constructive: they give a way of constructing new features from old. But the resulting features quickly become very difficult to interpret. There is another, much cleaner way to do this: Mercer's Theorem.

Mercer's theorem

- Assume that $\mathcal{X}$ is a compact metric space, $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ a continuous kernel and fix a finite measure ν on $\mathcal{X}$ with $\text{supp}\nu = \mathcal{X}$.
- To k we can associate a certain operator T_k on $L_2(\mathcal{X}; \nu)$ which is compact, positive and self-adjoint

$$[T_k f](y) = \int f(x)k(x,y)\nu(dx)$$

- There exist an orthonormal set of **continuous** L_2 functions $\{e_j\}_{j\in J}$ and $\{\lambda_j\}_{j\in J}$ (strictly positive eigenvalues with $\lambda_j \to 0$; J at most countable).

Theorem (Mercer's theorem)

$\forall x, y \in \mathcal{X}$ *with convergence uniform on* $\mathcal{X} \times \mathcal{X}$*:*

$$k(x,y) = \sum_{j\in J} \lambda_j e_j(x) e_j(y).$$

Mercer's theorem

$$\begin{aligned} k(x,y) &= \sum_{j\in J} \lambda_j e_j(x) e_j(y) \\ &= \left\langle \left\{ \sqrt{\lambda_j} e_j(x) \right\}, \left\{ \sqrt{\lambda_j} e_j(y) \right\} \right\rangle_{\ell^2(J)} \end{aligned}$$

Another (Mercer) feature map:

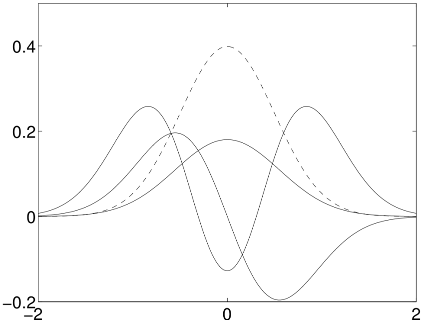
$$\begin{aligned} \varphi : \mathcal{X} &\to \ell^2(J) \\ \varphi : x &\mapsto \left\{ \sqrt{\lambda_j} e_j(x) \right\}_{j\in J} \end{aligned}$$

Mercer's Theorem and Smoothness

What does $\|f\|_{\mathcal{H}}$ have to do with smoothing? For the Gaussian kernel:

$$f(x) = \sum_{r=1}^{\infty} a_r e_r(x), \qquad \|f\|_{\mathcal{H}}^2 = \sum_{r=1}^{\infty} \frac{a_r^2}{\lambda_r}.$$

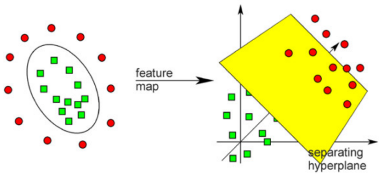
$\lambda_r \sim B^r \to 0$, as $r \to \infty$ for $B \in (0, 1)$ and $e_r(x)$ are functions of increasing complexity as r increases (r zero-crossings) – related to r-th order **Hermite polynomials.** Figure from Rasmussen and Williams, 2006



RKHS Embeddings of Distributions

Kernel Trick and Kernel Mean Trick

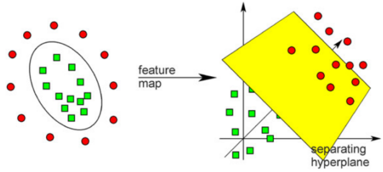
- implicit feature map $x \mapsto k(\cdot, x) \in \mathcal{H}_k$
  replaces $x \mapsto [\varphi_1(x), \ldots, \varphi_s(x)] \in \mathbb{R}^s$
- $\langle k(\cdot, x), k(\cdot, y)\rangle_{\mathcal{H}_k} = k(x, y)$
  inner products readily available
  - nonlinear decision boundaries, nonlinear regression functions, learning on non-Euclidean/structured data



[Cortes & Vapnik, 1995; Schölkopf & Smola, 2001]

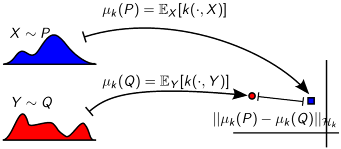
Kernel Trick and Kernel Mean Trick

- implicit feature map $x \mapsto k(\cdot, x) \in \mathcal{H}_k$
  replaces $x \mapsto [\varphi_1(x), \ldots, \varphi_s(x)] \in \mathbb{R}^s$
- $\langle k(\cdot, x), k(\cdot, y)\rangle_{\mathcal{H}_k} = k(x, y)$
  inner products readily available
  - nonlinear decision boundaries, nonlinear regression functions, learning on non-Euclidean/structured data



[Cortes & Vapnik, 1995; Schölkopf & Smola, 2001]

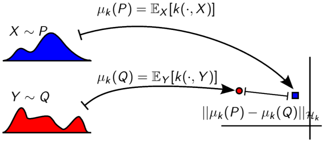
- **RKHS embedding**: implicit feature mean
  [Smola et al, 2007; Sriperumbudur et al, 2010]
  $P \mapsto \mu_k(P) = \mathbb{E}_{X\sim P} k(\cdot, X) \in \mathcal{H}_k$
  replaces $P \mapsto [\mathbb{E}\varphi_1(X), \ldots, \mathbb{E}\varphi_s(X)] \in \mathbb{R}^s$
- $\langle \mu_k(P), \mu_k(Q)\rangle_{\mathcal{H}_k} = \mathbb{E}_{X\sim P, Y\sim Q} k(X, Y)$
  inner products easy to estimate
  - multiple instance learning / learning on distributions, nonparametric testing for homogeneity, independence, conditional independence, three-variable interaction

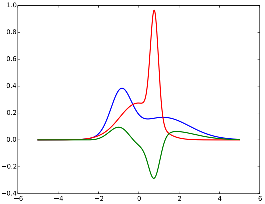


[Gretton et al, 2005; Gretton et al, 2006; Fukumizu et al, 2007; DS, Bergsma & Gretton, 2013; Szabo et al, 2015]

Maximum Mean Discrepancy

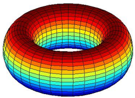
- **Maximum Mean Discrepancy (MMD)** [Borgwardt et al, 2006; Gretton et al, 2007] between P and Q:

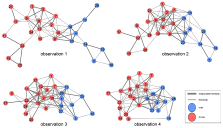


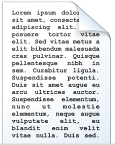


$$\text{MMD}_k(P, Q) = \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_k} = \sup_{f \in \mathcal{H}_k:\, \|f\|_{\mathcal{H}_k} \leq 1} |\mathbb{E}f(X) - \mathbb{E}f(Y)|$$

- **Characteristic** kernels: $\text{MMD}_k(P, Q) = 0$ iff $P = Q$ (also metrizes weak* [Sriperumbudur,2010]).
  - Gaussian RBF $\exp(-\frac{1}{2\sigma^2} \|x - x'\|_2^2)$, Matérn family, inverse multiquadrics.
- Can encode structural properties in the data: kernels on non-Euclidean domains, networks, images, text...







Two-sample testing on nonstandard domains

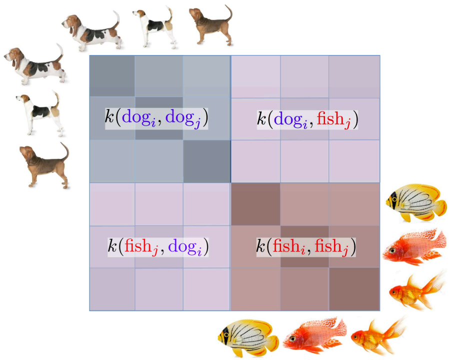


Figure by Arthur Gretton

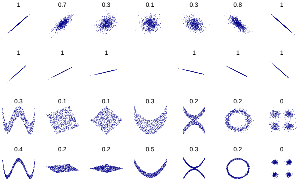
Average similarity within two samples vs average similarity across two samples.

MMD has been applied to:

- independence tests on text data [Gretton et al, 2009]
- two-sample tests on graphs [Gretton et al, 2012]
- training generative neural networks for image data [Dziugaite, Roy and Ghahramani, 2015]
- two-sample tests on persistence diagrams in topological data analysis [Kwitt et al, 2015]
- similarity measure between observed and simulated data in ABC [Park, Jitkrittum and DS, 2015]

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

Kernel dependence measures: HSIC



cor vs. dcor

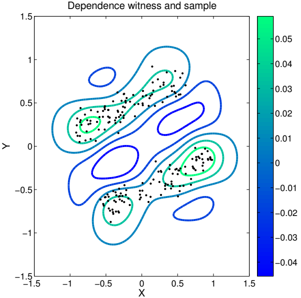
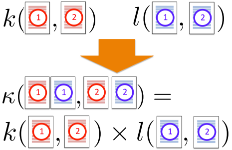


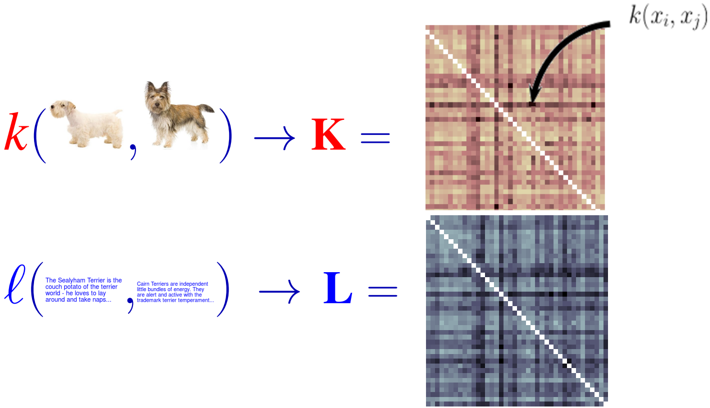
Figure by Arthur Gretton

- $HSIC^2(X, Y; \kappa) = \|\mu_\kappa(P_{XY}) - \mu_\kappa(P_X P_Y)\|^2_{\mathcal{H}_\kappa}$
- Hilbert-Schmidt norm of the feature-space cross-covariance [Gretton et al, 2009]
- dependence witness is a smooth function in the RKHS $\mathcal{H}_\kappa$ of functions on $\mathcal{X} \times \mathcal{Y}$



- Independence testing framework that generalises Distance Correlation (dcor) of [Szekely et al, 2007]: HSIC with Brownian motion kernels [DS et al, 2013]
- Extends to multivariate interaction and joint dependence measures [DS et al, 2013; Pfister et al, 2017]

Kernel dependence measures: HSIC (2)



Hilbert-Schmidt Independence Criterion (**HSIC**): similarity between the kernel matrices $\left\langle \tilde{\mathbf{K}}, \tilde{\mathbf{L}} \right\rangle = \boxed{\text{Tr}\left(\tilde{\mathbf{K}}\tilde{\mathbf{L}}\right)}$, where $\tilde{\mathbf{K}} = \mathbf{HKH}$, and $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbb{1}\mathbb{1}^\top$ is the centering matrix. [Gretton et al, 2008; Fukumizu et al, 2008; Song et al, 2012]

Distribution Regression

- supervised learning where labels are available at the group, rather than at the individual level.

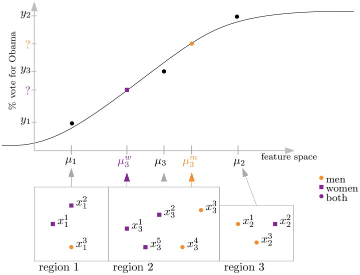


Figure from Flaxman et al, 2015

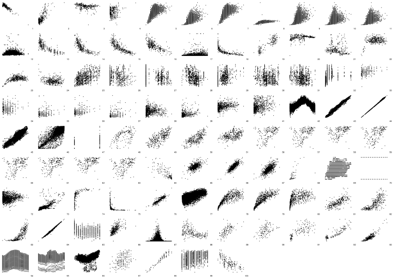


Figure from Mooij et al, 2014

- classifying text based on word features [Yoshikawa et al, 2014; Kusner et al, 2015]
- aggregate voting behaviour of demographic groups [Flaxman et al, 2015; 2016]
- image labels based on a distribution of small patches [Szabo et al, 2016]
- “traditional” parametric statistical inference by learning a function from sets of samples to parameters: ABC [Mitrovic et al, 2016], EP [Jitkrittum et al, 2015]
- identify the cause-effect direction between a pair of variables from a joint sample [Lopez-Paz et al,2015]

Distribution Regression (2)

- Multiple-Instance Learning: Input is a bag of B_i vectors $\{x_{i1}, \ldots, x_{iB_i}\}$, each $x_{ia} \in X$ assumed to arise from a probability distribution P_i on $\mathcal{X}$.
- Represent the i-th bag by the corresponding empirical kernel embedding $\mathfrak{m}_i = \mu_k[\mathsf{P}_i] = \frac{1}{B_i}\sum_{a=1}^{B_i} k(\cdot, x_{ia})$ w.r.t. a kernel k on $\mathcal{X}$.
- Now treat the problem as having inputs $\mathfrak{m}_i \in \mathcal{H}_k$: just need to define a **kernel** K on $\mathcal{H}_k$.

Linear: $$K(\mathfrak{m}_i, \mathfrak{m}_j) = \langle \mathfrak{m}_i, \mathfrak{m}_j \rangle_{\mathcal{H}_k} = \frac{1}{B_i B_j} \sum_{a=1}^{B_i} \sum_{b=1}^{B_j} k(x_{ia}, x_{jb})$$

Gaussian: $$K(\mathfrak{m}_i, \mathfrak{m}_j) = \exp\left(-\frac{1}{2\gamma^2} \|\mathfrak{m}_i - \mathfrak{m}_j\|^2_{\mathcal{H}_k}\right).$$

Term $\|\mathfrak{m}_i - \mathfrak{m}_j\|^2_{\mathcal{H}_k}$ can be thought of as a distance between empirical measures corresponding to bags i and j (this is empirical Maximum Mean Discrepancy (MMD)).

Kernel Methods – Discussion

- Kernel methods allows for very flexible and powerful machine learning models.
- **Nonparametric** method: parameter space (e.g., normal vector w in SVM) can be infinite-dimensional
- Kernels can be defined over more complex structures than vectors, e.g. graphs, strings, images, bags of instances, probability distributions.
- In naïve implementation, computational cost is at least quadratic in the number of observations, often $O(n^3)$ computation and $O(n^2)$ memory, but there are various approximations with good scaling up properties.
- Further reading:
  - Schölkopf and Smola, Learning with Kernels, 2001.
  - Rasmussen and Williams, Gaussian Processes for Machine Learning, 2006.
  - Steinwart and Christmann, Support Vector Machines, 2008.
  - Berlinet and Thomas-Agnan, Reproducing Kernel Hilbert Spaces in Probability and Statistics, 2004.
  - Bishop, Pattern Recognition and Machine Learning, Chapter 6.