Hilbert Space Methods

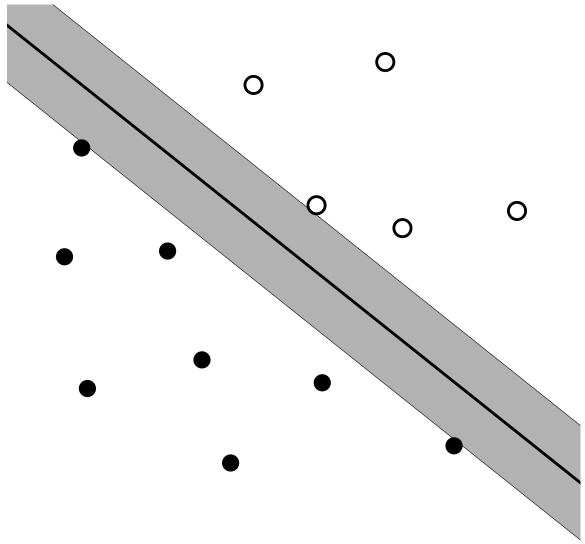
a unifying framework

Risi Kondor

Columbia University, New York.

Support Vector Machines

Learn $f : \mathcal{X} \mapsto \mathcal{Y}$ (e.g. $\mathcal{Y} = \{-1, +1\}$)
from $\{(x_1, y_1), \ldots, (x_m, y_m)\}$.



Kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ pos.def.

Feature map $\Phi : \mathcal{X} \mapsto \mathcal{F}$
obeys $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$

Gaussian Kernel:

$$k(x, x') = e^{-\| x - x' \|^2 / (2\sigma^2)}$$

$$f(x) = b + \sum_{i=1}^{m} \alpha_i \, k(x_i, x)$$

Regularized spaces of functions

Given $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ look for $f : \mathcal{X} \mapsto \mathcal{Y}$ in some linear space of functions $\mathcal{H}$ minimizing $R_{\text{reg}}[f]$.

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \| f \|_{\mathcal{H}}^2$$

Regularized spaces of functions

Given $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ look for $f : \mathcal{X} \mapsto \mathcal{Y}$ in some linear space of functions $\mathcal{H}$ minimizing $R_{\text{reg}}[f]$.

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \| f \|_{\mathcal{H}}^2$$

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \langle f, f \rangle_{\mathcal{H}}$$ Hilbert space

Hilbert space

An **inner product** $\langle \cdot, \cdot \rangle$ is a function $\mathcal{H} \times \mathcal{H} \mapsto \mathbb{R}$ that satisfies

1. $\langle \alpha a + \beta b \, , \, c \rangle = \alpha \langle a, c \rangle + \beta \langle b, c \rangle$
2. $\langle a \, , \, \beta b + \gamma c \rangle = \beta \langle a, b \rangle + \gamma \langle a, c \rangle$
3. $\langle a, b \rangle = \langle b, a \rangle$
4. $\langle a, b \rangle \geq 0$
5. $\langle a, a \rangle = 0$ if and only if $a = 0$.

A **Hilbert space** is a linear space $\mathcal{H}$ endowed with an inner product $\langle \cdot, \cdot \rangle$ that is complete with respect to the norm $\| \, a \, \| = \langle a, a \rangle^{1/2}$.

Regularized spaces of functions

Given $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ look for $f : \mathcal{X} \mapsto \mathcal{Y}$ in some linear space of functions $\mathcal{H}$ minimizing $R_{\text{reg}}[f]$.

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \| f \|_{\mathcal{H}}^2$$

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \langle f, f \rangle_{\mathcal{H}}$$

Hilbert space

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(\langle f, k_{x_i} \rangle, y_i) + \langle f, f \rangle_{\mathcal{H}}$$

RKHS

Regularized spaces of functions

Given $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ look for $f : \mathcal{X} \mapsto \mathcal{Y}$ in some linear space of functions $\mathcal{H}$ minimizing $R_{\text{reg}}[f]$.

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \| \, f \, \|_{\mathcal{H}}^2$$

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \langle f, f \rangle_{\mathcal{H}}$$ Hilbert space

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(\langle f, k_{x_i} \rangle \, , y_i) + \langle f, f \rangle_{\mathcal{H}}$$ RKHS

The k_x are “prototypical” functions s.t. $f(x) = \langle f, k_x \rangle$. Defining $\{ k_x \mid x \in \mathcal{X} \}$ sets $\langle k_x, k_{x'} \rangle = k_x(x')$: in fact, $k_x(x') = k(x, x')$ turns out to be the kernel.

$$f = \sum_{i=1}^{m} \alpha_i \, k_{x_i} \quad \Longrightarrow \quad f(x) = \sum_{i=1}^{m} \alpha_i \, k(x_i, x)$$

Reproducing kernel Hilbert spaces (RKHS)

Let $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be a positive definite function. We construct the corresponding RKHS by:

1. Building a linear space of functions spanned by $\{ k_x = k(x, \cdot) \mid x \in \mathcal{X} \}$
2. Defining the inner product as $\langle k_x, k_{x'} \rangle = k(x, x')$ and extending it to the whole space
3. Completing the space with respect to the norm $\| x \| = \langle x, x \rangle^{1/2}$

Note that for any $f = \sum_i \alpha_i k_{x_i}$, (2) implies

$$\langle f, k_x \rangle = \sum_i \alpha_i \langle k_{x_i}, x \rangle = \sum_i \alpha_i \, k(x_i, x) = \sum_i \alpha_i \, k_{x_i}(x) = f(x),$$

hence in general, $\langle f, k_x \rangle = f(x)$. This is the reproducing property.

General form of Hilbert space learning

Given data $\{z_1, z_2, \ldots, z_m\}$, a Hilbert space $\mathcal{H}$ and a mapping $\Phi : z_i \mapsto \{(\phi_i^{(1)}, t_i^{(1)}), \ldots (\phi_i^{(r_i)}, t_i^{(r_i)})\}$, learn $f \in \mathcal{H}$ minimizing

$$R_{\text{reg}}[f] = \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{r_i} L(\langle f, \phi_i^{(j)} \rangle, t_i^{(j)}) + \langle f, f \rangle$$

for some loss function L. Solution will be of the form

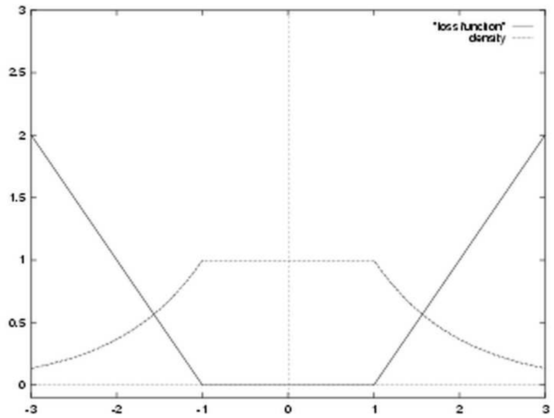
$$f = \sum_{i=1}^{m} \sum_{j=1}^{r_i} \alpha_i^{(j)} \phi_i^{(j)}.$$

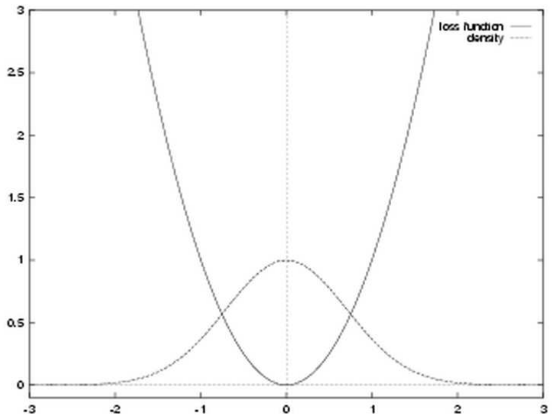
For conventional kernel methods $\Phi : (x_i, y_i) \mapsto (k_{x_i}, y_i)$,

$$f = \sum_{i=1}^{m} \alpha_i \, k_{x_i}.$$

The kernel and the loss function

The kernel k selects the regularization scheme.
The loss function L selects the algorithm.





- SVM classification: hinge loss
- SVM regression: ϵ-insensitive loss
- Gaussian processes: squared loss

All algorithms lead to some QP, LP or just matrix inversion problem.

“Reductions” to Hilbert space methods 1.

Generalizing the input space $\mathcal{X}$:

- Sequences: string kernels [Watkins 2000] [Leslie et. al. 2002] [Vishwanathan & Smola 2002]
- Trees: tree kernels [Collins & Duffy 2001]
- Graphs: [Tsuda 2003] [Gaertner 2003]
- Automata: rational kernels [Cortes, Haffner & Mohri 2002]
- Vertices of graphs and points on manifolds: diffusion kernels [Kondor & Lafferty 2002] [Smola & Kondor 2003] [Belkin & Niyogi 2001, 2002]
- Elements of groups: via Fourier analysis [Kondor 2002]
- Distributions: Fisher kernel [Jaakkola & Haussler 1999], information diffusion kernels [Lafferty & Lebanon 2002], Bhattacharyya kernels [Jebara & Kondor 2003]

“Reductions” to Hilbert space methods 2.

Generalizing the output space $\mathcal{Y}$:

- Multiclass: [Crammer and Singer 2001]
- Ranking: [Herbrich et. al. 2000]
- Learning functions: KDE [Weston et. al. 2002]
- Tagging and parsing: [Collins and Duffy 2001], HM-SVM [Alton, Tsochantaridis and Hofmann 2002]

Changing the problem:

- Learning from functionals
- Multi-facet learning

Learning from functionals (derivatives)

Let $\mathcal{O} : \mathcal{H} \mapsto \mathcal{H}$ be a linear operator. For each x_i, instead of $y_i = f(x_i)$, assume we are given $z_i = (\mathcal{O} f)(x_i)$.

Example: derivatives, $\mathcal{O} = \frac{d}{dx}$. We now want a loss of the form

$$L((\tfrac{d}{dx} f)(x_i), z_i) = L(\tfrac{d}{dx_i} \langle f, k_{x_i} \rangle, z_i) = L(\langle f, \phi_{x_i} \rangle, z_i)$$

where $\phi_x(x') = \frac{\partial}{\partial x} k_x(x')$. For mixed case, kernel expansion of solution will be of the form

$$f = \sum_{i=1}^{m} \alpha_i \, k_{x_i} + \sum_{i=1}^{m} \alpha_i^* \, \phi_{x_i}.$$

All we need to run algorithm is to be able to compute terms like $\langle k_x, k_{x'} \rangle$, $\langle k_x, \phi_{x'} \rangle$ and $\langle \phi_x, \phi_{x'} \rangle$.

Smola, Schölkopf: On a Kernel-Based Method for Pattern Recognition, Regression, Approximation and Operator Inversion. Algorithmica 1998.

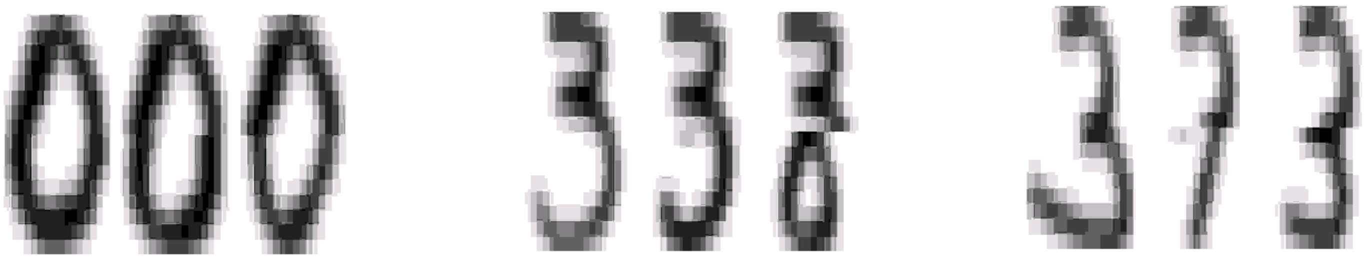
Kernel Dependency Estimation

Learn general mapping from $\mathcal{X}$ to $\mathcal{Y}$!

Separate kernels $k^{\mathcal{X}}$ and $k^{\mathcal{Y}}$ on $\mathcal{X}$ and $\mathcal{Y}$ inducing Hilbert spaces $\mathcal{H}_{\mathcal{X}}$ and $\mathcal{H}_{\mathcal{Y}}$.

Do KPCA in $\mathcal{H}_{\mathcal{Y}}$, take first r components, and learn r separate regression estimates $\mathcal{X} \mapsto \mathbb{R}$ with kernel method of choice.

Suffers from pre-image problem.



Weston, Chapelle, Elisseeff, Schölkopf and Vapnik: Kernel Dependency Estimation (NIPS 2002)

Kernel Dependency Estimation

Can be thought of as learning $f \in \mathcal{H}_{\mathcal{X}} \otimes \mathcal{H}_{\mathcal{Y}}$. The corresponding kernel is

$$k((x,y),(x',y')) = k^{\mathcal{X}}(x,x') \cdot k^{\mathcal{Y}}(y,y').$$

Given KPCA components $\phi_1, \phi_2, \ldots, \phi_r$, the loss on a single digit (x,y) is

$$\sum_{i=1}^{r} L(\langle f, k_x \otimes \phi_i \rangle_{\mathcal{H}}, \langle k_y, \phi_i \rangle_{\mathcal{H}_{\mathcal{Y}}}).$$

To evaluate for an example, project f onto the subspace spanned by

$$\{\, k_x \otimes \phi_1, k_x \otimes \phi_2, \ldots, k_x \otimes \phi_r \,\}$$

Multiclass

When $\mathcal{Y} = \{1, 2, \ldots, n\},$ the natural SVM-like formulation is

$$F(x_i, y_i) - \max_{y \neq y_i} F(x_i, y) \geq 1 - \eta$$

for some slack variable $\eta \geq 0$.

$$F(x_i, y_i) = \left\langle f, k_{(x,y)} \right\rangle$$

Again, joint input-output space $\mathcal{Z} = \mathcal{X} \oplus \mathcal{Y}$ and joint kernel $k : \mathcal{Z} \times \mathcal{Z} \mapsto \mathbb{R}$.

Collins did something similar with the perceptron [Collins and Duffy, 2001].

Alton, Tsochantaridis and Hofmann: Hidden Markov Support Vector Machines (ICML 2003)

Hidden Markov Suppoert Vector Machines

For tagging natural sentences, $x = x^1 \ldots x^p$ is a sequence of words, $y = y^1 \ldots y^p$ a corresponding sequence of tags. Two types of features:

$$\phi_{rq}^{st} = [\![y^t = q]\!][\![x^s = r]\!]$$

$$\overline{\phi}_{uv}^{st} = [\![y^t = u]\!][\![y^s = v]\!].$$

Construct kernel on $\mathcal{X} \oplus \mathcal{Y}$

$$k((x,y),(x',y')) = \sum_{strq} \phi_{rq}^{st}(x) \cdot \phi_{rq}^{st}(x') + \sum_{stuv} \overline{\phi}_{uv}^{st}(x) \cdot \overline{\phi}_{uv}^{st}(x')$$

Not just a product kernel!!!

For decoding use Viterbi.

Alton, Tsochantaridis and Hofmann: Hidden Markov Support Vector Machines (ICML 2003)

Joint input-output space with common kernel

- Interesting idea, many possibilities.
- Still lead to familiar QP, LP problems but maybe of higher dimensionality.
- Joint probability model on $\mathcal{X} \oplus \mathcal{Y}$?
- Maybe combining evidence from multiple sources, multiple directions.
- ...
- ...

Alex Smola, private communications

Extensions

Beyond Hilbert space:

- Matthias Hein and Olivier Bousquet: **Maximal Margin Classification for Metric Spaces**
- Ulrike von Luxburg and Olivier Bousquet: **Distance-Based Classification with Lipschitz Functions**

More general models in $\mathcal{X} \oplus \mathcal{Y}$:

- Alex Smola et. al.: **to appear**