Kernel Tricks, Means and Ends

Bernhard Schölkopf
Max Planck Institute for Biological Cybernetics
Tübingen, Germany

Empirical Inference Department
http://www.kyb.tuebingen.mpg.de/bs

Learning theory in a nutshell

Learn $f : \mathcal{X} \to \{\pm 1\}$ from examples

$(x_1, y_1), \ldots, (x_m, y_m) \in \mathcal{X} \times \{\pm 1\}$ generated from $\mathrm{P}(x, y)$

Goal: minimize expected error

$$R[f] = \int \frac{1}{2} |f(x) - y)| \, d\mathrm{P}(x, y)$$

V. Vapnik

Problem: P is unknown.

Induction principle: "empirical risk minimization"

$$R_{\mathrm{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2} |f(x_i) - y_i|$$

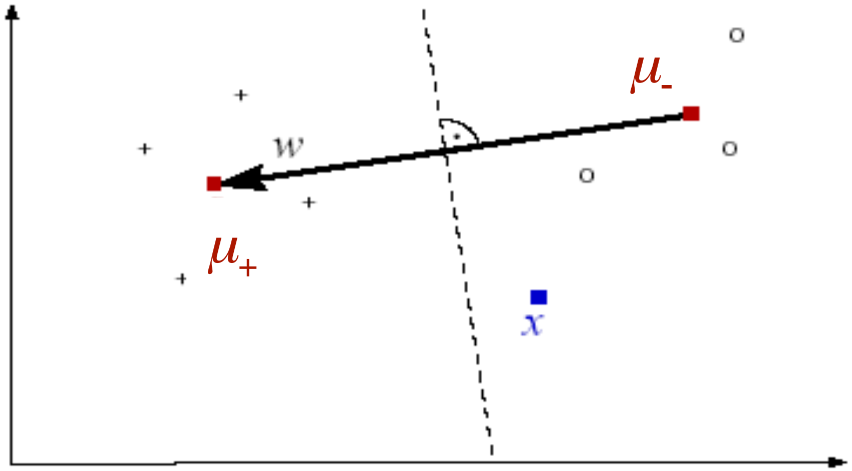
Vapnik & Chervonenkis: this is consistent* iff the "capacity" of the function class is asymptotically well-behaved (e.g., finite VC dim).

Computing the capacity is nontrivial…

Example of a Pattern Recognition Algorithm

Idea: classify points x according to which of the two class means is closer.

$$\mu_+ := \frac{1}{m_+} \sum_{y_i=1} x_i, \quad \mu_- := \frac{1}{m_-} \sum_{y_i=-1} x_i$$



- Decision function: hyperplane with normal vector $w := \mu_+ - \mu_-$
- How about problems that are not linearly separable?

Feature Spaces

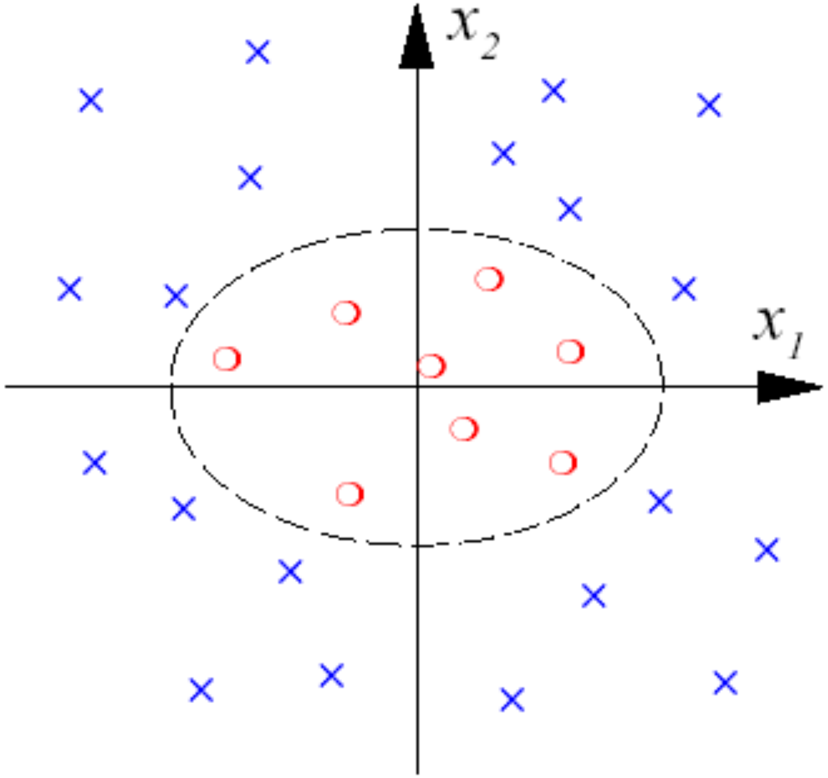
Preprocess the inputs with

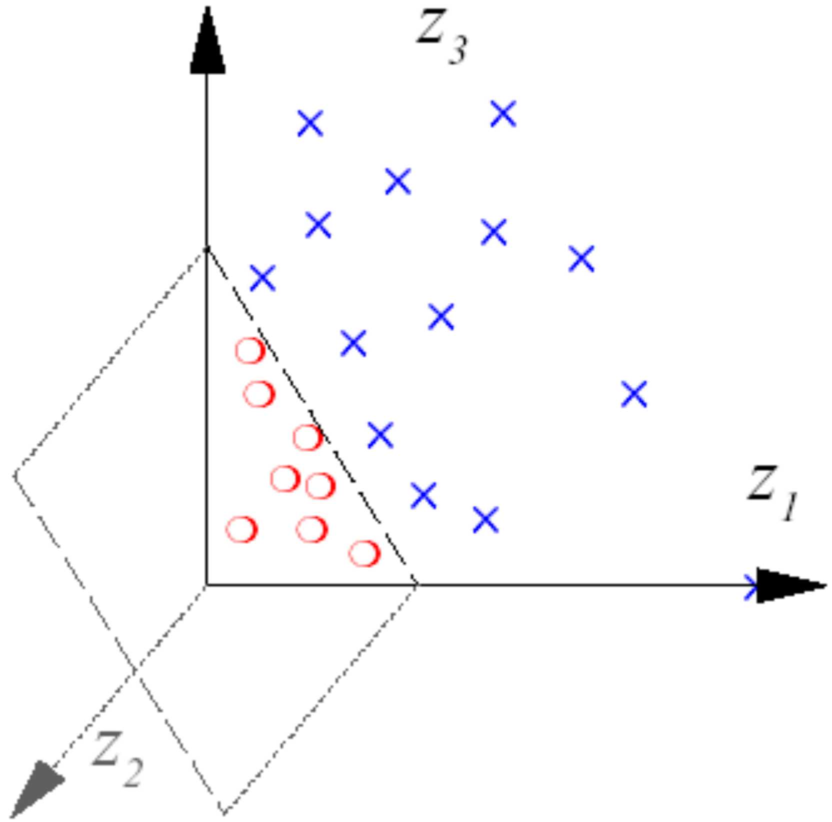
$$\begin{aligned}\Phi : \mathcal{X} &\to \mathcal{H} \\ x &\mapsto \Phi(x),\end{aligned}$$

where $\mathcal{H}$ is a dot product space, and learn the mapping from $\Phi(x)$ to y.

Example: All Degree 2 Monomials

$$\Phi : \mathbb{R}^2 \to \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$





The Kernel Trick

$$
\begin{aligned}
\langle \Phi(x), \Phi(x') \rangle &= (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)(x_1'^2, \sqrt{2}\, x_1' x_2', x_2'^2)^\top \\
&= (x_1 x_1' + x_2 x_2')^2 \\
&= \langle x, x' \rangle^2 \\
&= : k(x, x')
\end{aligned}
$$

$\longrightarrow$ the dot product in $\mathcal{H}$ can be computed from the dot product in $\mathbb{R}^2$

More generally: for $x, x' \in \mathbb{R}^N$, $d \in \mathbb{N}$,

$$
\langle x, x' \rangle^d = \left(\sum_{j=1}^{N} x_j \cdot x_j' \right)^d = \sum_{j_1, \ldots, j_d = 1}^{N} x_{j_1} \cdot \cdots \cdot x_{j_d} \cdot x_{j_1}' \cdot \cdots \cdot x_{j_d}' = \langle \Phi(x), \Phi(x') \rangle
$$

More generally: works for *positive definite kernels*

Positive Definite Kernels

Let $\mathcal{X}$ be a nonempty set. The following two are equivalent:

- k is *positive definite (pd)*, i.e., k is symmetric, and for
  - any set of training points $x_1, \ldots, x_m \in \mathcal{X}$ and
  - any $a_1, \ldots, a_m \in \mathbb{R}$

  we have

$$\sum_{i,j} a_i a_j K_{ij} \geq 0, \quad \text{where } K_{ij} := k(x_i, x_j)$$

- there exists a map Φ into a dot product space $\mathcal{H}$ such that

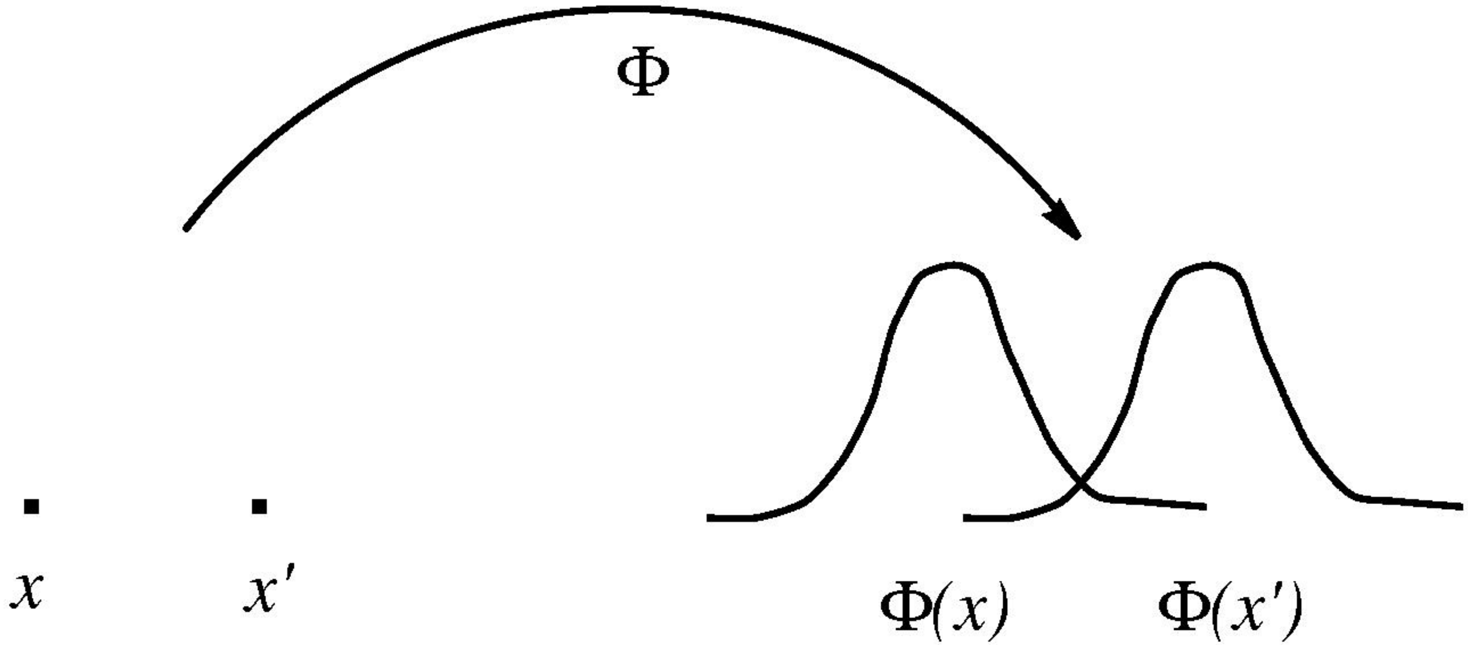
$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle$$

(RKHS)

$\mathcal{H}$ is a so-called *reproducing kernel Hilbert space.*

If for pairwise distinct points, $\Sigma=0$ iff all $a_i = 0$, call k *strictly p.d.*

Construction of Φ



$\Phi(x) := k(x,.)$ *(Aronszajn 1950),* take linear hull -> vector space

$< \Phi(x), \Phi(x') > := k(x,x')$, linear extension, can prove this is a dot product

Point evaluation: $f(x) = < f, k(x,.) >$. *"Reproducing kernel Hilbert space"*

The Kernel Trick – Main Points

- *any* algorithm that only depends on dot products can benefit from the kernel trick
- $\mathcal{X}$ need not be a vector space
- think of the kernel as a (nonlinear) *similarity measure*
- examples of common kernels:

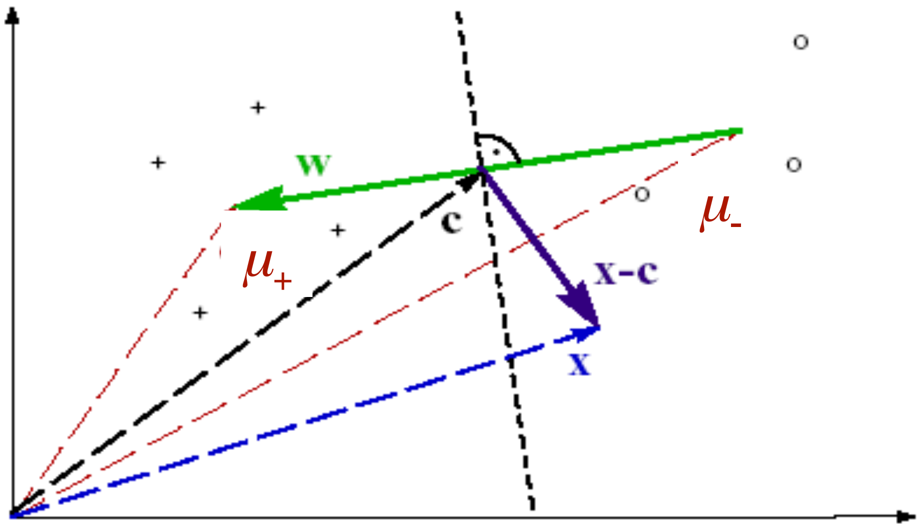
$$\text{Polynomial} \quad k(x,x') = (\langle x,x' \rangle + c)^d$$

$$\text{Gaussian} \quad k(x,x') = \exp(-\|x-x'\|^2/(2\,\sigma^2))$$

An Example of a Kernel Algorithm *(Schölkopf & Smola 2002)*

Classify points $\mathbf{x} := \Phi(x)$ *in feature space* according to which of the two class means is closer.

$$\mu_+ := \frac{1}{m_+} \sum_{\{i:y_i=1\}} \Phi(x_i), \quad \mu_- := \frac{1}{m_-} \sum_{\{i:y_i=-1\}} \Phi(x_i)$$



Compute the sign of the dot product between $\mathbf{w} := \mu_+ - \mu_-$ and $\mathbf{x} - \mathbf{c}$.

ctd.

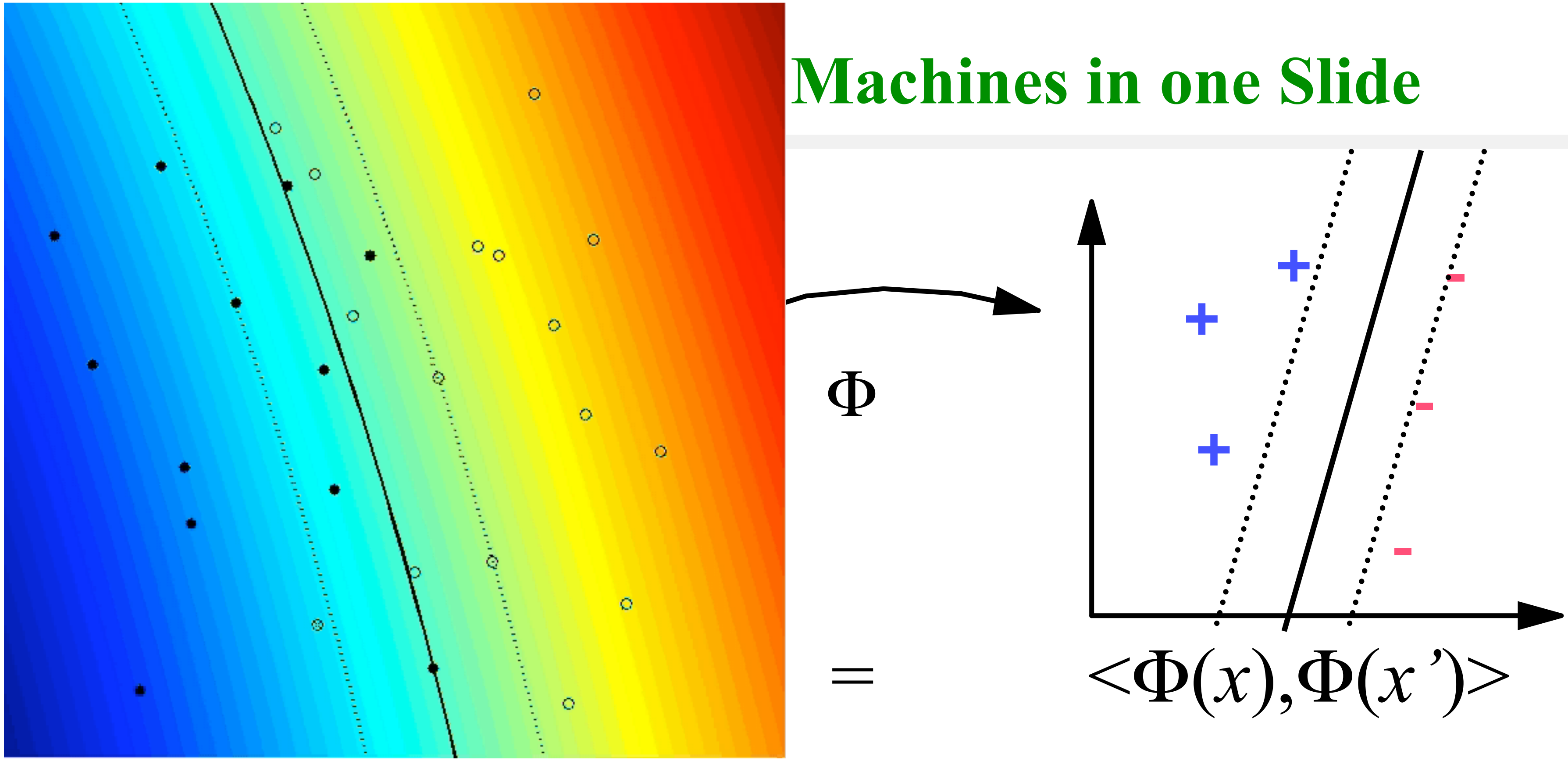
$$f(x) = \text{sgn}\left(\frac{1}{m_+}\sum_{\{i:y_i=1\}}\langle\Phi(x),\Phi(x_i)\rangle - \frac{1}{m_-}\sum_{\{i:y_i=-1\}}\langle\Phi(x),\Phi(x_i)\rangle + b\right)$$
$$= \text{sgn}\left(\frac{1}{m_+}\sum_{\{i:y_i=1\}} k(x,x_i) - \frac{1}{m_-}\sum_{\{i:y_i=-1\}} k(x,x_i) + b\right)$$

with the constant offset

$$b = \frac{1}{2}\left(\frac{1}{m_-^2}\sum_{\{(i,j):y_i=y_j=-1\}} k(x_i,x_j) - \frac{1}{m_+^2}\sum_{\{(i,j):y_i=y_j=1\}} k(x_i,x_j)\right).$$

If k is a density, this is a classifier based on *Parzen windows* plug-in estimates of the two classes.

Machines in one Slide



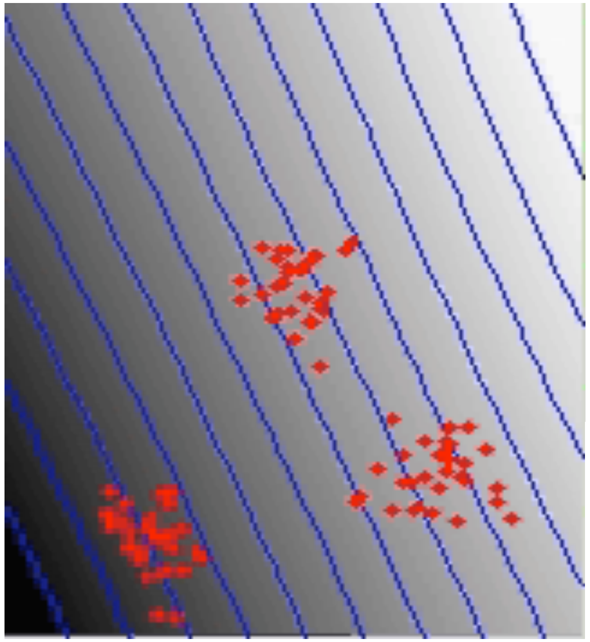
$$= \quad \langle \Phi(x), \Phi(x') \rangle$$

- $$f(x) = \mathrm{sgn}\left(\sum_i \lambda_i k(x_i, x) + b\right)$$

  representer theorem *(Kimeldorf & Wahba 1971, Schölkopf et al. 2000)*

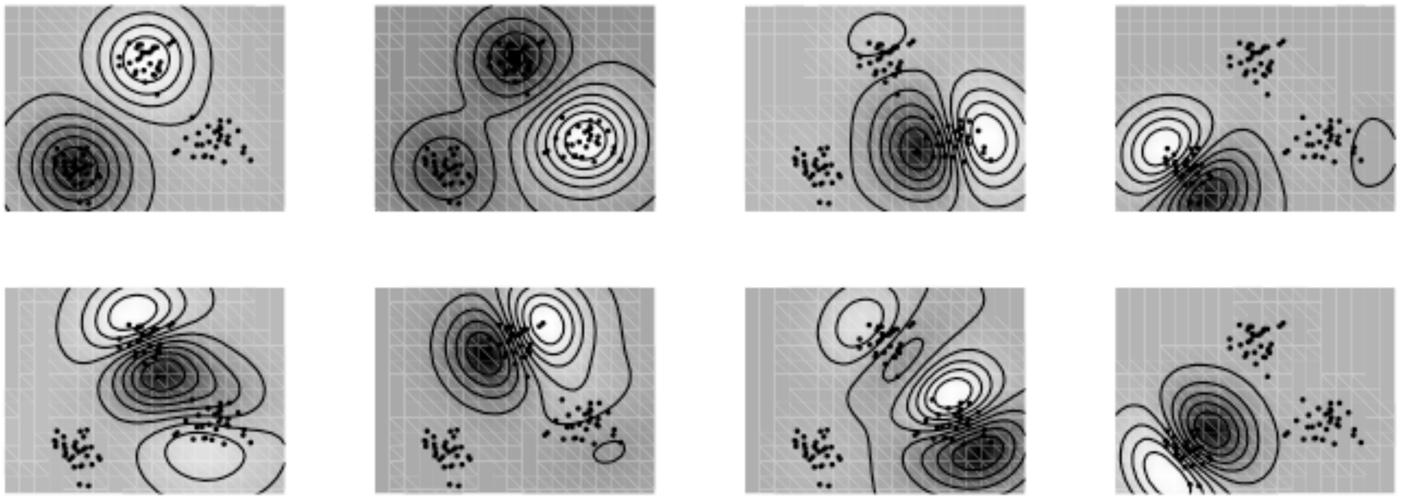
- unique solution found by convex QP

Kernel PCA



PCA in the RKHS

Contains LLE, Laplacian Eigenmap, and (in the limit) Isomap as special cases with data dependent kernels *(Ham et al. 2004)*

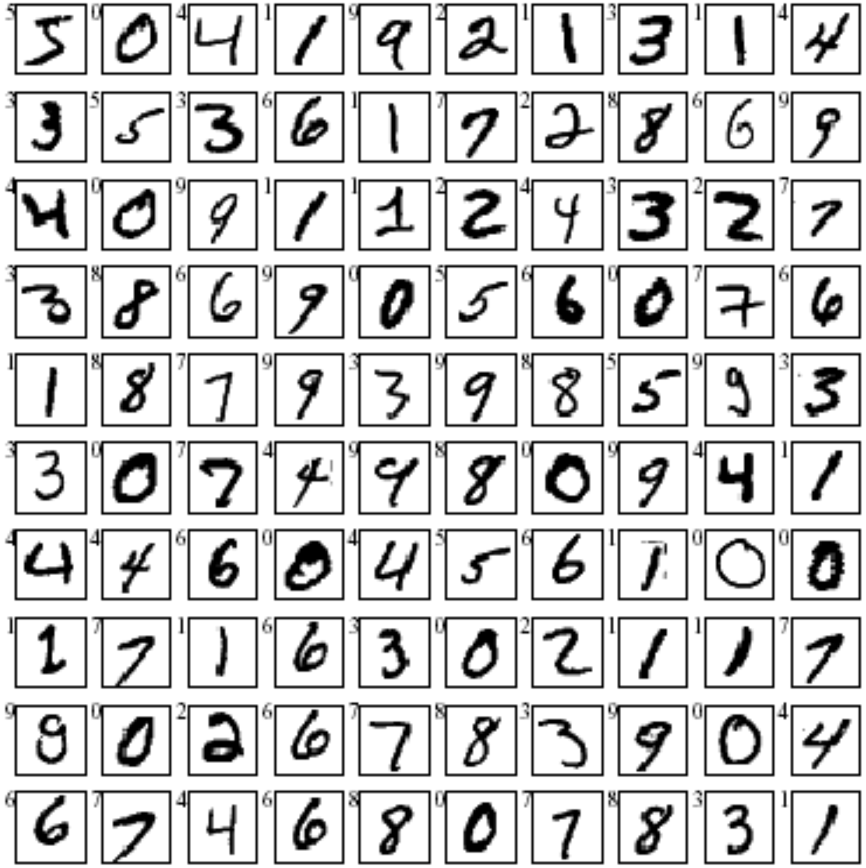


Schölkopf, Smola & Müller, 1998

Application examples

MNIST Benchmark

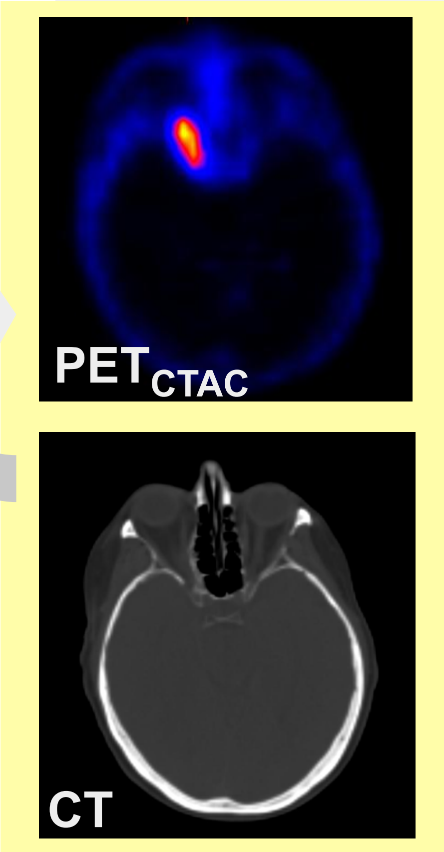
handwritten character benchmark (60000 training & 10000 test examples, 28×28)



MNIST Error Rates

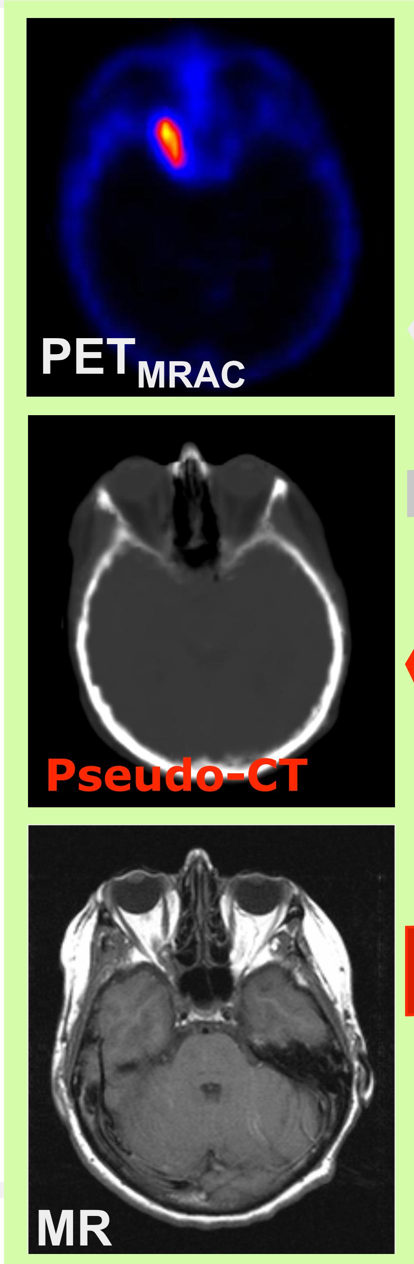
Classifier	test error	reference
linear classifier	8.4%	*Bottou et al. (1994)*
3-nearest-neighbour	2.4%	*Bottou et al. (1994)*
SVM	1.4%	*Burges and Schölkopf (1997)*
Tangent distance	1.1%	*Simard et al. (1993)*
LeNet4	1.1%	*LeCun et al. (1998)*
Boosted LeNet4	0.7%	*LeCun et al. (1998)*
Translation invariant SVM	0.56%	*DeCoste and Schölkopf (2002)*

PET attenuation correction



Visual Impression of PET_{MRAC} and PET_{CTAC} almost identical

Quantification Error below 1%



With M. Hofmann, B. Pichler, Radiologische Klinik, Tübingen

Tracer: [^{68}Ga]DOTA-TOC

Reconstruction performed on PET/CT Scanner, using Image Size 128x128, OSEM Reconstruction with 4 Iterations, 8 Subsets; Gaussian Filter FWHM 5 mm

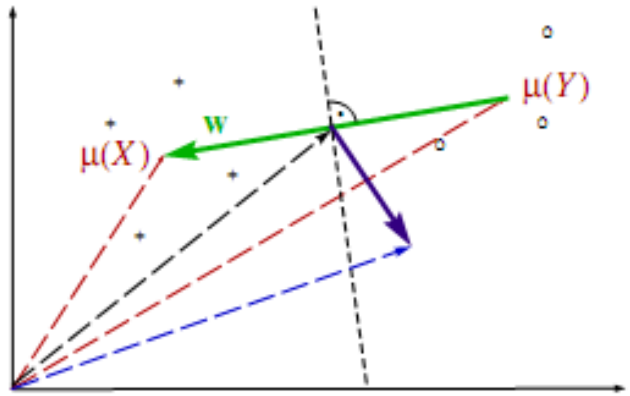
Learning of a Motor Primitive (Work in Progress)

MAX-PLANCK-GESELLSCHAFT

Kernel Means

Joint work with: K. Borgwardt, K. Fukumizu, A. Gretton, J. Huang, D. Janzing, Q. Le, M. Rasch, A. Smola, L. Song, B. Sriperumbudur, X. Sun

An example of a kernel algorithm, revisited



$\mathcal{X}$ compact subset of a separable metric space, $m, n \in \mathbb{N}$.

Positive class $X := \{x_1, \ldots, x_m\} \subset \mathcal{X}$
Negative class $Y := \{y_1, \ldots, y_n\} \subset \mathcal{X}$

RKHS means $\mu(X) = \frac{1}{m}\sum_{i=1}^{m} k(x_i, \cdot)$, $\mu(Y) = \frac{1}{n}\sum_{i=1}^{n} k(y_i, \cdot)$.

Get a problem if $\mu(X) = \mu(Y)$.

Schölkopf & Smola, 2002

When do the means coincide?

$k(x, x') = \langle x, x' \rangle$:	the means coincide
$k(x, x') = (\langle x, x' \rangle + 1)^d$:	all empirical moments up to order d coincide
k strictly pd:	$X = Y$.

The mean "remembers" each point that contributed to it.

Proposition 1 *Assume that k is strictly pd, and for all i, j, $x_i \neq x_j$, and $y_i \neq y_j$. If for some $\alpha_i, \beta_j \in \mathbb{R} - \{0\}$, we have*

$$\sum_{i=1}^{m} \alpha_i k(x_i, .) = \sum_{j=1}^{n} \beta_j k(y_j, .), \tag{1}$$

then $X = Y$.

Proof (by contradiction): W.l.o.g., assume that $x_1 \notin Y$. Subtract $\sum_{j=1}^{n} \beta_j k(y_j, .)$ from (1), and make it a sum over distinct points, to get

$$0 = \sum_i \gamma_i k(z_i, .),$$

where $z_1 = x_1, \gamma_1 = \alpha_1 \neq 0$, and $z_2, \dots \in X \cup Y - \{x_1\}$, $\gamma_2, \dots \in \mathbb{R}$.
Take the dot product with $\sum_j \gamma_j k(z_j, .)$, using $\langle k(z_i, .), k(z_j, .)\rangle = k(z_i, z_j)$, to get

$$0 = \sum_{ij} \gamma_i \gamma_j k(z_i, z_j),$$

with $\gamma \neq 0$, hence k cannot be strictly pd.

The mean map

$$\mu\colon X = (x_1, \ldots, x_m) \mapsto \frac{1}{m}\sum_{i=1}^{m} k(x_i, \cdot)$$

satisfies

$$\langle \mu(X), f \rangle = \left\langle \frac{1}{m}\sum_{i=1}^{m} k(x_i, \cdot), f \right\rangle = \frac{1}{m}\sum_{i=1}^{m} f(x_i)$$

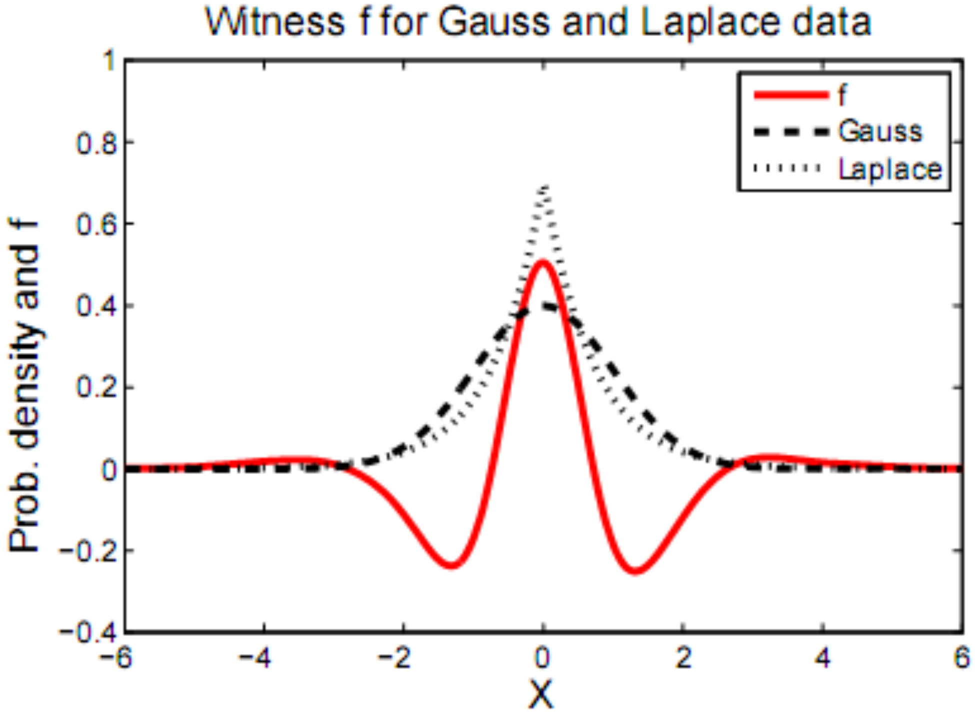
and

$$\|\mu(X) - \mu(Y)\| = \sup_{\|f\| \le 1} |\langle \mu(X) - \mu(Y), f \rangle| = \sup_{\|f\| \le 1} \left| \frac{1}{m}\sum_{i=1}^{m} f(x_i) - \frac{1}{n}\sum_{i=1}^{n} f(y_i) \right| .$$

Large distance ⇔ can find a function distinguishing the two samples

Witness function

$$f = \frac{\mu(X)-\mu(Y)}{\|\mu(X)-\mu(Y)\|}, \text{ thus } f(x) \propto \langle \mu(X) - \mu(Y), k(x,.)\rangle):$$



This function is in the RKHS of a Gaussian kernel, but not in the RKHS of the linear kernel.

The mean map for measures

p, q Borel probability measures,

$\mathbf{E}_{x,x'\sim p}[k(x,x')]$, $\mathbf{E}_{x,x'\sim q}[k(x,x')] < \infty$ ($\|k(x,.)\| \leq M < \infty$ is sufficient)

Define

$$\mu \colon p \mapsto \mathbf{E}_{x\sim p}[k(x,\cdot)].$$

Note

$$\langle \mu(p), f \rangle = \mathbf{E}_{x\sim p}[f(x)]$$

and

$$\|\mu(p) - \mu(q)\| = \sup_{\|f\|\leq 1} \left| \mathbf{E}_{x\sim p}[f(x)] - \mathbf{E}_{x\sim q}[f(x)] \right|.$$

Recall that in the finite sample case, for strictly p.d. kernels, μ was injective — how about now?

Smola et al., ALT'07, Fukumizu et al., NIPS'07

Theorem 2 *[Fortet and Mourier (1953); Dudley (2002)]*

$$p = q \iff \sup_{f \in C(\mathcal{X})} \left| \mathbf{E}_{x\sim p}(f(x)) - \mathbf{E}_{x\sim q}(f(x)) \right| = 0,$$

where $C(\mathcal{X})$ is the space of continuous bounded functions on $\mathcal{X}$.

Theorem 3 *[Gretton et al. (2007)] If k is universal, then*

$$p = q \iff \|\mu(p) - \mu(q)\| = 0.$$

Proof Idea: combine Theorem 2 with

$$\|\mu(p) - \mu(q)\| = \sup_{\|f\| \le 1} \left| \mathbf{E}_{x\sim p}[f(x)] - \mathbf{E}_{x\sim q}[f(x)] \right|$$

Replace $C(\mathcal{X})$ by the unit ball in an RKHS that is dense in $C(\mathcal{X})$ — universal kernel [51], e.g., Gaussian.

Discussion: solves a high-dim. optimization problem…

- μ is invertible on its image
  $\mathcal{M} = \{\mu(p) \mid p \text{ is a probability distribution}\}$
  (the "marginal polytope", *Wainwright and Jordan (2003)*)
- generalization of the *moment generating function* of a RV x with distribution p:

$$M_p(.) = \mathbf{E}_{x \sim p}\left[e^{\langle x, \cdot \rangle}\right].$$

- assume we have densities, the kernel is shift invariant, $k(x,y) = \phi(x-y)$, and all Fourier transforms exist. Note that μ is invertible iff

$$\int k(x-y)p(y)dx = \int k(x-y)q(y)dx \quad \Rightarrow \quad p = q$$

i.e.,

$$\hat{\phi}(\hat{p} - \hat{q}) = 0 \quad \Rightarrow \quad p = q$$

(Sriperumbudur et al., 2008)

Application 1: Two-sample problem *(Gretton et al., 2007)*

X, Y i.i.d. m-samples from p, q, respectively.

$$\begin{aligned}\|\mu(p)-\mu(q)\|^2 =& \mathbf{E}_{x,x'\sim p}\left[k(x,x')\right] - 2\mathbf{E}_{x\sim p, y\sim q}\left[k(x,y)\right] + \mathbf{E}_{y,y'\sim q}\left[k(y,y')\right] \\ =& \mathbf{E}_{x,x'\sim p, y,y'\sim q}\left[h((x,y),(x',y'))\right]\end{aligned}$$

with

$$h((x,y),(x',y')) := k(x,x') - k(x,y') - k(y,x') + k(y,y').$$

Define

$$D(p,q)^2 := \mathbf{E}_{x,x'\sim p,y,y'\sim q} h((x,y),(x',y'))$$
$$\hat{D}(X,Y)^2 := \frac{1}{m(m-1)} \sum_{i\neq j} h((x_i,y_i),(x_j,y_j)).$$

$\hat{D}(X,Y)^2$ is an unbiased estimator of $D(p,q)^2$.
It's easy to compute, and works on structured data.

Theorem 4 *Assume k is bounded.*
$\hat{D}(X,Y)^2$ converges to $D(p,q)^2$ in probability with rate $\mathcal{O}(m^{-\frac{1}{2}})$.

This *could* be used as a basis for a test, but uniform convergence bounds are often loose..

Theorem 5 *We assume $\mathbf{E}\left(h^2\right) < \infty$. When $p \neq q$, then $\sqrt{m}(\hat{D}(X,Y)^2 - D(p,q)^2)$ converges in distribution to a zero mean Gaussian with variance*

$$\sigma_u^2 = 4\left(\mathbf{E}_z\left[(\mathbf{E}_{z'}h(z,z'))^2\right] - \left[\mathbf{E}_{z,z'}(h(z,z'))\right]^2\right).$$

When $p = q$, then $m(\hat{D}(X,Y)^2 - D(p,q)^2) = m\hat{D}(X,Y)^2$ converges in distribution to

$$\sum_{l=1}^{\infty} \lambda_l \left[q_l^2 - 2\right], \qquad (2)$$

where $q_l \sim \mathcal{N}(0,2)$ i.i.d., λ_i are the solutions to the eigenvalue equation

$$\int_{\mathcal{X}} \tilde{k}(x,x')\psi_i(x)dp(x) = \lambda_i \psi_i(x'),$$

and $\tilde{k}(x_i,x_j) := k(x_i,x_j) - \mathbf{E}_x k(x_i,x) - \mathbf{E}_x k(x,x_j) + \mathbf{E}_{x,x'}k(x,x')$ is the centred RKHS kernel.

Application 2: Dependence Measures

Assume that (x, y) are drawn from p_{xy}, with marginals p_x, p_y.
Want to know whether p_{xy} factorizes into its marginals.

Bach and Jordan (2002); Fukumizu et al. (2004): kernel generalized variance
Gretton et al. (2005a,b): kernel constrained covariance, HSIC
Main idea *(Rényi, 1959; Jacod and Protter, 2000)*:
x and y independent $\iff$

$$\sup_{f,g \text{ bounded \& continuous}} \operatorname{Cov}(f(x), g(y)) = 0$$

Kernel version:

$$\sup_{f,g \in \text{unit balls in RKHS}} \operatorname{Cov}(f(x), g(y)) = 0$$

$\operatorname{cov}(x, y) := \mathbf{E}_{x,y}[xy] - \mathbf{E}_x[x]\mathbf{E}_y[y]$

k kernel on $\mathcal{X} \times \mathcal{Y}$.

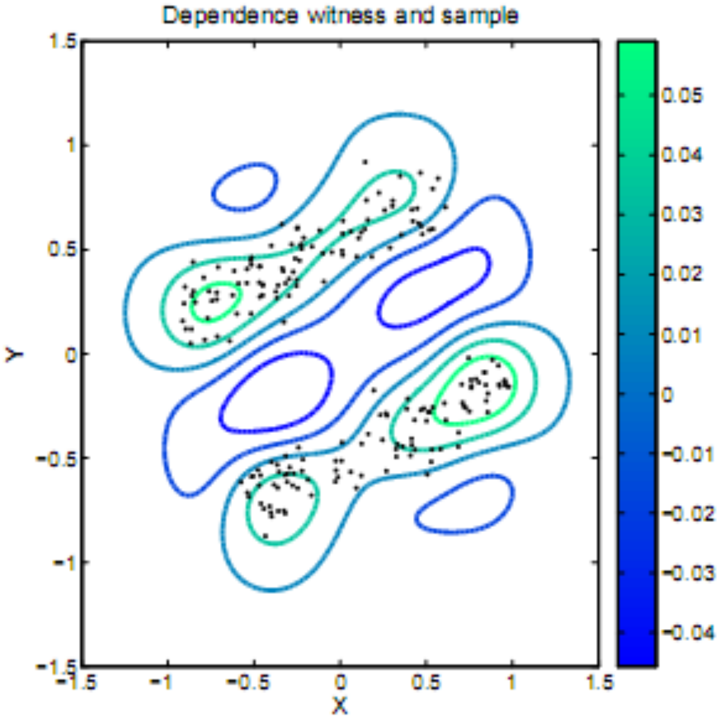
$$\mu(p_{xy}) := \mathbf{E}_{(x,y)\sim p_{xy}} [k((x,y),\cdot)]$$
$$\mu(p_x \times p_y) := \mathbf{E}_{x\sim p_x, y\sim p_y} [k((x,y),\cdot)] \, .$$

Use $\Delta := \left\| \mu(p_{xy}) - \mu(p_x \times p_y) \right\|$ as a measure of dependence.

For $k((x,y),(x',y')) = k_x(x,x')k_y(y,y')$:
Δ^2 equals the Hilbert-Schmidt norm of the covariance operator between the two RKHSs (HSIC), with empirical estimate $m^{-2} \operatorname{tr} H K_x H K_y$, where $H = I - \mathbf{1}/m$

Gretton et al. (2005a); Smola et al. (2007).

Witness function of the equivalent optimisation problem:



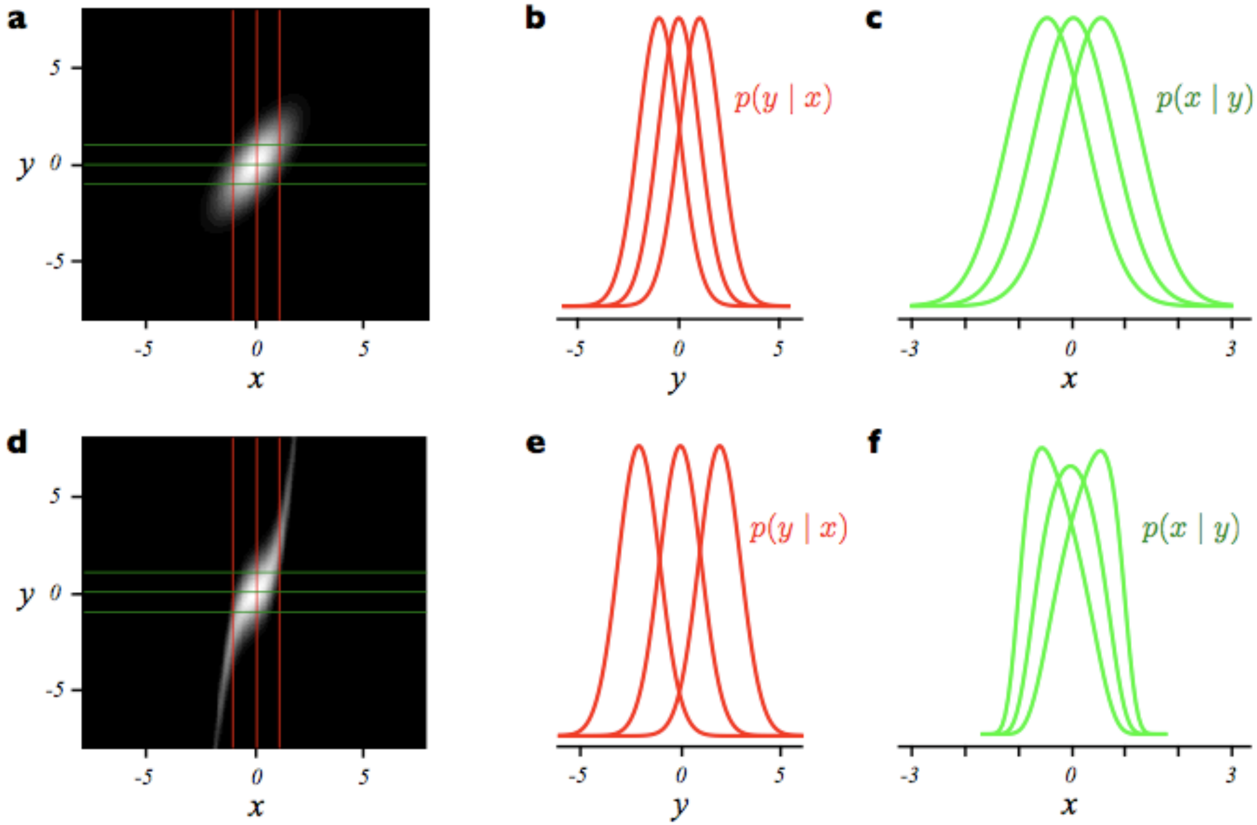
Application: learning causal structures *(Sun, Janzing, Schölkopf, Fukumizu, ICML 2007; Fukumizu, Gretton, Sun, Schölkopf, NIPS 2007)*

Causal Inference

Forward model: $y = f(x) + n$ with x, n independent.
Question: when is there a corresponding backward model?

$f(x)=x$

$f(x)=x+x^3$



Theorem 1 *Let the joint probability density of x and y be given by*

$$p(x,y) = p_n(y - f(x))p_x(x)\,, \tag{2}$$

where p_n, p_x are probability densities on $\mathbb{R}$. If there is a backward model of the same form, i.e.,

$$p(x,y) = p_{\tilde{n}}(x - g(y))p_y(y)\,, \tag{3}$$

then, denoting $\nu := \log p_n$ and $\xi := \log p_x$, the triple (f, p_x, p_n) must satisfy the following differential equation for all x, y with $\nu''(y - f(x))f'(x) \neq 0$:

$$\xi''' = \xi''\left(-\frac{\nu''' f'}{\nu''} + \frac{f''}{f'}\right) - 2\nu'' f'' f' + \nu' f''' + \frac{\nu' \nu''' f'' f'}{\nu''} - \frac{\nu' (f'')^2}{f'}\,, \tag{4}$$

where we have skipped the arguments $y - f(x)$, x, and x for ν, ξ, and f, respectively. Moreover, if for a fixed pair (f, ν) there exists $y \in \mathbb{R}$ such that $\nu''(y - f(x))f'(x) \neq 0$ for almost all $x \in \mathbb{R}$, the set of all p_x for which p has a backward model is contained in a 3-dimensional affine space.

A simple corollary is that if both the marginal density $p_x(x)$ and the noise density $p_n(y - f(x))$ are Gaussian then the existence of a backward model implies linearity of f:

Corollary 1 *Assume that $\nu''' = \xi''' = 0$ everywhere. If a backward model exists, then f is linear.*

(Hoyer, Janzing, Mooij, Peters, Schölkopf, 2008)

Application 3: Covariate Shift Correction and Local Learning

training set $X = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ drawn from p,
test set $X' = \{(x'_1, y'_1), \ldots, (x'_n, y'_n)\}$ from $p' \neq p$.

Assume $p_{y|x} = p'_{y|x}$.

Shimodaira (2000): reweight training set

Minimize

$$\left\| \sum_{i=1}^{m} \beta_i k(x_i, \cdot) - \mu(X') \right\|^2 + \lambda \|\beta\|_2^2 \quad \text{subject to} \quad \beta_i \geq 0, \ \sum_i \beta_i = 1.$$

Equivalent QP:

$$\underset{\beta}{\text{minimize}} \ \frac{1}{2}\beta^\top (K + \lambda \mathbf{1})\, \beta - \beta^\top l$$

$$\text{subject to } \beta_i \geq 0 \text{ and } \sum_i \beta_i = 1,$$

where $K_{ij} := k(x_i, x_j)$, $l_i = \langle k(x_i, \cdot), \mu(X') \rangle$.

Experiments show that in underspecified situations (e.g., large kernel widths), this helps *(Huang et al., 2007b)*.

$X' = \{x'\}$ leads to a local sample weighting scheme.

MAX-PLANCK-GESELLSCHAFT

Application 4: Measure estimation and dataset squashing *(Dudík et al., 2004; Smola et al., 2007)*

Given a sample X, minimize

$$\|\mu(X) - \mu(p)\|^2$$

over a convex combination of measures p_i,

$$p = \sum_i \alpha_i p_i, \quad \alpha_i \geq 0, \quad \sum_i \alpha_i = 1.$$

This can be written as a convex QP with objective function

$$\|\mu(X) - \mu(p)\|^2 = \alpha^\top Q\alpha + 1_m^\top K 1_m - 2\alpha^\top L 1_m,$$

where

$$\begin{aligned} L_{ij} &:= \mathbf{E}_{x\sim p_i}\left[k(x, x_j)\right] \\ Q_{ij} &:= \mathbf{E}_{x\sim p_i, x'\sim p_j}\left[k(x, x')\right] \\ K_{ij} &= k(x_i, x_j) \\ 1_m &:= (1/m, \ldots, 1/m)^\top \in \mathbb{R}^m. \end{aligned}$$

In practice, use

$$\alpha^\top [Q + \lambda I]\alpha - 2\alpha^\top L 1_m$$

Some cases where Q and L can be computed in closed form *(Smola et al., 2007)*:

- Gaussian p_i and k (cf. *Balakrishnan and Schonfeld (2006); Walder et al. (2007)*)
- X training set, Dirac measures $p_i = \delta_{x_i}$: dataset squashing, *DuMouchel et al. (1999)*
- X test set, Dirac measures $p_i = \delta_{y_i}$ centered on the training points Y: covariate shift correction *Huang et al. (2007a)*

Implicit Surface Fitting

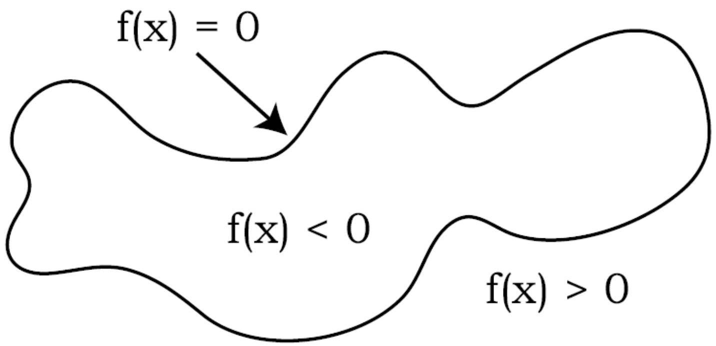
Given a sampling of a surface

$$\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_m \subset \mathbb{R}^d$$

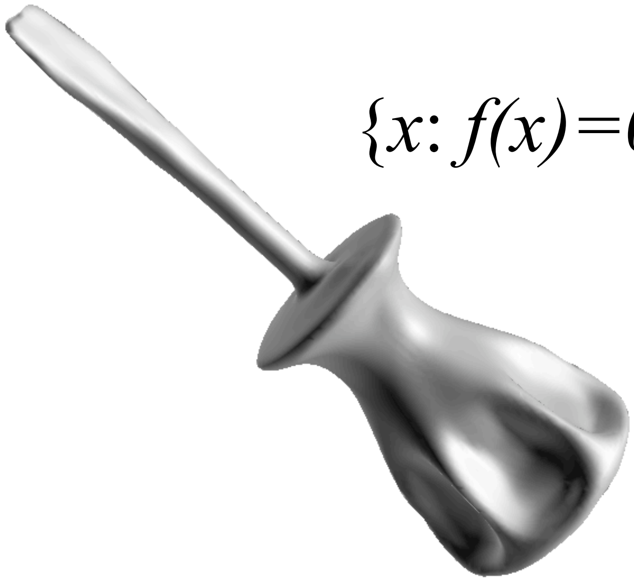
possibly with corresponding surface normals

$$\boldsymbol{n}_1, \boldsymbol{n}_2, \ldots \boldsymbol{n}_m \subset \mathbb{R}^d$$

Construct a function f whose zero level approximates the surface



SVM Implicit Surface Approximation



$\{x: f(x)=0\}$

$\min(f_1,f_2)$

Schölkopf, Giesen, & Spalinger, 2005
Walder, Chapelle, & Schölkopf, 2005

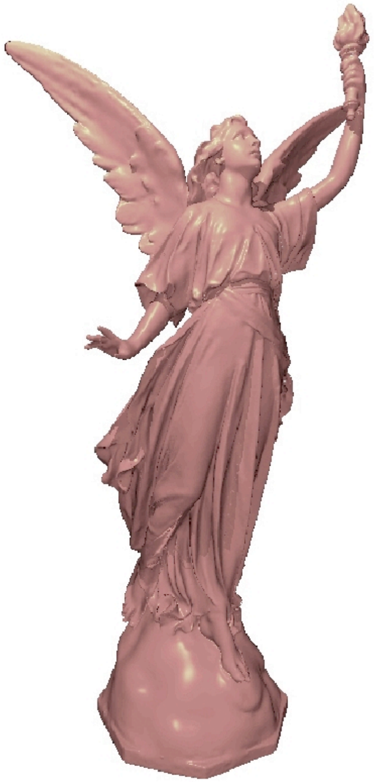
Steinke, Schölkopf, & Blanz, 2005

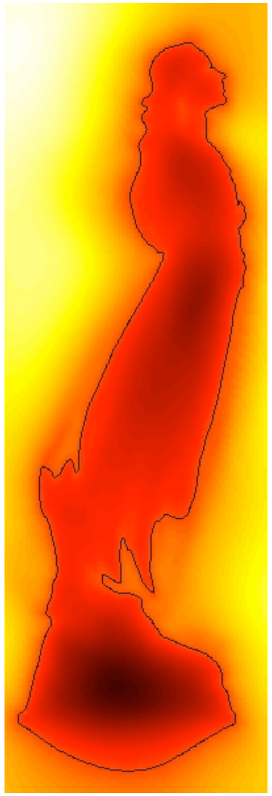
Signed distance functions f:

$|f(x)|$ = distance of x to the surface

$\text{sign}(f(x))=1$ iff x is outside the object

Large Scale Example *(Walder et al. 2006)*





Left: *Rendered model of Lucy, constructed from 14 million points with normals.*
Middle: *Each of the 364,982 basis function centres*
Right: *A planar slice that cuts the nose.*

More Examples

Dragon 1: 440K points – decreasing regularisation

Dragon 2: 3.6M points

Thai Statue: 5M points

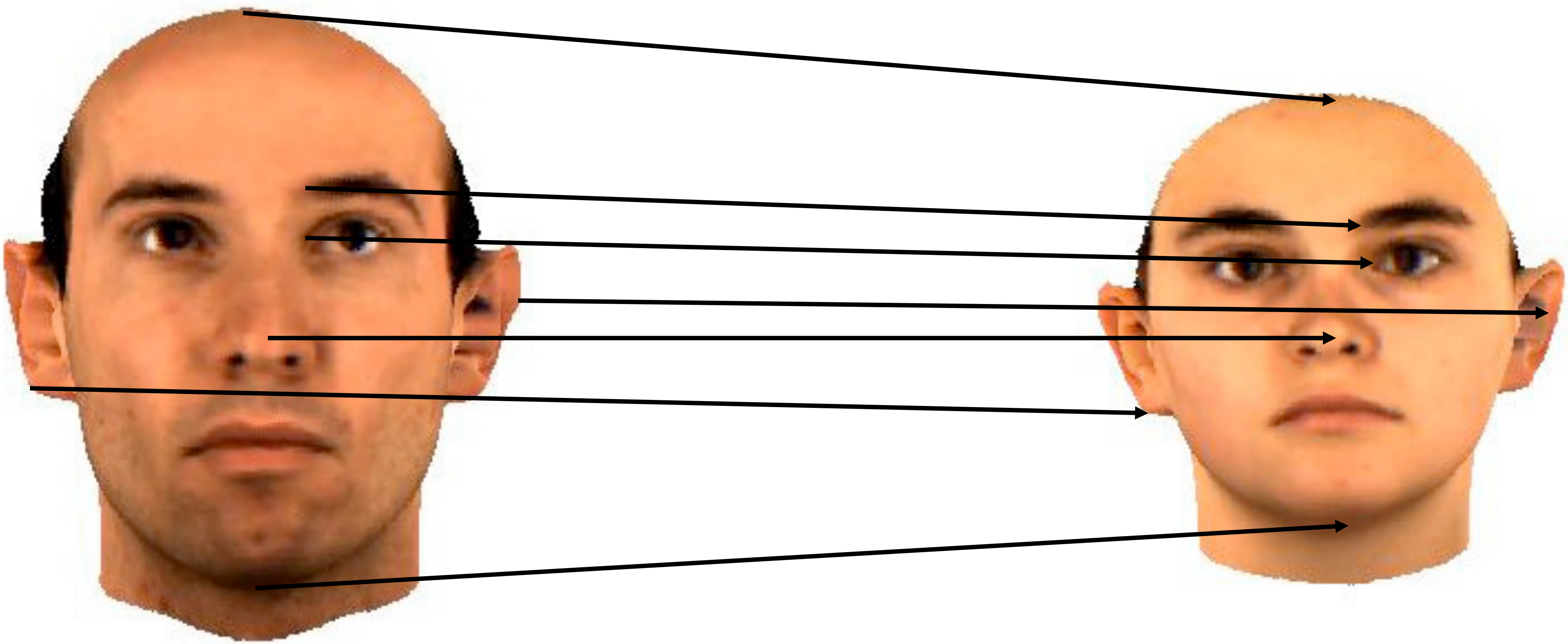
Interpolation in 4D

4D implicit. No data during red interval.

The Morphing Problem

I_1 $\frac{1}{2}(I_1 + I_2)$ I_2

Correspondence



Given a dense *correspondence field* (or *warp*), we can interpolate (and extrapolate) images, almost as in a linear space

(cf. Blanz & Vetter, 1999)

Correspondence via Machine Learning *(Schölkopf, Steinke, Blanz, 2005)*

- Objects O_1 and O_2 living in X. The warp is a mapping

  $$\tau : X \to X.$$

- Given surface points x_i of the O_1 and z_i of O_2.
- *If* they are in correspondence, we have a training set $(x_1,z_1),\ldots,(x_m,z_m)$ of "*landmark points*" and can do regression.
- What if they are **not** in correspondence?
- Main idea: τ should be such that

  O_1 relative to x "looks like" O_2 relative to $\tau(x)$

- Formalize this as a *locational cost*

  $$c\,(O_1, x, O_2, \tau\,(x))$$

Locational Cost Functions

$\textit{feature functions } f_1, f_2 : \quad \mathcal{X} \to \mathbb{R}$

think of f_1, f_2 as the signed distance functions of O_1, O_2.

1. $d(f_1(x), f_2(\tau(x)))^2$
2. $\sum_{i=0}^{\infty} \alpha_i d\left(\nabla^i f_1(x), \nabla^i f_2(\tau(x))\right)^2$
3. If Ψ is the feature map associated with a p.d. kernel on $(\mathcal{O} \times \mathcal{X}) \times (\mathcal{O} \times \mathcal{X})$.

   $\|\Psi(O_1, x) - \Psi(O_2, \tau(x))\|^2$

Optimization Problem

- Component functions: for $d=1,\ldots,D$,

$$\tau_d(x) = x_d + \langle \mathbf{w}_d, \Phi(x) \rangle$$

- Minimize

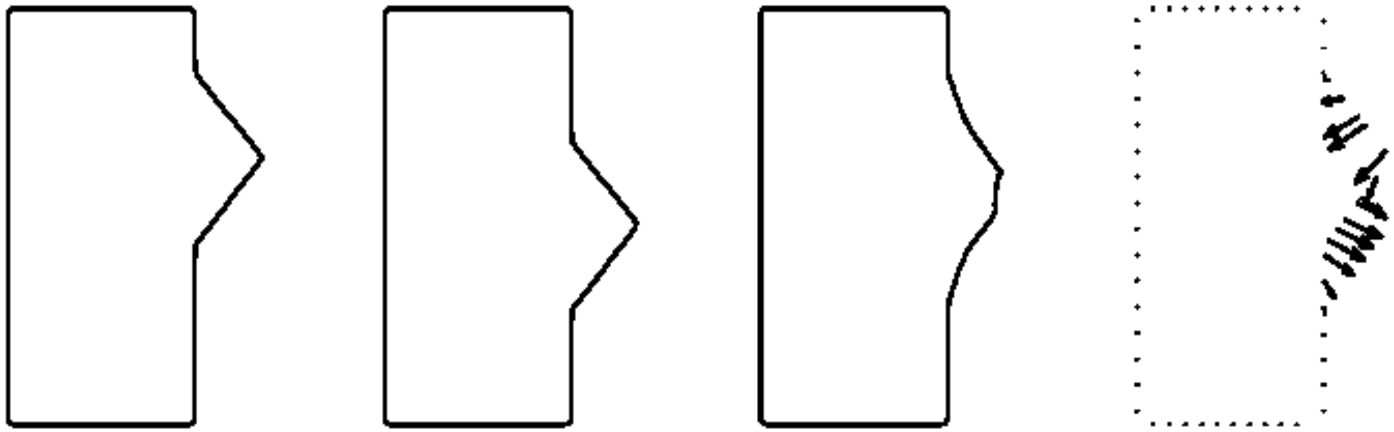
$$\frac{1}{2}\sum_{d=1}^{D} \|\mathbf{w}_d\|^2 + \lambda_p \sum_{i=1}^{m} \|\tau(x_i) - z_i\|^2$$

$$+\lambda_{loc} \int_{\mathcal{X}} c_{loc}(O_1, x, O_2, \tau(x))\, d\mu(x)$$

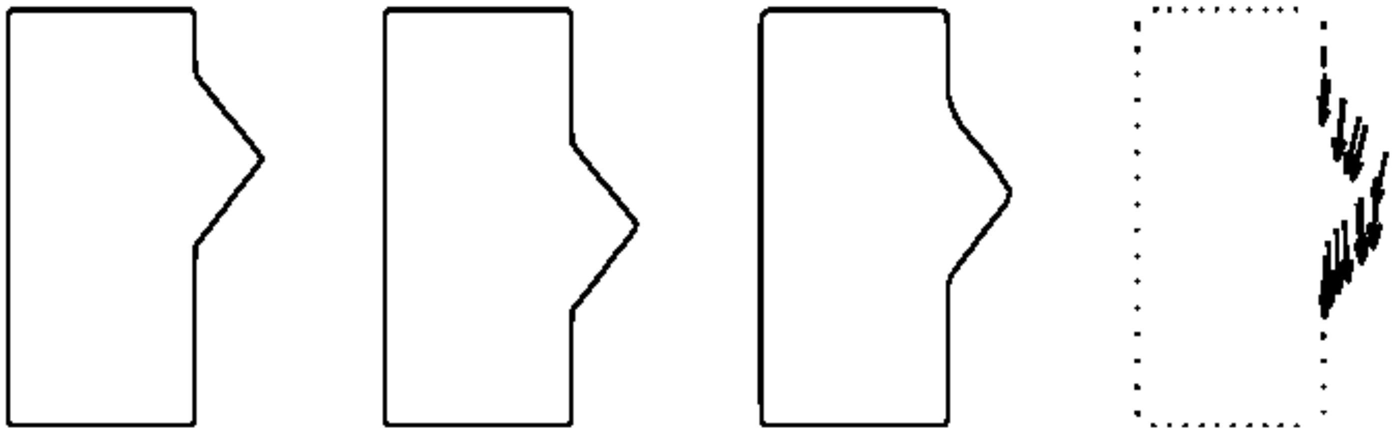
- For $\lambda_{loc} = 0$: D SVR problems with quadratic loss
- in the generic case, nonconvex

Toy Example

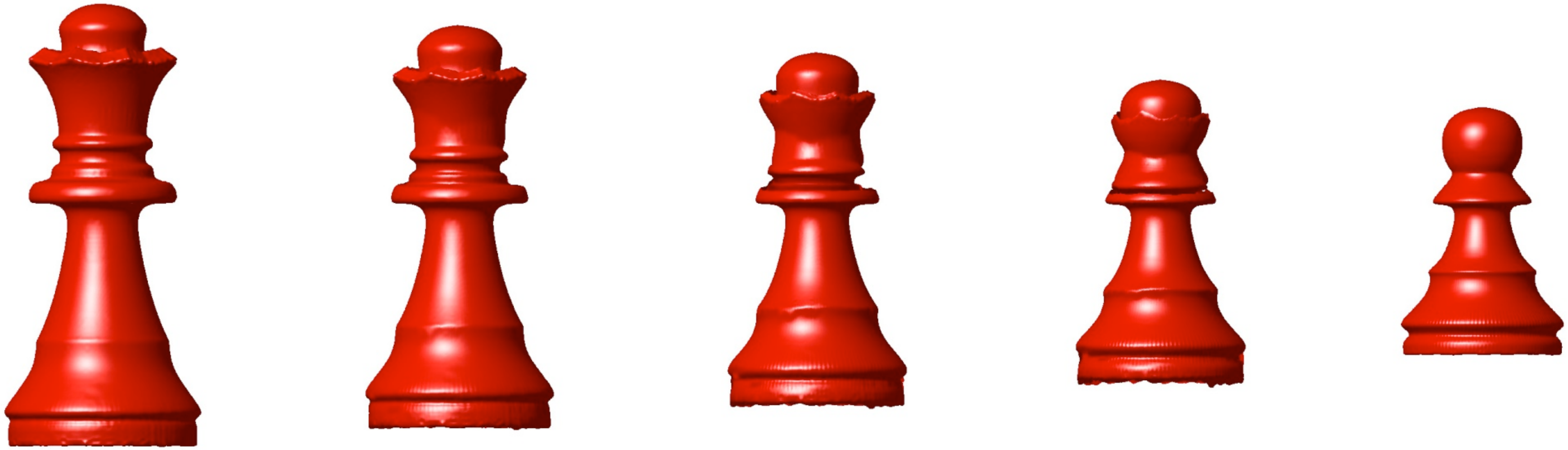
Signed distance



Signed distance + normals



Object Morphing



(signed distance and normals, no landmark points, no color information)

Head Morphing

Start

Target

(signed distan[…] […]olor information)

Steinke et al., NIPS 2006

with Dept. of Physiology, MPI for Biological Cybernetics

Walder et al., 2008


thank you for your attention
Bernhard Schölkopf, September 11, 2008