Hilbert Space Methods in Learning

guest lecturer: Risi Kondor

6772 Advanced Machine Learning and Perception (Jebara),
Columbia University, October 15, 2003.

1. A general formulation of the learning problem
   - Empirical and true errors — over£tting
   - Error bounds and what they tell us about the design of algorithms

2. Hilbert space methods
   - Reproducing Kernel Hilbert Spaces
   - Kernels
   - Algorithms: SVM, Gaussian Processes, Kernel PCA

Tutorial online at

`http://www.cs.columbia.edu/risi/notes/tutorial6672.pdf`

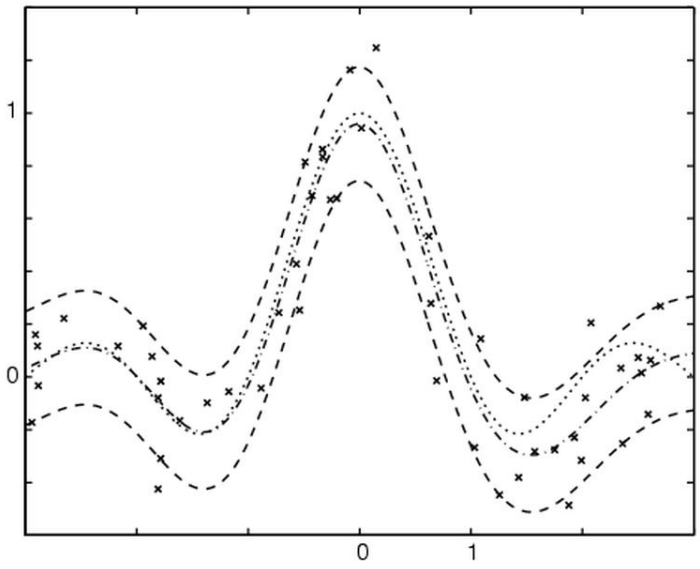
The Learning Problem

Regression

Learn a function $f : x \mapsto y$

Linear functions, order p polynomials, splines, etc.

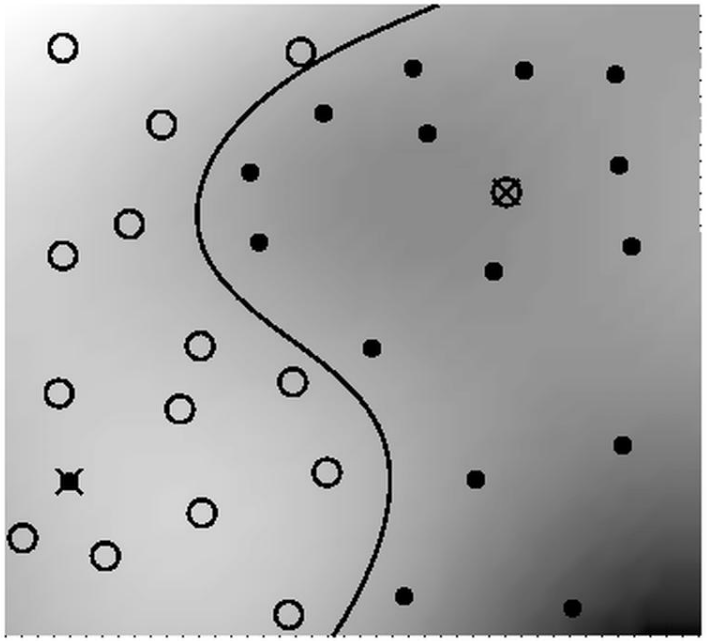
Examples:
Boston housing problem, robot grasps, motorcycle data, etc.



Classi£cation

Separate $+1$ labeled points from -1 labeled points

Examples:
Face recognition, DNA splice site identi£cation, document classi£cation, call type classi£cation



Supervised learning

Input space: $\mathcal{X}$ e.g. $\mathcal{X} = \mathbb{R}^n$

Output space: $\mathcal{Y}$ $\mathcal{Y} = \{-1, +1\}$ for classi£cation

$\mathcal{Y} = \mathbb{R}$ for regression

Training set: $S = (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$ $x_i \in \mathcal{X}, \quad y_t \in \mathcal{Y}$

“Truth”:

- Deterministic: $y = f_0(x)$
- Probabilistic: $y \sim p(y \mid x)$ (more general)

Goal: construct **hypothesis** $f : \mathcal{X} \mapsto \mathcal{Y}$ to predict y given x.

The Empirical Risk

Empirical risk (training error):

$$R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i)$$

where $L : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ is the **loss function**.

Zero-one loss for classi£cation: $L(\hat{y}, y) = \begin{cases} 1 & \text{if } \hat{y} \neq y \\ 0 & \text{otherwise.} \end{cases}$

Squared error loss for regression: $L(\hat{y}, y) = (y - \hat{y})^2$.

A Bad Learning Algorithm (memorization algorithm)

Set

$$f(x) = \begin{cases} 1 & \text{when } x = x_i \text{ and } y_i = 1 \\ -1 & \text{otherwise.} \end{cases}$$

For zero-one loss perfect performance on training data!

$$R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) = 0$$

Will it **generalize** well to testing examples? Why not?

The True Risk

- Assume some distribution on inputs: $p(x)$
- Distribution on (x, y) examples: $p(x, y) = p\,(\,y \,|\, x\,)\,p(x)$ or $p(x, y) = \delta(y - f_0(x))\,p(x)$

True risk:

$$R[f] = \mathbb{E}\,[L(f(x), y)] = \int_{\mathcal{X}\times\mathcal{Y}} p(x, y)\,L(f(x), y)\,dx\,dy.$$

This is what we really want to minimize in **discriminative** learning.

True Risk vs. Empirical Risk

$$R[f] = \mathbb{E}\left[L(f(x), y)\right] \qquad \longleftrightarrow \qquad R_{\text{emp}}[f] = \frac{1}{m}\sum_{i=1}^{m} L(f(x_i), y_i).$$

Just minimizing R_{emp} is BAD (see previous algorithm). Optimizing the training error at the expense of the testing error is called **over£tting**.

But we do not know $p(x, y)$!!! Can we still do anything?

Bounding the True Risk

For many practical learning algorithms

$$R[f] = \mathbb{E}\left[L(f(x), y)\right] \qquad \sim \qquad R_{\text{emp}}[f] = \frac{1}{m}\sum_{i=1}^{m} L(f(x_i), y_i).$$

Uniform error bounds:

For any distribution D, with probability $1-\delta$ (over the choice of training set)

$$|\, R[f] - R_{\text{emp}}[f] \,| \leq \epsilon$$

for all hypotheses $f \in \mathcal{F}$ simultaneously.

“PAC” bound: probably approximately correct

KEY CONCEPT: Capacity Control

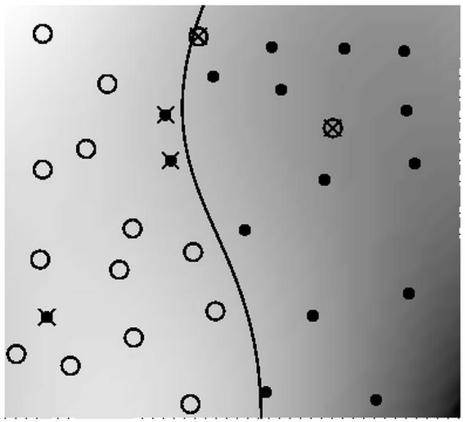
$$\mathbb{P}\Big[\ \big|R[f] - R_{\mathsf{emp}}[f]\big| \leq \epsilon\ \Big] \geq 1 - \delta$$

Generally, ϵ is a complicated function of δ depending crucially on $\mathcal{F}$ (**hypothesis class**) that f is chosen from.

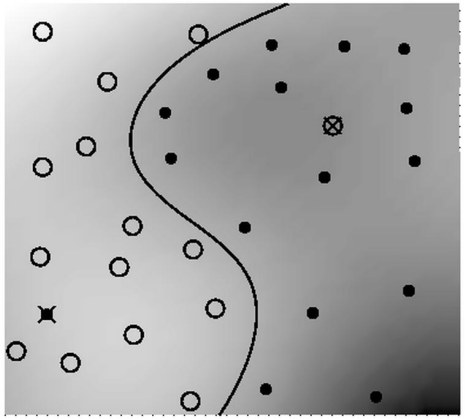
Compromise:

model ¤exibility		generalization performance
large $\mathcal{F}$	$\longleftrightarrow$	want small ϵ
complexity		generality

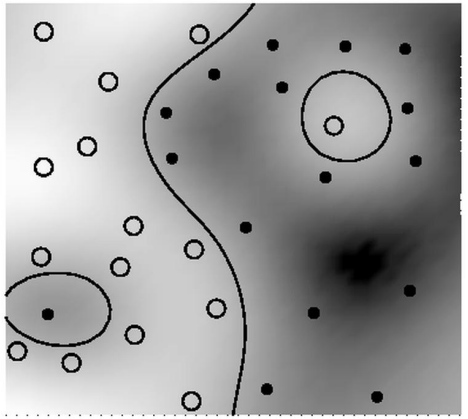
Capacity control



Too in¤exible?



Just right.



Over£tting?

How do we quantify complexity of f ?

Uniform Error Bounds

$$\mathbb{P}\left[\, R[f] - R_{\mathsf{emp}}[f] \,\leq \epsilon \,\middle|\, \forall f \in \mathcal{F} \,\right] \,\geq\, 1 - \delta.$$

$$\mathbb{P}\left[\, \sup_{f \in \mathcal{F}} \left[R[f] - R_{\mathsf{emp}}[f] \,\right] \leq \epsilon \,\right] \,\geq\, 1 - \delta,$$

Not equivalent to:

$$\mathbb{P}\left[\, R[f] - R_{\mathsf{emp}}[f] \,\leq \epsilon \,\right] \,\geq\, 1 - \delta \qquad \forall f \in \mathcal{F}.$$

Vapnik-Chervonenkis type bounds

With probability $1-\delta$

$$\sup_{f\in\mathcal{F}}\left[R[f]-R_{\text{emp}}[f]\,\right]\leq\sqrt{\frac{h\left(\log\left(2m/h\right)+1\right)-\log\left(\delta/4\right)}{m}}$$

where h is the VC dimension of $\mathcal{F}$.

Linear discriminators in $\mathbb{R}^n$: $\quad h=n+1$

. . . with margin γ in ball of radius D: $\quad h=\min\left(n,\lceil D^2/\gamma^2\rceil\right)+1$

Large margin is good!

Covering number bounds

With probability $1-\delta$

$$\sup_{f \in \mathcal{F}} \left[R[f] - R_{\text{emp}}[f] \right] \leq 16M \sqrt{\frac{\log\left(12m\, \mathbb{E}\mathcal{N}_1\left(S, \epsilon/8\right)\right) - \log\delta}{m}}$$

where M is an upper bound on $|\, L(f(x), y)\,|$.

The covering number $\mathcal{N}_1\left(S, \epsilon\right)$ is the number of vectors $v_1, v_2, \ldots, v_n$ needed to ensure that for any $f \in \mathcal{F}$, there is a v_k such that

$$\sum_{i=1}^{m} L(f(x_i), v_k) \leq \epsilon.$$

Stability-based bounds

If f^* is the hypothesis returned by a β-stable algorithm, then with probability $1-\delta$

$$R[f^*] - R_{\text{emp}}[f^*] \leq \beta + \sqrt{\frac{-2\left(m\beta + M\right)^2 \log\left(\delta/2\right)}{m}}$$

where M is an upper bound on $|\, L(f(x), y)\,|$.

An algorithm is β-stable if for all training sets, and any example (x, y) , $L(f^*(x), y)$ changes by at most β when we replace any one of the training examples by any other example.

Rademacher bounds

For $\mathsf{Ra}_r[f] = r$ with probability $1 - \delta$

$$R[f] \leq \inf_{\alpha > 0} \left[(1+\alpha)\, R_{\mathsf{emp}}[f] + \left(1 + \frac{1}{4\alpha}\right) \left(31 r \log^2 \frac{b}{r} + 50 \frac{b\epsilon}{n} \right) \right].$$

Rademacher average:

$$\mathsf{Ra}_r[f] = \mathbb{E}_{S,\sigma} \left[\sup_{f \in \mathcal{F} \,:\, \mathbb{E} L(f(x), y) \leq r} \sum_{i=1}^{m} \sigma_i \, L\left(f(x_i), y_i\right) \right]$$

where $\mathbb{P}[\sigma = 1] = \mathbb{P}[\sigma = -1] = 1/2$.

Structural Risk Minimization

If we have bound of form

$$\mathbb{P}\left[\sup_{f \in \mathcal{F}} [R[f] - R_{\text{emp}}[f]\,] \leq \epsilon_{\mathcal{F}} \right] \geq 1 - \delta$$

1. Fix δ
2. Compute $f^*_{\mathcal{F}} = \arg\min_{f \in \mathcal{F}} [R_{\text{emp}}[f] + \epsilon_{\mathcal{F}}]$ for a sequence of spaces $\mathcal{F}_1 \subset \mathcal{F}_2 \subset \ldots \subset \mathcal{F}_k$
3. Return f^*_i with smallest $R_{\text{emp}}[f^*_i] + \epsilon_{\mathcal{F}_i}$

Does this work?

The problem with error bounds

Most bounds are hopelessly loose. Typically, we get for $1-\delta = .95$

$$\epsilon = 3000.$$

Main culprit is the uniformity requirement. Can we still use them for anything or are they just a weird sport?

Form of bounds is important, even if their value is not. In particular, **large margin** is good.

Hilbert Space Methods

SVM's: the old story

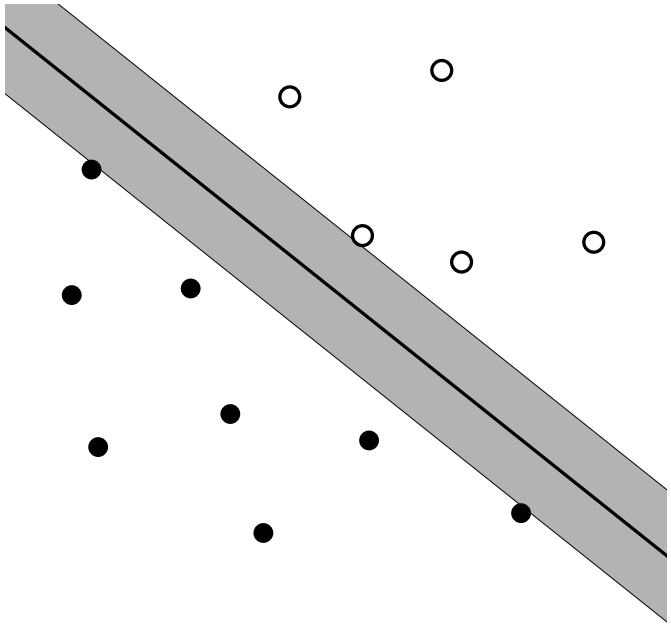
Kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ pos.def. similarity measure

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

e.g. Gaussian Kernel:

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

Find maximum margin separating hyperplane in high dimensional space!



$$f(x) = \operatorname{sgn}\left[b + \sum_{i=1}^{m} \alpha_i \, k(x_i, x) \right]$$

Want more general story behind Hilbert space methods. How do we tell what is a good kernel, anyway? Want large margin. What kernel will give us large margin?

Lessons so far:

- capacity control is crucial;
- large margin is good;
- pursue abstract approach looking for general $f : \mathcal{X} \mapsto \mathcal{Y}$, worry about actual algorithm later.

Regularized Risk

Motivated by form of error bounds, minimize

$$R_{\text{reg}}[f] = \underbrace{\frac{1}{m}\sum_{i=1}^{m} L(f(x_i), y_i)}_{R_{\text{emp}}[f]} + \underbrace{\Omega[f]}_{\text{regularizer}}$$

over some large space of functions $\mathcal{H}$.

$\Omega[f]$ is a penalty term penalizing hypotheses that are too complex. Effectively SRM.

See Regularization networks of Poggio & Girosi.

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

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

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

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

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

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

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

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

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

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 $R_{\text{reg}}[f]$.

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

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

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

The k_x are "prototypical" functions s.t. $\boxed{f(x) = \langle f, k_x \rangle}$.

Representer Theorem

Minimizer of

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

will be in the span of $k_{x_1}, k_{x_2}, \ldots, k_{x_m}$!

The hypothesis can be written

$$f(x) = \langle f, k_x\rangle = \sum_{i=1}^{m} \alpha_i \langle k_{x_i}, k_x\rangle = \sum_{i=1}^{m} \alpha_i\, k(x, x_i).$$

where $k(x, x') = \langle k_x, k_{x'}\rangle$.

All we need to £nd are $\alpha_1, \alpha_2, \ldots, \alpha_m$. How do we construct the RKHS?

Constructing the RKHS $f(x) = \langle f, k_x \rangle$

Bootstrap everything from $k(x, x') = \langle k_x, k_{x'} \rangle$ for $x, x' \in \mathcal{X}$!

1. Anything outside span $\{ k_x \mid x \in \mathcal{X} \}$ is uninteresting, so $f = \int_{\mathcal{X}} \beta(x)\, k_x \, dx.$
2. To evaluate $f(x')$ use

$$f(x) = \langle f, k_{x'} \rangle = \int_{\mathcal{X}} \beta(x) \langle k_x, k_{x'} \rangle \, dx = \int_{\mathcal{X}} \beta(x)\, k(x, x') \, dx$$

3. To compute $\langle f, f' \rangle$ use

$$\langle f, f' \rangle = \int_{\mathcal{X}} \int_{\mathcal{X}} \beta(x)\, \beta(x') \langle k_x, k_{x'} \rangle \, dx dx' = \int_{\mathcal{X}} \int_{\mathcal{X}} \beta(x)\, \beta(x')\, k(x, x') \, dx dx'$$

4. Note that $k_x(x') = \langle k_x, k_{x'} \rangle = k(x, x')$ so we simply have $k_x = k(x, x')$.
5. $\mathcal{H}$ is a particular instance of a feature space $\mathcal{F}$ if we set $\Phi(x) = k_x$.

Correspondence

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

Kernel methods make sense from Regularization Theory point of view if kernel corresponds to sensible operator $\Omega[f] = \langle f, f \rangle_{\mathcal{H}}$.

Fourier regularization

Fourier transform on $\mathbb{R}^n$:

$$\hat{f}(\omega) = \frac{1}{(2\pi)^{n/2}} \int^{\mathbb{R}^n} f(x)\, e^{-i\omega \cdot x} dx$$

Inverse trasform:

$$f(x) = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{C}^n} \hat{f}(x)\, e^{i\omega \cdot x} d\omega$$

Fourier regularization:

$$\Omega[f] = \langle f, f \rangle_{\mathcal{H}} = \int e^{\sigma^2 \| \omega \|^2 / 2} \hat{f}(\omega)^2 \, d\omega$$

Corresponding kernel:

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

The Gaussian kernel will heavily penalize non-smooth functions!

Other kernels

- Homogeneous polynomial: $k(x, x') = (x \cdot x')^p$
- Non-homogeneous polynomial: $k(x, x') = (x \cdot x' + 1)^p$
- tanh kernel: $k(x, x') = \tanh(\kappa\,((\,)\,x \cdot x') + \delta)$
- Triangular kernel: $k(x, x') = 1 - |\,(x - x')\,/d\,|$
- String kernels: $k(\text{string}_1, \text{string}_2)$
- Kernels on distributions: Fisher, etc.
- Diffusion kernels: $k(x, x') = \left[e^{\beta\Delta}\right]_{x,x'}$

Similarity measure $\longleftrightarrow$ Regularization

Algorithms

Modularity of Hilbert space methods

$$f^* = \arg \min_{ftin\mathcal{H}} \left[\frac{1}{m} \sum_{i=1}^{m} \underbrace{L(\langle f, k_{x_i} \rangle, y_i)}_{\text{Determines algorithm}} + \underbrace{\langle f, f \rangle_{\mathcal{H}}}_{\text{Determines kernel}} \right]$$

- Same algorithm (SVM) can be used in very different contexts by changing the kernel $\rightarrow$ kernel engineering
- Regularization scheme can be studied independent of application (classi£cation, regression, etc.)
- ANY kernel method can be formulated as one of these minimization problems

Soft margin SVM's

Relax problem to learning continuous functions $f : \mathcal{X} \mapsto \mathbb{R}$ with hinge loss

$$L(f(x), y) = C \, \max\left(0, -y \, f(x) + 1\right)$$

Then

$$f^* = \arg\min_{f \in \mathcal{H}} \left[\frac{1}{m} \sum_{i=1}^{m} L(\langle f, k_{x_i} \rangle, y_i) + \langle f, f \rangle_{\mathcal{H}} \right]$$

reduces to soft margin SVM

$$f^* = \arg\min_{f \in \mathcal{H}} \left[\langle f, f \rangle + C \sum_{i=1}^{m} \xi_i \right] \qquad \text{subject to} \qquad y_i f(x_i) \geq 1 - \xi_i$$

Probably the most popular algorithm for classi£cation.

Kernel Regression

If we set then

$$f^* = \arg\min_{f \in \mathcal{H}} \left[\frac{1}{m} \sum_{i=1}^{m} L(\langle f, k_{x_i} \rangle, y_i) + \langle f, f \rangle_{\mathcal{H}} \right]$$

reduces to soft kernel regression

$$f^* = \arg\min_{f \in \mathcal{H}} \left[\langle f, f \rangle + C \sum_{i=1}^{m} (\xi_i + \xi_i') \right] \qquad \text{subject to} \qquad \begin{aligned} y_i - f(x_i) &\leq \epsilon + \xi \\ y_i - f(x_i) &\geq -\epsilon - \xi' \end{aligned}$$