SVM et Noyaux

Stéphane Canu
asi.insa-rouen.fr/enseignants/~scanu

JES 2016, Fréjus

October 6, 2016

Plan

1. Kernels and kernel machines

2. Suport vector machines

3. Support Vector Data Description (SVDD)

4. Tuning the kernel: multiple kernel learning (MKL)

The linear least mean square

the linear model

$$y_i = \sum_{j=1}^{p} w_j x_{ij} + \varepsilon_i \quad , \qquad i = 1, n$$

n observations and p variables; $p < n$

$$\min_{\mathbf{w}} = \sum_{i=1}^{n} \left(\sum_{j=1}^{p} x_{ij} w_j - y_i \right)^2 = \|X\mathbf{w} - Y\|^2$$

Solution: $\widehat{\mathbf{w}} = (X^\top X)^{-1} X^\top Y$

$$f(\mathbf{x}) = \mathbf{x}^\top \underbrace{(X^\top X)^{-1} X^\top Y}_{\widehat{\mathbf{w}}}$$

What is the influence of each example (X rows)?

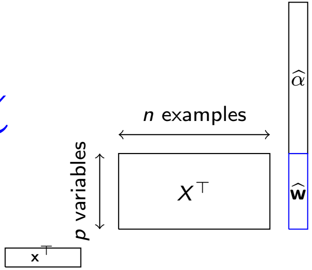
e-Taylor & Cristianini's Book, 2004

The influence of the examples

for a new input $\mathbf{x}$

$$f(\mathbf{x}) = \mathbf{x}^\top (X^\top X)(X^\top X)^{-1} \underbrace{(X^\top X)^{-1} X^\top Y}_{\widehat{\mathbf{w}}}$$

$$= \mathbf{x}^\top X^\top \underbrace{X(X^\top X)^{-1}(X^\top X)^{-1} X^\top Y}_{\widehat{\alpha}}$$



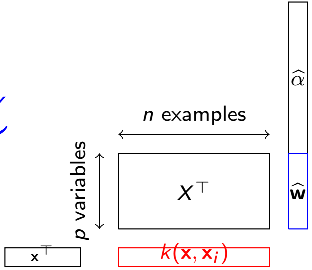
$$f(\mathbf{x}) = \sum_{j=1}^{p} \widehat{\mathbf{w}}_j x_j$$

The influence of the examples

for a new input $\mathbf{x}$

$$f(\mathbf{x}) = \mathbf{x}^\top (X^\top X)(X^\top X)^{-1} \underbrace{(X^\top X)^{-1} X^\top Y}_{\widehat{\mathbf{w}}}$$

$$= \mathbf{x}^\top X^\top \underbrace{X(X^\top X)^{-1}(X^\top X)^{-1} X^\top Y}_{\widehat{\alpha}}$$



$$f(\mathbf{x}) = \sum_{j=1}^{p} \widehat{\mathbf{w}}_j x_j = \sum_{i=1}^{n} \widehat{\alpha}_i \, (\mathbf{x}^\top \mathbf{x}_i)$$

from variables to examples

$$\underbrace{\widehat{\alpha} = X(X^\top X)^{-1} \widehat{\mathbf{w}}}_{n \text{ examples}} \qquad \text{and} \qquad \underbrace{\widehat{\mathbf{w}} = X^\top \widehat{\alpha}}_{p \text{ variables}}$$

What if $p \geq n$?

Introducing non linearities through the feature map

$$f(\mathbf{x}) \quad = \quad \sum_{j=1}^{p} x_j w_j + b \quad = \quad \sum_{i=1}^{n} \alpha_i (\mathbf{x}_i^\top \mathbf{x}) + b$$

$$\begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \in \mathbb{R}^2$$

	x_1
	x_2
	x_3
	x_4
	x_5

linear in $\mathbf{x} \in \mathbb{R}^5$

Introducing non linearities through the feature map

$$f(\mathbf{x}) = \sum_{j=1}^{p} x_j w_j + b = \sum_{i=1}^{n} \alpha_i (\mathbf{x}_i^\top \mathbf{x}) + b$$

$$\begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \in \mathbb{R}^2 \qquad \phi(t) = \begin{array}{|c|c|} \hline t_1 & x_1 \\ t_1^2 & x_2 \\ t_2 & x_3 \\ t_2^2 & x_4 \\ t_1 t_2 & x_5 \\ \hline \end{array}$$

linear in $\mathbf{x} \in \mathbb{R}^5$
quadratic in $\mathbf{t} \in \mathbb{R}^2$

The feature map

$$\begin{array}{rccc} \phi : & \mathbb{R}^2 & \longrightarrow & \mathbb{R}^5 \\ & \mathbf{t} & \longmapsto & \phi(\mathbf{t}) = \mathbf{x} \end{array}$$

$$\mathbf{x}_i^\top \mathbf{x} = \phi(\mathbf{t}_i)^\top \phi(\mathbf{t})$$

Introducing non linearities through the feature map

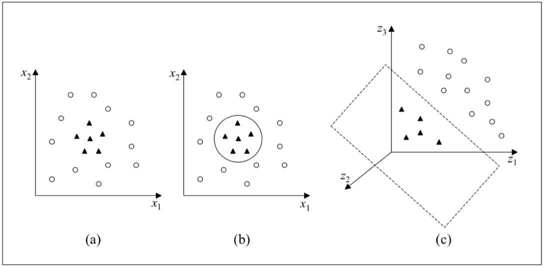


Figura 8. (a) Conjunto de dados não linear; (b) Fronteira não linear no espaço de entradas; (c) Fronteira linear no espaço de características [28]

A. Lorena & A. de Carvalho, Uma Introducão às Support Vector Machines, 2007

Non linear case: dictionnary *vs.* kernel

in the non linear case: use a dictionary of functions

$$\phi_j(\mathbf{x}), j = 1, p \qquad \text{with possibly} \qquad p = \infty$$

for instance polynomials, wavelets...

$$f(\mathbf{x}) = \sum_{j=1}^{p} w_j \phi_j(\mathbf{x}) \qquad \text{with} \qquad w_j = \sum_{i=1}^{n} \alpha_i y_i \phi_j(\mathbf{x}_i)$$

so that

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i \underbrace{\sum_{j=1}^{p} \phi_j(\mathbf{x}_i)\phi_j(\mathbf{x})}_{k(\mathbf{x}_i,\mathbf{x})}$$

Non linear case: dictionnary *vs.* kernel

in the non linear case: use a dictionary of functions

$$\phi_j(\mathbf{x}), j = 1, p \qquad \text{with possibly} \qquad p = \infty$$

for instance polynomials, wavelets...

$$f(\mathbf{x}) = \sum_{j=1}^{p} w_j \phi_j(\mathbf{x}) \qquad \text{with} \qquad w_j = \sum_{i=1}^{n} \alpha_i y_i \phi_j(\mathbf{x}_i)$$

so that

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i \underbrace{\sum_{j=1}^{p} \phi_j(\mathbf{x}_i)\phi_j(\mathbf{x})}_{k(\mathbf{x}_i, \mathbf{x})}$$

$p \geq n$ so what since $k(\mathbf{x}_i, \mathbf{x}) = \sum_{j=1}^{p} \phi_j(\mathbf{x}_i)\phi_j(\mathbf{x})$

closed form kernel: the quadratic kernel

The quadratic dictionary in $\mathbb{R}^d$:

$$\begin{array}{rccl} \mathbf{\Phi}: & \mathbb{R}^d & \to & \mathbb{R}^{p=1+d+\frac{d(d+1)}{2}} \\ & \mathbf{s} & \mapsto & \mathbf{\Phi} = \big(1, s_1, s_2, ..., s_d, s_1^2, s_2^2, ..., s_d^2, ..., s_i s_j, ...\big) \end{array}$$

in this case

$$\mathbf{\Phi}(\mathbf{s})^\top \mathbf{\Phi}(\mathbf{t}) = 1 + s_1 t_1 + s_2 t_2 + ... + s_d t_d + s_1^2 t_1^2 + ... + s_d^2 t_d^2 + ... + s_i s_j t_i t_j + ...$$

closed form kernel: the quadratic kernel

The quadratic dictionary in $\mathbb{R}^d$:

$$\begin{array}{rccl} \mathbf{\Phi}: & \mathbb{R}^d & \to & \mathbb{R}^{p=1+d+\frac{d(d+1)}{2}} \\ & \mathbf{s} & \mapsto & \mathbf{\Phi} = \big(1, s_1, s_2, ..., s_d, s_1^2, s_2^2, ..., s_d^2, ..., s_i s_j, ...\big) \end{array}$$

in this case

$$\mathbf{\Phi}(\mathbf{s})^\top \mathbf{\Phi}(\mathbf{t}) = 1 + s_1 t_1 + s_2 t_2 + ... + s_d t_d + s_1^2 t_1^2 + ... + s_d^2 t_d^2 + ... + s_i s_j t_i t_j + ...$$

The quadratic kenrel: $\mathbf{s}, \mathbf{t} \in \mathbb{R}^d, \quad k(\mathbf{s}, \mathbf{t}) \begin{array}{l} = \big(\mathbf{s}^\top \mathbf{t} + 1\big)^2 \\ = 1 + 2\mathbf{s}^\top \mathbf{t} + \big(\mathbf{s}^\top \mathbf{t}\big)^2 \end{array}$

computes the dot product of the reweighted dictionary:

$$\begin{array}{rccl} \mathbf{\Phi}: & \mathbb{R}^d & \to & \mathbb{R}^{p=1+d+\frac{d(d+1)}{2}} \\ & \mathbf{s} & \mapsto & \mathbf{\Phi} = \big(1, \sqrt{2}s_1, \sqrt{2}s_2, ..., \sqrt{2}s_d, s_1^2, s_2^2, ..., s_d^2, ..., \sqrt{2}s_i s_j, ...\big) \end{array}$$

closed form kernel: the quadratic kernel

The quadratic dictionary in $\mathbb{R}^d$:

$$\begin{array}{rcl} \mathbf{\Phi}: \quad \mathbb{R}^d & \rightarrow & \mathbb{R}^{p=1+d+\frac{d(d+1)}{2}} \\ \mathbf{s} & \mapsto & \mathbf{\Phi} = \big(1, s_1, s_2, ..., s_d, s_1^2, s_2^2, ..., s_d^2, ..., s_i s_j, ...\big) \end{array}$$

in this case

$$\mathbf{\Phi}(\mathbf{s})^\top \mathbf{\Phi}(\mathbf{t}) = 1 + s_1 t_1 + s_2 t_2 + ... + s_d t_d + s_1^2 t_1^2 + ... + s_d^2 t_d^2 + ... + s_i s_j t_i t_j + ...$$

The quadratic kenrel: $\mathbf{s}, \mathbf{t} \in \mathbb{R}^d, \quad k(\mathbf{s}, \mathbf{t}) \begin{array}{l} = \big(\mathbf{s}^\top \mathbf{t} + 1\big)^2 \\ = 1 + 2\mathbf{s}^\top \mathbf{t} + \big(\mathbf{s}^\top \mathbf{t}\big)^2 \end{array}$

computes the dot product of the reweighted dictionary:

$$\begin{array}{rcl} \mathbf{\Phi}: \quad \mathbb{R}^d & \rightarrow & \mathbb{R}^{p=1+d+\frac{d(d+1)}{2}} \\ \mathbf{s} & \mapsto & \mathbf{\Phi} = \big(1, \sqrt{2}s_1, \sqrt{2}s_2, ..., \sqrt{2}s_d, s_1^2, s_2^2, ..., s_d^2, ..., \sqrt{2}s_i s_j, ...\big) \end{array}$$

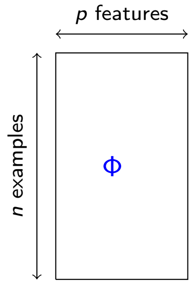
$p = 1 + d + \frac{d(d+1)}{2}$ multiplications *vs.* $d + 1$

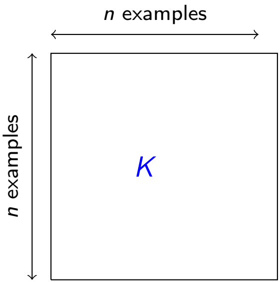
use kernel to save computration

kernel: features throught pairwise comparizons

$\mathbf{x}$ $\phi(\mathbf{x})$

e.g. a text *e.g.* BOW





$$k(\mathbf{x}_i, \mathbf{x}_j) = \sum_{j=1}^{p} \phi_j(\mathbf{x}_i)\phi_j(\mathbf{x}_j)$$

K The matrix of *pairwise* comparizons ($\mathcal{O}(n^2)$)

Kenrel machine

kernel as a dictionary

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$

- α_i influence of example i — depends on y_i
- $k(\mathbf{x}, \mathbf{x}_i)$ the kernel — do NOT depend on y_i

Definition (Kernel)

Let $\mathcal{X}$ be a non empty set (the input space).

A *kernel* is a function k from $\mathcal{X} \times \mathcal{X}$ onto $\mathbb{R}$.

$$\begin{array}{rccc} k: & \mathcal{X} \times \mathcal{X} & \longmapsto & \mathbb{R} \\ & \mathbf{s}, \mathbf{t} & \longrightarrow & k(\mathbf{s}, \mathbf{t}) \end{array}$$

Kenrel machine

kernel as a dictionary

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$

- α_i influence of example i — depends on y_i
- $k(\mathbf{x}, \mathbf{x}_i)$ the kernel — do NOT depend on y_i

Definition (Kernel)

Let $\mathcal{X}$ be a non empty set (the input space).

A *kernel* is a function k from $\mathcal{X} \times \mathcal{X}$ onto $\mathbb{R}$.

$$\begin{array}{rccc} k: & \mathcal{X} \times \mathcal{X} & \longmapsto & \mathbb{R} \\ & \mathbf{s}, \mathbf{t} & \longrightarrow & k(\mathbf{s}, \mathbf{t}) \end{array}$$

semi-parametric version: given the family $q_j(\mathbf{x}),\ j = 1, p$

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i) + \sum_{j=1}^{p} \beta_j q_j(\mathbf{x})$$

Kernel Machine

Definition (Kernel machines)

$$\mathcal{A}\big((\mathbf{x}_i, y_i)_{i=1,n}\big)(\mathbf{x}) = \psi\Big(\sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i) + \sum_{j=1}^{p} \beta_j q_j(\mathbf{x})\Big)$$

α et β: parameters to be estimated.

Exemples

$$\mathcal{A}(x) = \sum_{i=1}^{n} \alpha_i (x - x_i)_+^3 + \beta_0 + \beta_1 x$$ splines

$$\mathcal{A}(\mathbf{x}) = \text{sign}\Big(\sum_{i \in I} \alpha_i \exp^{-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{b}} + \beta_0\Big)$$ SVM

$$\mathbb{P}(y|\mathbf{x}) = \frac{1}{Z} \exp\Big(\sum_{i \in I} \alpha_i \mathbb{1}_{\{y=y_i\}} (\mathbf{x}^\top \mathbf{x}_i + b)^2\Big)$$ exponential family

Plan

1. Kernels and kernel machines
2. Suport vector machines
3. Support Vector Data Description (SVDD)
4. Tuning the kernel: multiple kernel learning (MKL)

In the beginning was the kernel...

Definition (Kernel)

a function of two variable k from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}$

Definition (Positive kernel)

A kernel $k(s,t)$ on $\mathcal{X}$ is said to be positive

- if it is symetric: $k(s,t) = k(t,s)$
- an if for any finite positive interger n:

$$\forall \{\alpha_i\}_{i=1,n} \in \mathbb{R}, \forall \{\mathbf{x}_i\}_{i=1,n} \in \mathcal{X}, \quad \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0$$

it is <u>strictly</u> positive if for $\alpha_i \neq 0$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) > 0$$

Examples of positive kernels

the linear kernel: $\mathbf{s},\mathbf{t}\in\mathbb{R}^d, \quad k(\mathbf{s},\mathbf{t}) = \mathbf{s}^\top\mathbf{t}$

symetric: $\mathbf{s}^\top\mathbf{t} = \mathbf{t}^\top\mathbf{s}$

positive:
$$\sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j) = \sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j \mathbf{x}_i^\top\mathbf{x}_j = \left(\sum_{i=1}^n \alpha_i\mathbf{x}_i\right)^\top \left(\sum_{j=1}^n \alpha_j\mathbf{x}_j\right) = \left\|\sum_{i=1}^n \alpha_i\mathbf{x}_i\right\|^2$$

the product kernel: $k(\mathbf{s},\mathbf{t}) = g(\mathbf{s})g(\mathbf{t})$ for some $g : \mathbb{R}^d \to \mathbb{R}$,

symetric by construction

positive:
$$\sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j) = \sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j g(\mathbf{x}_i)g(\mathbf{x}_j) = \left(\sum_{i=1}^n \alpha_i g(\mathbf{x}_i)\right)\left(\sum_{j=1}^n \alpha_j g(\mathbf{x}_j)\right) = \left(\sum_{i=1}^n \alpha_i g(\mathbf{x}_i)\right)^2$$

k is positive $\Leftrightarrow$ (its square root exists) $\Leftrightarrow$ $k(\mathbf{s},\mathbf{t}) = \langle\phi_\mathbf{s},\phi_\mathbf{t}\rangle$

Positive definite Kernel (PDK) algebra (closure)

if $k_1(\mathbf{s},\mathbf{t})$ and $k_2(\mathbf{s},\mathbf{t})$ are two positive kernels

- DPK are a convex cone: $\forall a_1 \in \mathbb{R}^+ \quad a_1 k_1(\mathbf{s},\mathbf{t}) + k_2(\mathbf{s},\mathbf{t})$
- product kernel $k_1(\mathbf{s},\mathbf{t}) k_2(\mathbf{s},\mathbf{t})$

proofs

- by linearity:

$$\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \big(a_1 k_1(i,j) + k_2(i,j)\big) = a_1 \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j k_1(i,j) + \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j k_2(i,j)$$

- assuming $\exists \psi_\ell$ s.t. $k_1(\mathbf{s},\mathbf{t}) = \sum_\ell \psi_\ell(\mathbf{s}) \psi_\ell(\mathbf{t})$

$$\begin{aligned}\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \; k_1(\mathbf{x}_i,\mathbf{x}_j) k_2(\mathbf{x}_i,\mathbf{x}_j) &= \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \Big(\sum_\ell \psi_\ell(\mathbf{x}_i) \psi_\ell(\mathbf{x}_j) k_2(\mathbf{x}_i,\mathbf{x}_j)\Big) \\ &= \sum_\ell \sum_{i=1}^n \sum_{j=1}^n \big(\alpha_i \psi_\ell(\mathbf{x}_i)\big) \; \big(\alpha_j \psi_\ell(\mathbf{x}_j)\big) \; k_2(\mathbf{x}_i,\mathbf{x}_j)\end{aligned}$$

N. Cristianini and J. Shawe Taylor, kernel methods for pattern analysis, 2004

Kernel engineering: building PDK

- for any polynomial with positive coef. ϕ from $\mathbb{R}$ to $\mathbb{R}$

$$\phi\big(k(\mathbf{s},\mathbf{t})\big)$$

- if $\Psi$is a function from $\mathbb{R}^d$ to $\mathbb{R}^d$

$$k\big(\Psi(\mathbf{s}),\Psi(\mathbf{t})\big)$$

- if φ from $\mathbb{R}^d$ to $\mathbb{R}^+$, is minimum in 0

$$k(\mathbf{s},\mathbf{t}) = \varphi(\mathbf{s}+\mathbf{t}) - \varphi(\mathbf{s}-\mathbf{t})$$

- convolution of two positive kernels is a positive kernel

$$K_1 \star K_2$$

Example : the Gaussian kernel is a PDK

$$\begin{aligned} \exp(-\|\mathbf{s}-\mathbf{t}\|^2) &= \exp(-\|\mathbf{s}\|^2 - \|\mathbf{t}\|^2 + 2\mathbf{s}^\top\mathbf{t}) \\ &= \exp(-\|\mathbf{s}\|^2)\exp(-\|\mathbf{t}\|^2)\exp(2\mathbf{s}^\top\mathbf{t}) \end{aligned}$$

- $\mathbf{s}^\top\mathbf{t}$ is a PDK and function exp as the limit of positive series expansion, so $\exp(2\mathbf{s}^\top\mathbf{t})$ is a PDK
- $\exp(-\|\mathbf{s}\|^2)\exp(-\|\mathbf{t}\|^2)$ is a PDK as a product kernel
- the product of two PDK is a PDK

O. Catoni, master lecture, 2005

some examples of PD kernels...

type	name	$k(s,t)$
radial	gaussian	$\exp\left(-\frac{r^2}{b}\right), \quad r = \|s-t\|$
radial	laplacian	$\exp(-r/b)$
radial	rationnal	$1 - \frac{r^2}{r^2+b}$
radial	loc. gauss.	$\max\left(0, 1-\frac{r}{3b}\right)^d \exp(-\frac{r^2}{b})$
non stat.	χ^2	$\exp(-r/b), \; r = \sum_k \frac{(s_k-t_k)^2}{s_k+t_k}$
projective	polynomial	$(s^\top t)^p$
projective	affine	$(s^\top t + b)^p$
projective	cosine	$s^\top t/\|s\|\|t\|$
projective	correlation	$\exp\left(\frac{s^\top t}{\|s\|\|t\|} - b\right)$

Most of the kernels depends on a quantity b called the bandwidth

kernels for objects and structures

kernels on histograms and probability distributions
kernel on strings

- spectral string kernel $k(\mathbf{s},\mathbf{t}) = \sum_u \phi_u(\mathbf{s})\phi_u(\mathbf{t})$
- using sub sequences
- similarities by alignements $k(\mathbf{s},\mathbf{t}) = \sum_\pi \exp(\beta(\mathbf{s},\mathbf{t},\pi))$

kernels on graphs

- the pseudo inverse of the (regularized) graph Laplacian

$L = D - A$ $\quad$ A is the adjency matrix$D$ the degree matrix

- diffusion kernels $\frac{1}{Z(b)} \exp^{bL}$
- subgraph kernel convolution (using random walks)

and kernels on HMM, automata, dynamical system...

Shawe-Taylor & Cristianini's Book, 2004 ; JP Vert, 2006

Multiple kernel

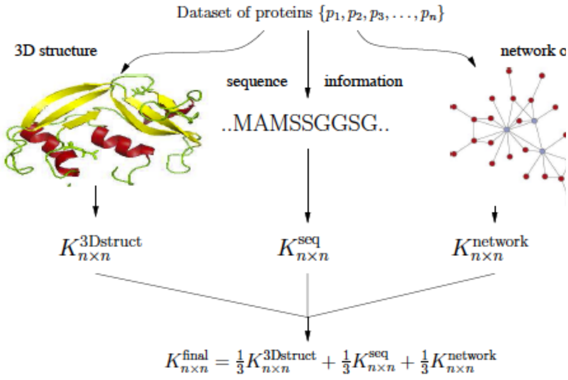
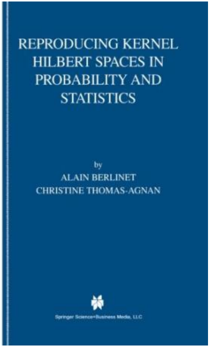


Figure 2: A dataset of proteins can be regarded in (at least) three different wa
of 3D structures, a dataset of sequences and a set of nodes in a network which in
other. A different kernel matrix can be extracted from each datatype, using know
shapes, strings and graphs. The resulting kernels can then be combined togethe
weights, as is the case above where a simple average is considered, or estimated
the subject of Section 5.2

Let's summarize

- positive kernels
- there is a lot of them
- can be rather complex
- the bandwith matters (more than the kernel itself)
- extentions to non positive kernels



Road map

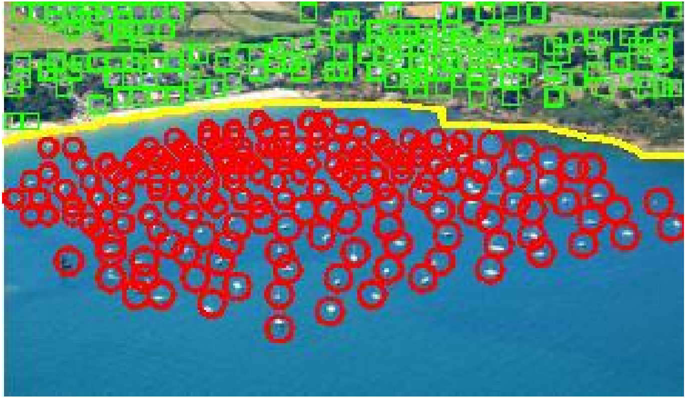
1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

Supervised classification as Learning from examples

The task, use longitude and latitude to predict: is it a boat or a house?

credit: A Gentle Introduction to Support Vector Machines in Biomedicine A. Statnikov, D. Hardin, I. Guyon and C. F. Aliferis
www.nyuinformatics.org/downloads/supplements/SVM_Tutorial_2010/Final_WB.pdf

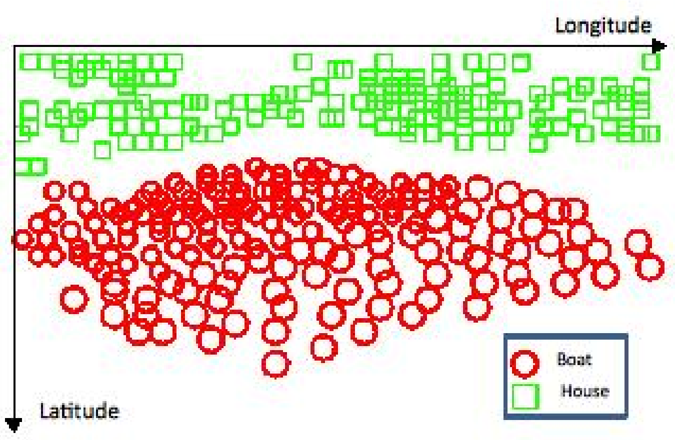
Supervised classification as Learning from examples



Using (red and green) labelled examples learn a (yellow) decision rule

credit: **A Gentle Introduction to Support Vector Machines in Biomedicine A. Statnikov, D. Hardin, I. Guyon and C. F. Aliferis**
www.nyuinformatics.org/downloads/supplements/SVM_Tutorial_2010/Final_WB.pdf

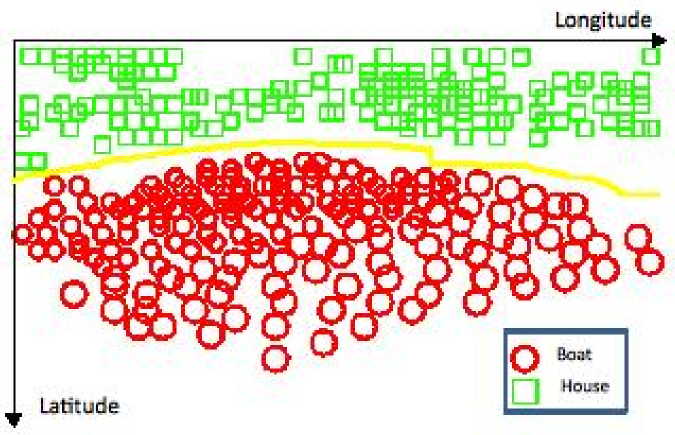
Supervised classification as Learning from examples



Using (red and green) labelled examples...

credit: **A Gentle Introduction to Support Vector Machines in Biomedicine A. Statnikov, D. Hardin, I. Guyon and C. F. Aliferis**
www.nyuinformatics.org/downloads/supplements/SVM_Tutorial_2010/Final_WB.pdf

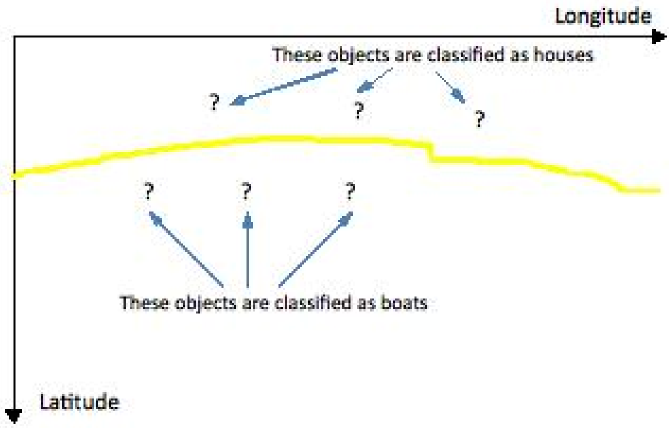
Supervised classification as Learning from examples



Using (red and green) labelled examples... learn a (yellow) decision rule

credit: A Gentle Introduction to Support Vector Machines in Biomedicine A. Statnikov, D. Hardin, I. Guyon and C. F. Aliferis
www.nyuinformatics.org/downloads/supplements/SVM_Tutorial_2010/Final_WB.pdf

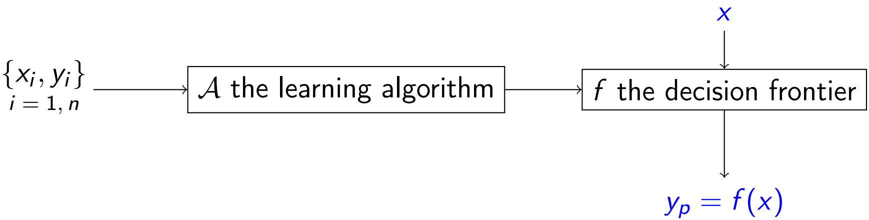
Supervised classification as Learning from examples



Use the decision border to predict unseen objects label

credit: A Gentle Introduction to Support Vector Machines in Biomedicine A. Statnikov, D. Hardin, I. Guyon and C. F. Aliferis
www.nyuinformatics.org/downloads/supplements/SVM_Tutorial_2010/Final_WB.pdf

Suppervised classification: the 2 steps



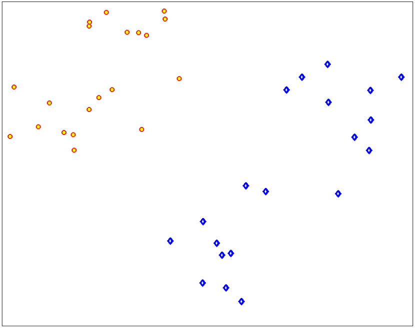
1. the border $\leftarrow$ *Learn*(xi, yi, n training data)   `%` $\mathcal{A}$ `is SVM_learn`
2. $y_p \leftarrow$ *Predict*(unseen x, the border)   `%` f `is SVM_val`

Road map

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

Separating hyperplanes

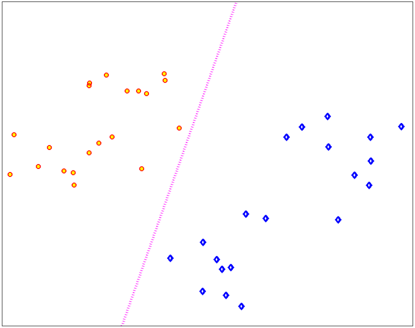
Find a line to separate (classify) blue from red



$$D(x) = \mathrm{sign}\left(\mathbf{v}^\top \mathbf{x} + a\right)$$

Separating hyperplanes

Find a line to separate (classify) blue from red



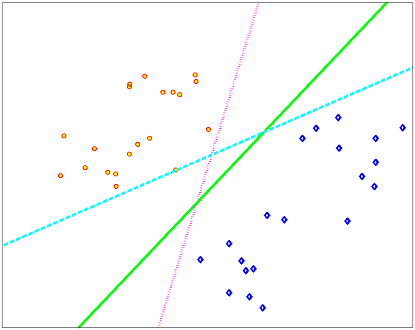
$$D(x) = \text{sign}\left(\mathbf{v}^\top \mathbf{x} + a\right)$$

the decision border:

$$\mathbf{v}^\top \mathbf{x} + a = 0$$

Separating hyperplanes

Find a line to separate (classify) blue from red



$$D(x) = \text{sign}\left(\mathbf{v}^\top \mathbf{x} + a\right)$$

the decision border:

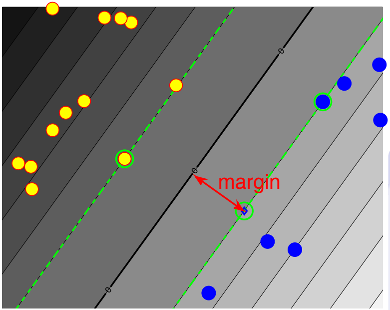
$$\mathbf{v}^\top \mathbf{x} + a = 0$$

there are many solutions...
The problem is ill posed

How to choose a solution?

Maximize our *confidence* = maximize the margin

the decision border: $\Delta(\mathbf{v}, a) = \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{v}^\top \mathbf{x} + a = 0\}$



maximize the margin

$$\max_{\mathbf{v},a} \underbrace{\min_{i \in [1,n]} \text{dist}(\mathbf{x}_i, \Delta(\mathbf{v}, a))}_{\text{margin: } m}$$

Maximize the confidence

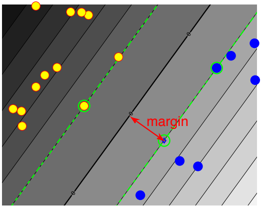
$$\begin{cases} \max\limits_{\mathbf{v},a} & m \\ \text{with} & \min\limits_{i=1,n} \dfrac{|\mathbf{v}^\top \mathbf{x}_i + a|}{\|\mathbf{v}\|} \geq m \end{cases}$$

the problem is still ill posed

if $(\mathbf{v}, a)$ is a solution, $\forall\ 0 < k$ $(k\mathbf{v}, ka)$ is also a solution...

Linear SVM: the problem

The maximal margin (=minimal norm)
canonical hyperplane



Linear SVMs are the solution of the following problem (called primal)

Let $\{(\mathbf{x}_i, y_i);\ i = 1:n\}$ be a set of labelled data with $\mathbf{x} \in \mathbb{R}^d, y_i \in \{1, -1\}$

A support vector machine (SVM) is a linear classifier associated with the following decision function: $D(x) = \text{sign}\big(\mathbf{w}^\top \mathbf{x} + b\big)$ where $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ a given thought the solution of the following problem:

$$\begin{cases} \min\limits_{\mathbf{w}\in\mathbb{R}^d,\ b\in\mathbb{R}} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1\ , \qquad i = 1, n \end{cases}$$

This is a quadratic program (QP): $\begin{cases} \min\limits_{\mathbf{z}} & \frac{1}{2}\mathbf{z}^\top A\mathbf{z} - \mathbf{d}^\top \mathbf{z} \\ \text{with} & B\mathbf{z} \leq \mathbf{e} \end{cases}$

Support vector machines as a QP

The Standart QP formulation

$$\begin{cases} \min_{\mathbf{w},b} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, i = 1, n \end{cases} \Leftrightarrow \begin{cases} \min_{\mathbf{z}\in\mathbb{R}^{d+1}} & \frac{1}{2}\mathbf{z}^\top A\mathbf{z} - \mathbf{d}^\top \mathbf{z} \\ \text{with} & B\mathbf{z} \leq \mathbf{e} \end{cases}$$

$\mathbf{z} = (\mathbf{w}, b)^\top$, $\mathbf{d} = (0, \ldots, 0)^\top$, $A = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$, $B = -[\text{diag}(\mathbf{y})X, \mathbf{y}]$ and $\mathbf{e} = -(1, \ldots, 1)^\top$

Solve it using a standard QP solver such as (for instance)

```
% QUADPROG Quadratic programming.
%    X = QUADPROG(H,f,A,b) attempts to solve the quadratic programming problem:
%
%             min 0.5*x'*H*x + f'*x   subject to:  A*x <= b
%              x
%  so that the solution is in the range LB <= X <= UB
```

For more solvers (just to name a few) have a look at:

- plato.asu.edu/sub/nlores.html#QP-problem
- www.numerical.rl.ac.uk/people/nimg/qp/qp.html

Linear SVM dual formulation - The lagrangian

$$\begin{cases} \min\limits_{\mathbf{w},b} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \qquad i = 1, n \end{cases}$$

Looking for the lagrangian saddle point $\max\limits_{\alpha} \min\limits_{\mathbf{w},b} \mathcal{L}(\mathbf{w}, b, \alpha)$ with so called lagrange multipliers $\alpha_i \geq 0$

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \big(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 \big)$$

α_i represents the influence of constraint thus the influence of the training example (x_i, y_i)

KKT conditions for SVM

$$\begin{cases} \min_{\mathbf{w},b} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \qquad i = 1, n \end{cases}$$

stationarity	$\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0$ and $\sum_{i=1}^{n} \alpha_i \; y_i = 0$	
primal admissibility	$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1$	$i = 1, \dots, n$
dual admissibility	$\alpha_i \geq 0$	$i = 1, \dots, n$
complementarity	$\alpha_i \Big(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 \Big) = 0$	$i = 1, \dots, n$

The complementary condition split the data into two sets

- $\mathcal{A}$ be the set of active constraints: usefull points

$$\mathcal{A} = \{ i \in [1, n] \;\big|\; y_i(\mathbf{w}^{*\top} \mathbf{x}_i + b^*) = 1 \}$$

- its complementary $\bar{\mathcal{A}}$ useless points

$$\text{if } i \notin \mathcal{A}, \alpha_i = 0$$

Linear SVM dual formulation

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \big(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1\big)$$

Optimality: $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \qquad \sum_{i=1}^{n} \alpha_i \; y_i = 0$

$$\begin{aligned}
\mathcal{L}(\alpha) &= \underbrace{\tfrac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_j \alpha_i y_i y_j \mathbf{x}_j^\top \mathbf{x}_i}_{\mathbf{w}^\top \mathbf{w}} - \textstyle\sum_{i=1}^{n} \alpha_i y_i \underbrace{\sum_{j=1}^{n} \alpha_j y_j \mathbf{x}_j^\top}_{\mathbf{w}^\top} \mathbf{x}_i - b \underbrace{\sum_{i=1}^{n} \alpha_i y_i}_{=0} + \textstyle\sum_{i=1}^{n} \alpha_i \\
&= -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_j \alpha_i y_i y_j \mathbf{x}_j^\top \mathbf{x}_i + \sum_{i=1}^{n} \alpha_i
\end{aligned}$$

Dual linear SVM is also a quadratic program

$$\text{problem } \mathcal{D} \quad \begin{cases} \min\limits_{\alpha \in \mathbb{R}^n} & \frac{1}{2}\alpha^\top G \alpha - \mathbf{e}^\top \alpha \\ \text{with} & \mathbf{y}^\top \alpha = 0 \\ \text{and} & 0 \le \alpha_i \qquad\qquad i = 1, n \end{cases}$$

with G a symmetric matrix $n \times n$ such that $G_{ij} = y_i y_j \mathbf{x}_j^\top \mathbf{x}_i$

SVM primal vs. dual

Primal

$$\begin{cases} \min\limits_{\mathbf{w}\in\mathbb{R}^d, b\in\mathbb{R}} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \\ & i = 1, n \end{cases}$$

- $d+1$ unknown
- n constraints
- classical QP
- perfect when $d << n$

Dual

$$\begin{cases} \min\limits_{\alpha\in\mathbb{R}^n} & \frac{1}{2}\alpha^\top G\alpha - \mathbf{e}^\top\alpha \\ \text{with} & \mathbf{y}^\top\alpha = 0 \\ \text{and} & 0 \leq \alpha_i \qquad i = 1, n \end{cases}$$

- n unknown
- G Gram matrix (pairwise influence matrix)
- n box constraints
- easy to solve
- to be used when $d > n$

SVM primal vs. dual

Primal

$$\begin{cases} \min_{\mathbf{w}\in\mathbb{R}^d, b\in\mathbb{R}} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \\ & i = 1, n \end{cases}$$

- $d + 1$ unknown
- n constraints
- classical QP
- perfect when $d << n$

Dual

$$\begin{cases} \min_{\alpha\in\mathbb{R}^n} & \frac{1}{2}\alpha^\top G\alpha - \mathbf{e}^\top \alpha \\ \text{with} & \mathbf{y}^\top \alpha = 0 \\ \text{and} & 0 \leq \alpha_i \qquad i = 1, n \end{cases}$$

- n unknown
- G Gram matrix (pairwise influence matrix)
- n box constraints
- easy to solve
- to be used when $d > n$

$$f(\mathbf{x}) = \sum_{j=1}^{d} w_j x_j + b = \sum_{i=1}^{n} \alpha_i \, y_i(\mathbf{x}^\top \mathbf{x}_i) + b$$

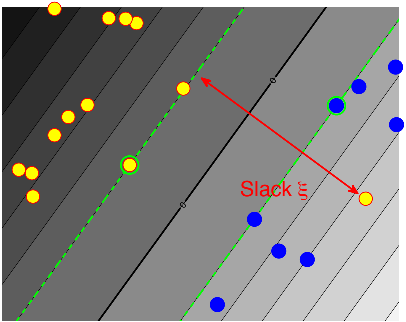
Road map

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

The non separable case: a bi criteria optimization problem

Modeling potential errors: introducing slack variables ξ_i

$$(x_i, y_i) \qquad \begin{cases} \text{no error:} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \Rightarrow \quad \xi_i = 0 \\ \text{error:} & \qquad \xi_i = 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b) > 0 \end{cases}$$



$$\begin{cases} \min\limits_{\mathbf{w},b,\xi} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \min\limits_{\mathbf{w},b,\xi} & \frac{C}{p}\sum_{i=1}^{n} \xi_i^p \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \quad i = 1, n \end{cases}$$

Our hope: almost all $\xi_i = 0$

The non separable case

Modeling potential errors: introducing slack variables ξ_i

$$(x_i, y_i) \qquad \begin{cases} \text{no error:} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \Rightarrow & \xi_i = 0 \\ \text{error:} & & \xi_i = 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b) > 0 \end{cases}$$

Minimizing also the slack (the error), for a given $C > 0$

$$\begin{cases} \min\limits_{\mathbf{w},b,\xi} & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{p}\sum_{i=1}^{n} \xi_i^p \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad & i = 1, n \\ & \xi_i \geq 0 & i = 1, n \end{cases}$$

Looking for the saddle point of the lagrangian with the Lagrange multipliers $\alpha_i \geq 0$ and $\beta_i \geq 0$

$$\mathcal{L}(\mathbf{w}, b, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{p}\sum_{i=1}^{n} \xi_i^p - \sum_{i=1}^{n} \alpha_i \big(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i\big) - \sum_{i=1}^{n} \beta_i \xi_i$$

The KKT

$$\mathcal{L}(\mathbf{w}, b, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{p}\sum_{i=1}^{n}\xi_i^p - \sum_{i=1}^{n}\alpha_i\big(y_i(\mathbf{w}^\top\mathbf{x}_i + b) - 1 + \xi_i\big) - \sum_{i=1}^{n}\beta_i\xi_i$$

stationarity	$\mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i = 0$ and $\sum_{i=1}^{n}\alpha_i\ y_i = 0$	
	$C - \alpha_i - \beta_i = 0$	$i = 1, \dots, n$
primal admissibility	$y_i(\mathbf{w}^\top\mathbf{x}_i + b) \geq 1$	$i = 1, \dots, n$
	$\xi_i \geq 0$	$i = 1, \dots, n$
dual admissibility	$\alpha_i \geq 0$	$i = 1, \dots, n$
	$\beta_i \geq 0$	$i = 1, \dots, n$
complementarity	$\alpha_i\Big(y_i(\mathbf{w}^\top\mathbf{x}_i + b) - 1 + \xi_i\Big) = 0$	$i = 1, \dots, n$
	$\beta_i\xi_i = 0$	$i = 1, \dots, n$

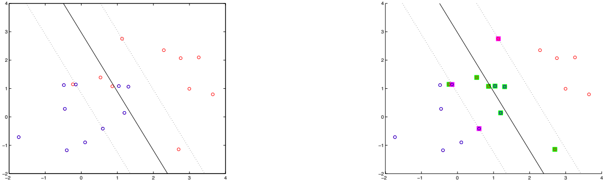
Let's eliminate β!

KKT

stationarity	$\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0$	and	$\sum_{i=1}^{n} \alpha_i \; y_i = 0$
primal admissibility	$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1$		$i = 1, \dots, n$
	$\xi_i \geq 0$		$i = 1, \dots, n;$
dual admissibility	$\alpha_i \geq 0$		$i = 1, \dots, n$
	$C - \alpha_i \geq 0$		$i = 1, \dots, n;$
complementarity	$\alpha_i \Big(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i \Big) = 0$		$i = 1, \dots, n$
	$(C - \alpha_i) \; \xi_i = 0$		$i = 1, \dots, n$

sets	I_0	$I_{\mathcal{A}}$	I_C
α_i	0	$0 < \alpha < C$	C
β_i	C	$C - \alpha$	0
ξ_i	0	0	$1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b)$
	$y_i(\mathbf{w}^\top \mathbf{x}_i + b) > 1$	$y_i(\mathbf{w}^\top \mathbf{x}_i + b) = 1$	$y_i(\mathbf{w}^\top \mathbf{x}_i + b) < 1$
	useless	usefull (support vec)	suspicious

The importance of being support



data point	α	constraint value	set
$\mathbf{x}_i$ *useless*	$\alpha_i = 0$	$y_i\big(\mathbf{w}^\top \mathbf{x}_i + b\big) > 1$	I_0
$\mathbf{x}_i$ *support*	$0 < \alpha_i < C$	$y_i\big(\mathbf{w}^\top \mathbf{x}_i + b\big) = 1$	I_α
$\mathbf{x}_i$ *suspicious*	$\alpha_i = C$	$y_i\big(\mathbf{w}^\top \mathbf{x}_i + b\big) < 1$	I_C

Table : When a data point is « support » it lies exactly on the margin.

here lies the efficiency of the algorithm (and its complexity)!

sparsity: $\alpha_i = 0$

Optimality conditions ($p = 1$)

$$\mathcal{L}(\mathbf{w}, b, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\big(y_i(\mathbf{w}^\top\mathbf{x}_i + b) - 1 + \xi_i\big) - \sum_{i=1}^{n}\beta_i\xi_i$$

Computing the gradients:

$$\begin{cases} \nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \alpha) & = \mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i \\ \dfrac{\partial \mathcal{L}(\mathbf{w}, b, \alpha)}{\partial b} & = \sum_{i=1}^{n}\alpha_i\ y_i \\ \nabla_{\xi_i}\mathcal{L}(\mathbf{w}, b, \alpha) & = C - \alpha_i - \beta_i \end{cases}$$

- no change for $\mathbf{w}$ and b
- $\beta_i \geq 0$ and $C - \alpha_i - \beta_i = 0 \quad \Rightarrow \quad \alpha_i \leq C$

The dual formulation:

$$\begin{cases} \min\limits_{\alpha \in \mathbb{R}^n} & \frac{1}{2}\alpha^\top G\alpha - \mathbf{e}^\top\alpha \\ \text{with} & \mathbf{y}^\top\alpha = 0 \\ \text{and} & 0 \leq \alpha_i \leq C \qquad i = 1, n \end{cases}$$

SVM primal vs. dual

Primal

$$\begin{cases} \min\limits_{\mathbf{w},b,\xi\in\mathbb{R}^n} & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{with} & y_i(\mathbf{w}^\top\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \quad i = 1, n \end{cases}$$

- $d + n + 1$ unknown
- $2n$ constraints
- classical QP
- to be used when n is too large to build G

Dual

$$\begin{cases} \min\limits_{\alpha\in\mathbb{R}^n} & \frac{1}{2}\alpha^\top G\alpha - \mathbf{e}^\top\alpha \\ \text{with} & \mathbf{y}^\top\alpha = 0 \\ \text{and} & 0 \leq \alpha_i \leq C \qquad i = 1, n \end{cases}$$

- n unknown
- G Gram matrix (pairwise influence matrix)
- $2n$ box constraints
- easy to solve
- to be used when n is not too large

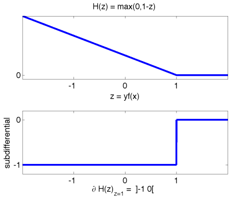
Eliminating the slack but not the possible mistakes

$$\begin{cases} \min_{\mathbf{w},b,\xi\in\mathbb{R}^n} & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{with} & y_i(\mathbf{w}^\top\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \quad i = 1, n \end{cases}$$

Introducing the hinge loss

$$\xi_i = \max\big(1 - y_i(\mathbf{w}^\top\mathbf{x}_i + b), 0\big)$$

$$\min_{\mathbf{w},b} \frac{1}{2} \;\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\max\big(0, 1 - y_i(\mathbf{w}^\top\mathbf{x}_i + b)\big)$$



Back to $d + 1$ variables, but this is no longer an explicit QP

The hinge and other loss

Square hinge: (huber/hinge) and Lasso SVM

$$\min_{\mathbf{w},b} \quad \|\mathbf{w}\|_1 + \; C \sum_{i=1}^{n} \max\left(1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b), 0\right)^p$$

Penalized Logistic regression (Maxent)

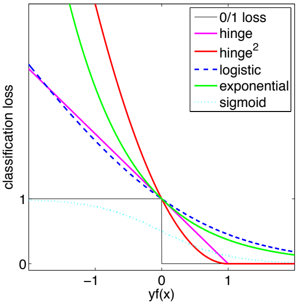
$$\min_{\mathbf{w},b} \quad \|\mathbf{w}\|_2^2 - \; C \sum_{i=1}^{n} \log\left(1 + \exp^{-2y_i(\mathbf{w}^\top \mathbf{x}_i + b)}\right)$$

The exponential loss (commonly used in boosting)

$$\min_{\mathbf{w},b} \quad \|\mathbf{w}\|_2^2 + \; C \sum_{i=1}^{n} \exp^{-y_i(\mathbf{w}^\top \mathbf{x}_i + b)}$$

The sigmoid loss

$$\min_{\mathbf{w},b} \quad \|\mathbf{w}\|_2^2 - \; C \sum_{i=1}^{n} \tanh\left(y_i(\mathbf{w}^\top \mathbf{x}_i + b)\right)$$

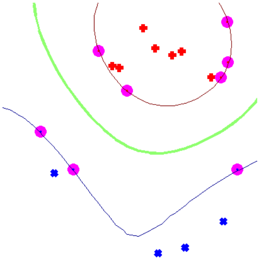


Roadmap

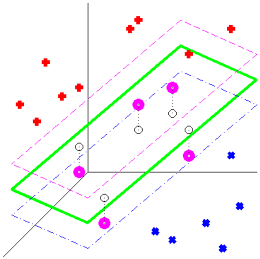
1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

using relevant features...

a data point becomes a function $\mathbf{x} \longrightarrow k(\mathbf{x}, \bullet)$



input space representation: x



feature space: k(x,.)

Representer theorem for SVM

$$\begin{cases} \min_{f,b} & \frac{1}{2}\|f\|^2_{\mathcal{H}} \\ \text{with} & y_i\big(f(\mathbf{x}_i)+b\big) \geq 1 \end{cases}$$

Lagrangian

$$L(f,b,\alpha) = \frac{1}{2}\|f\|^2_{\mathcal{H}} - \sum_{i=1}^{n} \alpha_i \big(y_i(f(\mathbf{x}_i)+b) - 1\big) \qquad \alpha \geq 0$$

optimility condition: $\nabla_f L(f,b,\alpha) = 0 \Leftrightarrow f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x})$

Eliminate f from L: $\begin{cases} \|f\|^2_{\mathcal{H}} = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j) \\ \sum_{i=1}^{n} \alpha_i y_i f(\mathbf{x}_i) = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j) \end{cases}$

$$Q(b,\alpha) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j) - \sum_{i=1}^{n} \alpha_i\big(y_i b - 1\big)$$

the general case: C-SVM

Primal formulation

$$(\mathcal{P}) \begin{cases} \min_{f\in\mathcal{H}, b, \xi\in\mathbb{R}^n} & \frac{1}{2}\|f\|^2 + \frac{C}{p}\sum_{i=1}^{n}\xi_i^p \\ \text{such that} & y_i\big(f(\mathbf{x}_i)+b\big) \geq 1-\xi_i, \;\; \xi_i \geq 0, \;\; i=1,n \end{cases}$$

C is the *regularization path* parameter (to be tuned)

$p = 1$, L_1 SVM

$$\begin{cases} \max_{\alpha\in\mathbb{R}^n} & -\frac{1}{2}\alpha^\top G\alpha + \alpha^\top \mathbb{1} \\ \text{such that} & \alpha^\top \mathbf{y} = 0 \text{ and } 0 \leq \alpha_i \leq C \quad i=1,n \end{cases}$$

$p = 2$, L_2 SVM

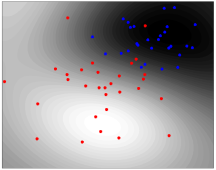
$$\begin{cases} \max_{\alpha\in\mathbb{R}^n} & -\frac{1}{2}\alpha^\top\left(G + \frac{1}{C}I\right)\alpha + \alpha^\top \mathbb{1} \\ \text{such that} & \alpha^\top \mathbf{y} = 0 \text{ and } 0 \leq \alpha_i \quad i=1,n \end{cases}$$

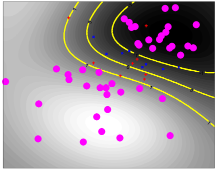
the regularization path: is the set of solutions $\alpha(C)$ when C varies

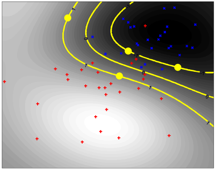
Data groups: illustration

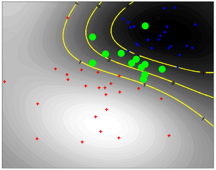
$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$

$$D(x) = \text{sign}\big(f(\mathbf{x}) + b\big)$$







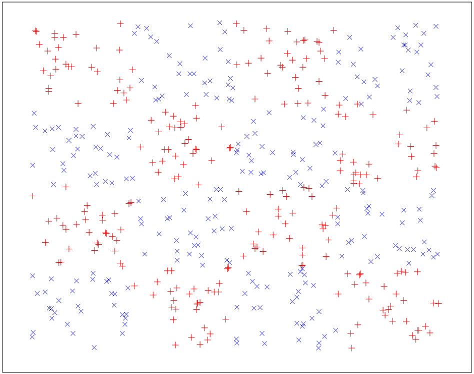


useless data	important data	suspicious data
well classified	support	
$\alpha = 0$	$0 < \alpha < C$	$\alpha = C$

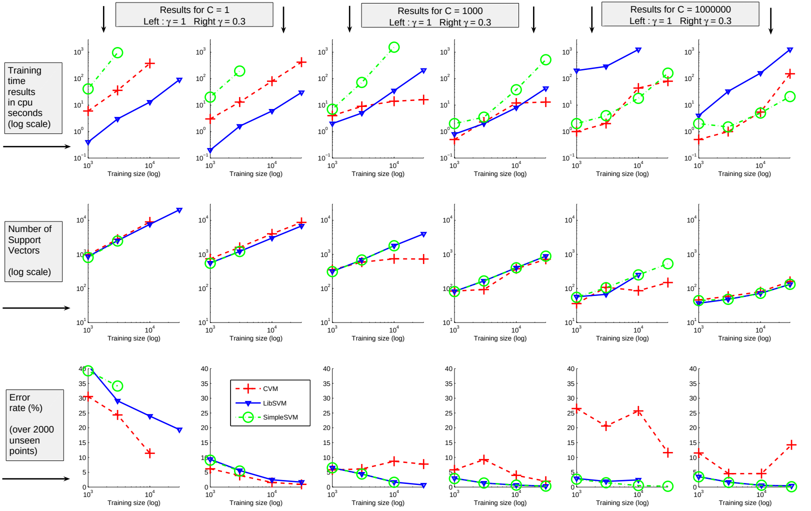
the regularization path: is the set of solutions $\alpha(C)$ when C varies

checker board

- 2 classes
- 500 examples
- separable



Empirical complexity



G. Loosli *et al* JMLR, 2007

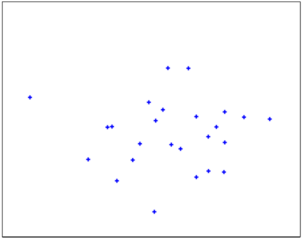
Conclusion

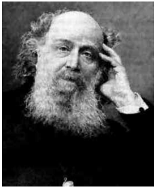
- Learning as an optimization problem
  - use CVX to prototype
  - MonQP
  - specific parallel and distributed solvers
- Universal through Kernelization (dual trick)
- Scalability
  - Sparsity provides scalability
  - Kernel implies "locality"
  - Big data limitations: back to primal (an linear)

Plan

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

The minimum enclosing ball problem

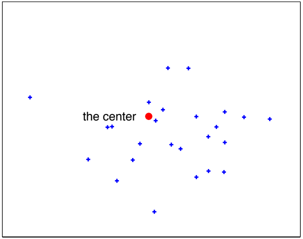




Original formulation

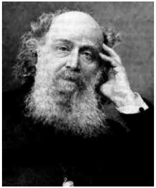
It is required to find the least circle which shall contain a given system of points in a plane

The minimum enclosing ball problem

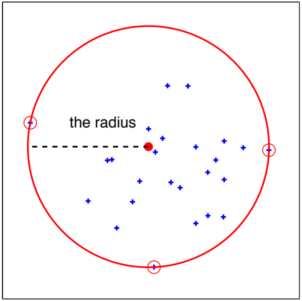


Original formulation

It is required to find the least circle which shall contain a given system of points in a plane



The minimum enclosing ball problem

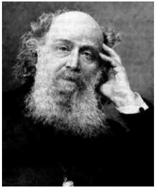


Given n points, $\{\mathbf{x}_i, i = 1, n\}$.

$$\begin{cases} \min\limits_{R \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^d} & R^2 \\ \text{with} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2, \quad i = 1, \dots, n \end{cases}$$

Original formulation

It is required to find the least circle which shall contain a given system of points in a plane



MEB as a QP in the primal [?]

Theorem (MEB as a QP)

The two following problems are equivalent,

$$\begin{cases} \min\limits_{R \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^d} & R^2 \\ \text{with} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2, \quad i = 1, \dots, n \end{cases} \qquad \begin{cases} \min\limits_{\mathbf{c}, \rho} & \frac{1}{2}\|\mathbf{c}\|^2 - \rho \\ \text{with} & \mathbf{c}^\top \mathbf{x}_i \geq \rho + \frac{1}{2}\|\mathbf{x}_i\|^2 \end{cases}$$

with $\rho = \frac{1}{2}(\|\mathbf{c}\|^2 - R^2)$

Proof:

$$\begin{array}{rcl} \|\mathbf{x}_i - \mathbf{c}\|^2 & \leq & R^2 \\ \|\mathbf{x}_i\|^2 - 2\mathbf{x}_i^\top \mathbf{c} + \|\mathbf{c}\|^2 & \leq & R^2 \\ -2\mathbf{x}_i^\top \mathbf{c} & \leq & R^2 - \|\mathbf{x}_i\|^2 - \|\mathbf{c}\|^2 \\ 2\mathbf{x}_i^\top \mathbf{c} & \geq & -R^2 + \|\mathbf{x}_i\|^2 + \|\mathbf{c}\|^2 \\ \mathbf{x}_i^\top \mathbf{c} & \geq & \underbrace{\frac{1}{2}(\|\mathbf{c}\|^2 - R^2)}_{\rho} + \frac{1}{2}\|\mathbf{x}_i\|^2 \end{array}$$

MEB and the one class SVM

SVDD: $$\begin{cases} \min\limits_{\mathbf{c},\rho} & \frac{1}{2}\|\mathbf{c}\|^2 - \rho \\ \text{with} & \mathbf{c}^\top \mathbf{x}_i \geq \rho + \frac{1}{2}\|\mathbf{x}_i\|^2 \end{cases}$$

SVDD and linear OCSVM (Supporting Hyperplane)

if $\forall i = 1, n, \ \|\mathbf{x}_i\|^2 = \text{constant}$, it is the the linear one class SVM (OC SVM)

The linear one class SVM [?]

$$\begin{cases} \min\limits_{\mathbf{c},\rho'} & \frac{1}{2}\|\mathbf{c}\|^2 - \rho' \\ \text{with} & \mathbf{c}^\top \mathbf{x}_i \geq \rho' \end{cases}$$

with $\rho' = \rho + \frac{1}{2}\|\mathbf{x}_i\|^2$ $\Rightarrow$ OC SVM is a particular case of SVDD

MEB: Lagrangian & KKT

$$\mathcal{L}(\mathbf{c}, R, \alpha) = R^2 + \sum_{i=1}^{n} \alpha_i \big(\|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 \big)$$

KKT conditions :

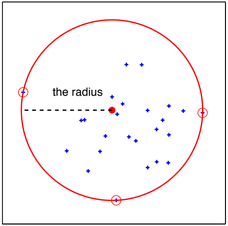
stationarity
- $2\mathbf{c} \sum_{i=1}^{n} \alpha_i - 2 \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = 0$ $\leftarrow$ The representer theorem
- $1 - \sum_{i=1}^{n} \alpha_i = 0$

primal admiss. $\|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2$

dual admiss. $\alpha_i \geq 0$ $\qquad i = 1, n$

MEB: Lagrangian & KKT

$$\mathcal{L}(\mathbf{c}, R, \alpha) = R^2 + \sum_{i=1}^{n} \alpha_i \big(\|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 \big)$$



KKT conditions :

stationarity
- $2\mathbf{c} \sum_{i=1}^{n} \alpha_i - 2 \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = 0$ ← The representer theorem
- $1 - \sum_{i=1}^{n} \alpha_i = 0$

primal admiss. $\|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2$

dual admiss. $\alpha_i \geq 0$ $\quad i = 1, n$

complementarity $\alpha_i \big(\|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 \big) = 0$ $\quad i = 1, n$

Complementarity tells us: two groups of points

the support vectors $\|\mathbf{x}_i - \mathbf{c}\|^2 = R^2$ and the insiders $\alpha_i = 0$

MEB: Dual

The representer theorem:

$$\nabla_{\mathbf{c}}\mathcal{L}(\mathbf{c}, R, \alpha) = 0 \qquad \Leftrightarrow \qquad \mathbf{c} = \frac{\sum_{i=1}^{n} \alpha_i \mathbf{x}_i}{\sum_{i=1}^{n} \alpha_i} = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i$$

The Lagrangian for the Dual

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i \big(\|\mathbf{x}_i - \sum_{j=1}^{n} \alpha_j \mathbf{x}_j\|^2\big)$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j x_i^\top x_j = \alpha^\top G\alpha \qquad \text{and} \qquad \sum_{i=1}^{n} \alpha_i x_i^\top x_i = \alpha^\top \text{diag}(G)$$

with $G = XX^\top$ the Gram matrix: $G_{ij} = x_i^\top x_j$,

The dual formulation of the MEB

$$\begin{cases} \min\limits_{\alpha \in \mathbb{R}^n} & \alpha^\top G\alpha - \alpha^\top diag(G) \\ \text{with} & e^\top \alpha = 1 \\ \text{and} & 0 \leq \alpha_i \qquad\qquad i = 1, \dots, n \end{cases}$$

SVDD primal vs. dual

Primal

$$\begin{cases} \min\limits_{R\in\mathbb{R},\mathbf{c}\in\mathbb{R}^d} & R^2 \\ \text{with} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2 \\ & i = 1, \dots, n \end{cases}$$

- $d + 1$ unknown
- n constraints
- can be recast as a QP
- perfect when $d << n$

Dual

$$\begin{cases} \min\limits_{\alpha} & \alpha^\top G\alpha - \alpha^\top diag(G) \\ \text{with} & e^\top \alpha = 1 \\ \text{and} & 0 \leq \alpha_i \\ & i = 1, \dots, n \end{cases}$$

- n unknown with G the pairwise influence Gram matrix
- n box constraints
- easy to solve
- to be used when $d > n$

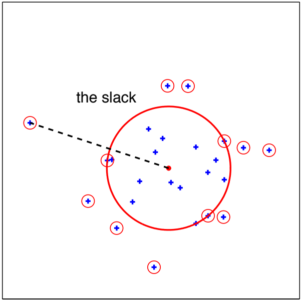
Plan

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

Dealing with outliers : a bi criteria optimization problem

Modeling potential errors: introducing slack variables ξ_i

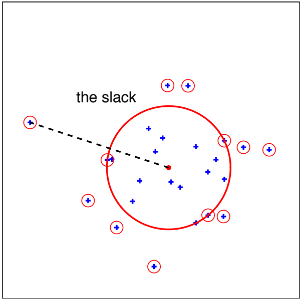
for all x_i $\begin{cases} \text{no error:} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2 \Rightarrow \quad \xi_i = 0 \\ \text{error:} & \|\mathbf{x}_i - \mathbf{c}\|^2 > R^2 \Rightarrow \quad \xi_i = \|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 \end{cases}$



$$\begin{cases} \min\limits_{R,\mathbf{c},\xi} & R^2 \\ \min\limits_{R,\mathbf{c},\xi} & \dfrac{1}{p}\displaystyle\sum_{i=1}^{n} \xi_i^p \\ \text{with} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2 + \xi_i, \quad i = 1, \dots, n \\ \text{and} & \xi_i \geq 0, \quad\quad\quad\quad i = 1, \dots, n \end{cases}$$

Our hope: almost all $\xi_i = 0$

The minimum enclosing ball problem with errors



The same road map:

- initial formulation
- reformulation (as a pQP)
- Lagrangian, KKT
- dual formulation
- bi dual

Initial formulation: for a given $\mu \geq 0$

$$\begin{cases} \min\limits_{R,\mathbf{c},\xi} & R^2 + \mu \sum_{i=1}^{n} \xi_i \\ \text{with} & \|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2 + \xi_i, \quad i = 1, \dots, n \\ \text{and} & \xi_i \geq 0, \quad i = 1, \dots, n \end{cases}$$

The MEB with slack: parametric QP, KKT, dual and R^2

SVDD as a pQP:

$$\begin{cases} \min\limits_{\mathbf{c},\rho} & \frac{1}{2}\|\mathbf{c}\|^2 - \rho + \frac{\mu}{2}\sum_{i=1}^{n}\xi_i \\ \text{with} & \mathbf{c}^\top \mathbf{x}_i \geq \rho + \frac{1}{2}\|\mathbf{x}_i\|^2 - \frac{1}{2}\xi_i \\ \text{and} & \xi_i \geq 0, \\ & i = 1, n \end{cases}$$

again with OC SVM as a particular case.
With $G = XX^\top$

Dual SVDD:

$$\begin{cases} \min\limits_{\alpha} & \alpha^\top G\alpha - \alpha^\top diag(G) \\ \text{with} & e^\top \alpha = 1 \\ \text{and} & 0 \leq \alpha_i \leq \mu, \\ & i = 1, n \end{cases}$$

for a given $\mu \leq 1$. If μ is larger than one it is useless (it's the no slack case)

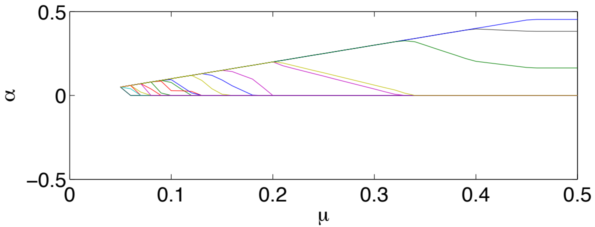
$$R^2 = \nu + \mathbf{c}^\top \mathbf{c}$$

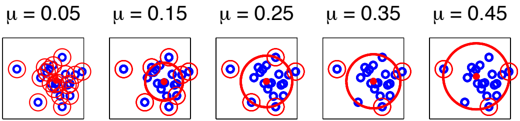
with ν denoting the Lagrange multiplier associated with the equality constraint $\sum_{i=1}^{n} \alpha_i = 1$.

Parametric QP

$$(\mathbf{c}^\star(\mu), \rho^\star(\mu)) = \left\{ \begin{array}{ll} \text{argmin}_{\mathbf{c},\rho} & \frac{1}{2}\|\mathbf{c}\|^2 - \rho + \frac{\mu}{2}\sum_{i=1}^{n} \xi_i \\ \text{with} & \mathbf{c}^\top \mathbf{x}_i \geq \rho + \frac{1}{2}\|\mathbf{x}_i\|^2 - \frac{1}{2}\xi_i \\ \text{and} & \xi_i \geq 0, \\ & i = 1, n \end{array} \right.$$

$$\alpha^\star(\mu) = \left\{ \begin{array}{ll} \text{argmin}_\alpha & \alpha^\top G \alpha - \alpha^\top d_G \\ \text{with} & e^\top \alpha = 1 \\ \text{and} & 0 \leq \alpha_i \leq \mu \\ & i = 1, n \end{array} \right.$$





Regularization path

$\alpha^\star(\mu)$ Piecewise linear

$$\alpha^\star(\mu') = \alpha^\star(\mu) + (\mu' - \mu)\mathbf{v}$$

SVDD as a penalized hinge loss minimization

The slack variables ξ_i

for all x_i $\begin{cases} \text{no error:} & \|\mathbf{x}_i - \mathbf{c}\|^2 \le R^2 \Rightarrow \quad \xi_i = 0 \\ \text{error:} & \|\mathbf{x}_i - \mathbf{c}\|^2 > R^2 \Rightarrow \quad \xi_i = \|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 \end{cases}$

$$\big(\mathbf{c}^*(\mu), \rho^*(\mu)\big) \;=\; \text{argmin}_{\mathbf{c},\rho} \quad \frac{1}{2}\|\mathbf{c}\|^2 - \rho + \mu \sum_{i=1}^{n} max\Big(0, \rho + \frac{1}{2}\|\mathbf{x}_i\|^2 - \mathbf{c}^\top \mathbf{x}_i\Big)$$

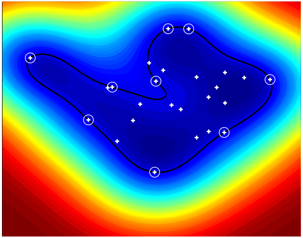
Generalize to

- other loss: exponential, logistic, L_0...
- other penalization: L_1, elastic net...

Plan

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

SVDD in a RKHS



The same road map:

- initial formulation
- reformulation (as a pQP)
- Lagrangian, KKT
- dual formulation
- bi dual

The feature map:

$$\begin{array}{rcl} \mathbb{R}^p & \longrightarrow & \mathcal{H} \\ c & \longrightarrow & f(\bullet) \\ \mathbf{x}_i & \longrightarrow & k(\mathbf{x}_i, \bullet) \\ \|\mathbf{x}_i - c\|_{\mathbb{R}^p} \leq R^2 & \longrightarrow & \|k(\mathbf{x}_i, \bullet) - f(\bullet)\|^2_{\mathcal{H}} \leq R^2 \end{array}$$

Kernelized SVDD (in a RKHS) is also a QP

$$\begin{cases} \displaystyle\min_{f \in \mathcal{H}, R \in \mathbb{R}, \xi \in \mathbb{R}^n} & R^2 + \mu \displaystyle\sum_{i=1}^{n} \xi_i \\ \text{with} & \|k(\mathbf{x}_i, \bullet) - f(\bullet)\|^2_{\mathcal{H}} \leq R^2 + \xi_i & i = 1, n \\ & \xi_i \geq 0 & i = 1, n \end{cases}$$

Equivalence between SVDD and OCSVM for translation invariant kernels (diagonal constant kernels)

Theorem

Let $\mathcal{H}$ be a RKHS on some domain $\mathbb{R}^p$ endowed with kernel k. If there exists some constant c such that $\forall \mathbf{x} \in \mathbb{R}^p,\ k(\mathbf{x},\mathbf{x}) = c$, then the two following problems are equivalent,

$$\begin{cases} \min\limits_{f,R,\xi} & R + \mu \sum\limits_{i=1}^{n} \xi_i \\ \text{with} & \|k(\mathbf{x}_i,.) - f(.)\|_{\mathcal{H}}^2 \le R + \xi_i \\ & \xi_i \ge 0 \qquad i = 1, n \end{cases} \qquad \begin{cases} \min\limits_{f,\rho,\xi} & \frac{1}{2}\|f\|_{\mathcal{H}}^2 - \rho + \mu \sum\limits_{i=1}^{n} \varepsilon_i \\ \text{with} & f(\mathbf{x}_i) \ge \rho - \varepsilon_i \\ & \varepsilon_i \ge 0 \qquad i = 1, n \end{cases}$$

with $\rho = \frac{1}{2}(c + \|f\|_{\mathcal{H}}^2 - R)$ and $\varepsilon_i = \frac{1}{2}\xi_i$.

SVDD in a RKHS: KKT, Dual and R^2

$$\begin{aligned}
\mathcal{L} &= R^2 + \mu \sum_{i=1}^n \xi_i + \sum_{i=1}^n \alpha_i \big(\|k(\mathbf{x}_i,.) - f(.)\|^2_{\mathcal{H}} - R^2 - \xi_i \big) - \sum_{i=1}^n \beta_i \xi_i \\
&= R^2 + \mu \sum_{i=1}^n \xi_i + \sum_{i=1}^n \alpha_i \big(k(\mathbf{x}_i,\mathbf{x}_i) - 2f(\mathbf{x}_i) + \|f\|^2_{\mathcal{H}} - R^2 - \xi_i \big) - \sum_{i=1}^n \beta_i \xi_i
\end{aligned}$$

KKT conditions

- Stationarity
  - $2f(.)\sum_{i=1}^n \alpha_i - 2\sum_{i=1}^n \alpha_i k(.,\mathbf{x}_i) = 0$ $\leftarrow$ The representer theorem
  - $1 - \sum_{i=1}^n \alpha_i = 0$
  - $\mu - \alpha_i - \beta_i = 0$
- Primal admissibility: $\|k(\mathbf{x}_i,.) - f(.)\|^2 \leq R^2 + \xi_i$, $\xi_i \geq 0$
- Dual admissibility: $\alpha_i \geq 0$, $\beta_i \geq 0$
- Complementarity
  - $\alpha_i \big(\|k(\mathbf{x}_i,.) - f(.)\|^2 - R^2 - \xi_i \big) = 0$
  - $\beta_i \xi_i = 0$

SVDD in a RKHS: Dual and R^2

$$\begin{aligned}
\mathcal{L}(\alpha) &= \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_i) - 2\sum_{i=1}^{n} f(\mathbf{x}_i) + \|f\|^2_{\mathcal{H}} \qquad \text{with } f(.) = \sum_{j=1}^{n} \alpha_j k(., \mathbf{x}_j) \\
&= \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j \underbrace{k(\mathbf{x}_i, \mathbf{x}_j)}_{G_{ij}}
\end{aligned}$$

$G_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$

$$\begin{cases}
\min\limits_{\alpha} & \alpha^\top G \alpha - \alpha^\top diag(G) \\
\text{with} & e^\top \alpha = 1 \\
\text{and} & 0 \leq \alpha_i \leq \mu, \qquad i = 1 \dots n
\end{cases}$$

As it is in the linear case:

$$R^2 = \nu + \|f\|^2_{\mathcal{H}}$$

with ν denoting the Lagrange multiplier associated with the equality constraint $\sum_{i=1}^{n} \alpha_i = 1$.

Kernelized SVDD primal vs. dual

Primal

$$
\begin{cases}
\min\limits_{f,R,\xi} & R + \mu \sum\limits_{i=1}^{n} \xi_i \\
\text{with} & \|k(\mathbf{x}_i, .) - f(.)\|^2_{\mathcal{H}} \leq R + \xi_i \\
& \xi_i \geq 0 \qquad i = 1, n
\end{cases}
$$

- $f \in \mathcal{H} + n + 1$ unknown
- $2n$ constraints
- can be recast as a QP
- intractable when $d = \infty$

Dual

$$
\begin{cases}
\min\limits_{\alpha} & \alpha^\top G \alpha - \alpha^\top diag(G) \\
\text{with} & e^\top \alpha = 1 \\
\text{and} & 0 \leq \alpha_i \leq \mu \\
& i = 1, \dots, n
\end{cases}
$$

- n unknown with G the pairwise influence Gram matrix
- $2n$ box constraints
- QP
- tractable

SVDD train and val in a RKHS

Train using the dual form (in: G, μ; out: α, ν)

$$\begin{cases} \min\limits_{\alpha} & \alpha^\top G\alpha - \alpha^\top diag(G) \\ \text{with} & e^\top \alpha = 1 \\ \text{and} & 0 \leq \alpha_i \leq \mu, \qquad i = 1 \dots n \end{cases}$$

Val with the center in the RKHS: $f(.) = \sum_{i=1}^n \alpha_i k(., \mathbf{x}_i)$

$$\begin{aligned} \phi(\mathbf{x}) &= \|k(\mathbf{x},.) - f(.)\|_{\mathcal{H}}^2 - R^2 \\ &= \|k(\mathbf{x},.)\|_{\mathcal{H}}^2 - 2\langle k(\mathbf{x},.), f(.)\rangle_{\mathcal{H}} + \|f(.)\|_{\mathcal{H}}^2 - R^2 \\ &= k(\mathbf{x},\mathbf{x}) - 2f(\mathbf{x}) + R^2 - \nu - R^2 \\ &= -2f(\mathbf{x}) + k(\mathbf{x},\mathbf{x}) - \nu \\ &= -2\sum_{i=1}^n \alpha_i k(\mathbf{x},\mathbf{x}_i) + k(\mathbf{x},\mathbf{x}) - \nu \end{aligned}$$

$\phi(\mathbf{x}) = 0$ is the decision border

An important theoretical result

For a well-calibrated bandwidth, The kernel SVDD estimates the underlying distribution level set [?]

The level sets of a probability density function $\mathbb{P}(\mathbf{x})$ are the set

$$C_p = \{\mathbf{x} \in \mathbb{R}^d \mid \mathbb{P}(\mathbf{x}) \geq p\}$$

It is well estimated by the empirical minimum volume set

$$V_p = \{\mathbf{x} \in \mathbb{R}^d \mid \|k(\mathbf{x}, .) - f(.)\|^2_{\mathcal{H}} - R^2 \geq 0\}$$

The frontiers coincides

Roadmap

1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

SVDD + outlier: the problem

$$
\begin{cases}
\min_{R,c,\xi} & R + \mu \sum_{i=1}^{n} \xi_i \\
\text{with} & \|x_i - c\|^2 \leq R + m + \xi_i, \quad i = 1, \dots, n \\
\text{and} & \xi_i \geq 0, \quad\quad\quad\quad\quad\quad\quad i = 1, \dots, n
\end{cases}
\qquad (1)
$$

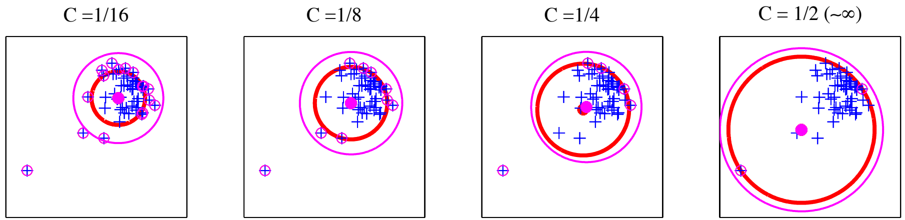


Figure : Example of SVDD solutions with different μ values, $m = 0$ (red) and $m = 5$ (magenta). The circled data points represent support vectors for both m.

Chasing outliers with the L_0 (pseudo) norm

SVDD is sensitive to the presence of outliers in the data
Allowing t outliers (and no errors)

$$\|\xi\|_0 = \text{card}\{i|\xi_i \neq 0\} \;\leq\; t$$

L_0 SVDD

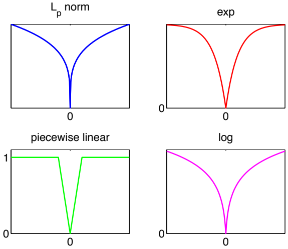
$$\begin{cases} \displaystyle\min_{c\in\mathbb{R}^p, R\in\mathbb{R}, \xi\in\mathbb{R}^n} & R + \mu\|\xi\|_0 \\ \text{with} & \|\mathbf{x}_i - c\|^2 \leq R + \xi_i \\ & \xi_i \geq 0 \qquad i = 1, n \end{cases}$$

However, the L_0 pseudo-norm is

non differentiable, combinatorially hard and does not lead to an effective algorithmic approach

L_0 relaxations

- p norm $\|\xi\|_p = \left(\sum_{i=1}^n |\xi_i|^p\right)^{\frac{1}{p}}$ $\quad p \to 0$
- exponential $\sum_{i=1}^n \left(1 - \exp^{-\alpha \xi_i}\right)$ $\quad \alpha \to \infty$
- piecewise linear $\sum_{i=1}^n \min\left(1, \frac{|\xi_i|}{\alpha}\right)$ $\quad \alpha \to 0$
- log $\sum_{i=1}^n \log\left(1 + \frac{|\xi_i|}{\alpha}\right)$ $\quad \alpha \to \infty$



L_0 log relaxation SVDD

$$\begin{cases} \min\limits_{c\in\mathbb{R}^p, R\in\mathbb{R}, \xi\in\mathbb{R}^n} & R + \mu \sum_{i=1}^n \log(\gamma + \xi_i) \\ \text{with} & \|\mathbf{x}_i - c\|^2 \le R + \xi_i \\ & \xi_i \ge 0 \qquad i = 1, n \end{cases}$$

The L_0 log relaxation SVDD is differentiable, however

it is not convex

DC programing

The DC (Difference of Convex Functions)

$$\log(\gamma + \xi) = f(\xi) - g(\xi) \qquad \text{with} \quad \begin{array}{l} f(\xi) = \xi \\ g(\xi) = \xi - \log(\gamma + \xi), \end{array}$$

both functions f and g being convex.

The DC algorithm consists in minimizing iteratively the convex term:

$$\log(\gamma + \xi) \quad \longrightarrow \quad f(\xi) - g'(\xi^{\text{old}})\xi \quad \begin{aligned} &= \xi - \left(1 - \frac{1}{\gamma + \xi^{\text{old}}}\right)\xi \\ &= \underbrace{\frac{1}{\gamma + \xi^{\text{old}}}}_{w}\, \xi \qquad \text{with} \quad w = \frac{1}{\gamma + \xi^{\text{old}}} \end{aligned}$$

where ξ_i^{old} denotes the solution at the previous iteration.

DC applied to the L_0 SVDD log relaxation

$$\begin{cases} \min_{c,R,\xi} & R + \mu \|\xi\|_0 \\ \text{with} & \|\mathbf{x}_i - c\|^2 \leq R + \xi_i \\ & \xi_i \geq 0 \qquad i = 1, n \end{cases} \longrightarrow \begin{cases} \min_{c,R,\xi} & R + \mu \sum_{i=1}^{n} \log(\gamma + \xi_i) \\ \text{with} & \|\mathbf{x}_i - c\|^2 \leq R + \xi_i \\ & \xi_i \geq 0 \qquad i = 1, n \end{cases}$$

The DC idea applied to our L_0 SVDD approximation consists in building a sequence of solutions of the following adaptive SVDD:

while not converged do

$$\begin{cases} \min_{c \in \mathbb{R}^p, R \in \mathbb{R}, \xi \in \mathbb{R}^n} & R + \mu \sum_{i=1}^{n} w_i \xi_i \\ \text{with} & \|\mathbf{x}_i - c\|^2 \leq R + \xi_i \\ & \xi_i \geq 0 \qquad i = 1, n \end{cases} \qquad \text{with} \qquad w_i = \frac{1}{\gamma + \xi_i^{\text{old}}}.$$

$$\xi_i^{\text{old}} = \xi_i, \qquad i = 1, n$$

Dual formulation (to be used with kernels)

$$\mathcal{L}(\mathbf{c}, R, \xi, \alpha, \gamma) = R^2 + \mu \sum_{i=1}^{n} w_i \xi_i + \sum_{i=1}^{n} \alpha_i \big(\|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 - \xi_i\big) - \sum_{i=1}^{n} \gamma_i \xi_i$$

KKT conditions :

stationarity
- $2\mathbf{c} \sum_{i=1}^{n} \alpha_i - 2 \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = 0$ ← The representer theorem
- $1 - \sum_{i=1}^{n} \alpha_i = 0$
- $\mu w_i - \alpha_i - \gamma_i = 0$ $\quad i = 1, n$

primal admiss. $\|\mathbf{x}_i - \mathbf{c}\|^2 \leq R^2 + \xi_i$ $\quad i = 1, n$

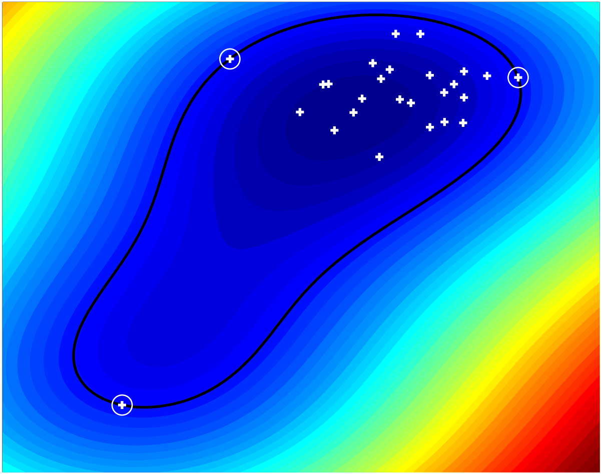
dual admiss. $\alpha_i \geq 0, \gamma_i \geq 0$ $\quad i = 1, n$

complementarity $\alpha_i \big(\|\mathbf{x}_i - \mathbf{c}\|^2 - R^2 - \xi_i\big) = 0$ $\quad i = 1, n$

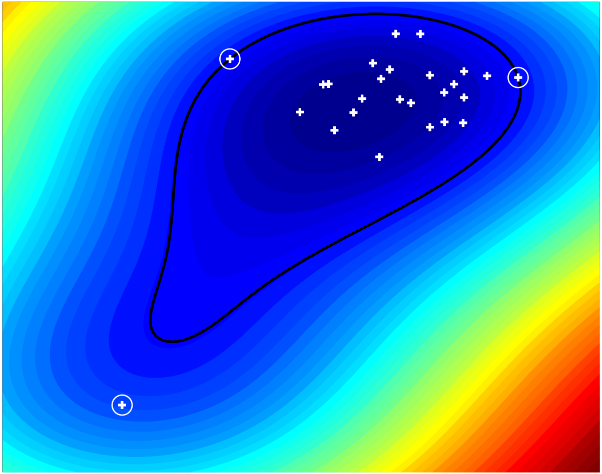
Adaptive SVDD in the Dual

$$\begin{cases} \min\limits_{\alpha \in \mathbb{R}^n} & \alpha^\top X X^\top \alpha - \alpha^\top diag(XX^\top) \\ \text{with} & \sum_{i=1}^{n} \alpha_i = 1 \qquad 0 \leq \alpha_i \leq \mu w_i \qquad i = 1, n \end{cases} \tag{2}$$

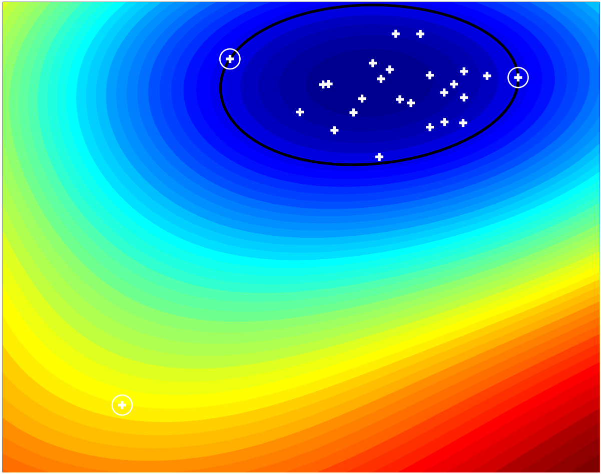
L_0 SVDD at work



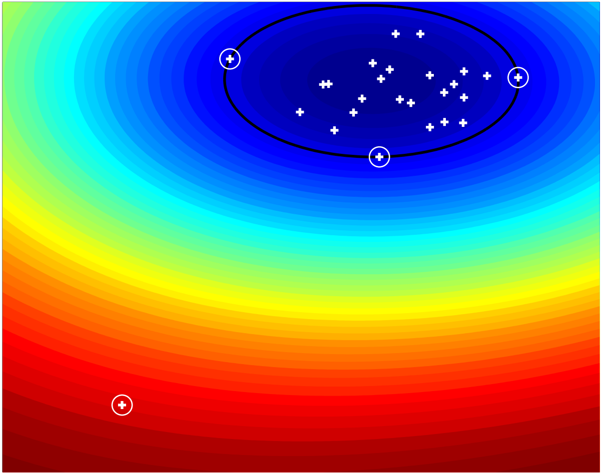
L_0 SVDD at work



L_0 SVDD at work

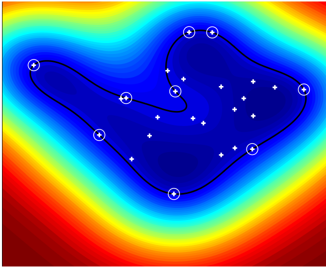


L_0 SVDD at work



Conclusion

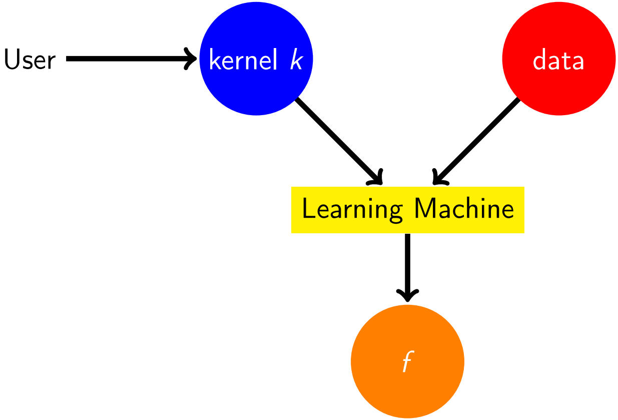
- Applications
  - outlier detection
  - change detection
  - clustering
  - large number of classes
  - variable selection…
- A clear path
  - reformulation (to a standard problem)
  - KKT
  - Dual
  - Bidual
- a lot of variations
  - L^2 SVDD
  - two classes non symmetric
  - two classes in the symmetric classes (SVM)
  - the multi classes issue
- problems with non translation invariant kernels



Roadmap

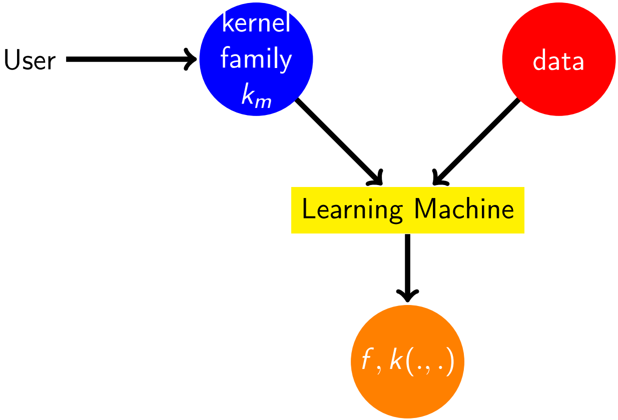
1. Kernels and kernel machines
   - Kernelizing the linear regression
   - Kernels
2. Suport vector machines
   - Supervised classification and prediction
   - Linear SVM
   - The non separable case
   - Kernelized support vector machine
3. Support Vector Data Description (SVDD)
   - SVDD, the smallest enclosing ball problem
   - The minimum enclosing ball problem with errors
   - The minimum enclosing ball problem in a RKHS
   - Robust outlier detection with L0-SVDD
4. Tuning the kernel: multiple kernel learning (MKL)

Standard Learning with Kernels



http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf

Learning Kernel framework



http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf

from SVM

- SVM: single kernel k

$$
\begin{aligned}
f(\mathbf{x}) &= \sum_{i=1}^{n} \alpha_i \qquad k \;(\mathbf{x}, \mathbf{x}_i) + b \\
&=
\end{aligned}
$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

from SVM → to Multiple Kernel Learning (MKL)

- SVM: single kernel k
- MKL: set of M kernels $k_1, \dots, k_m, \dots, k_M$
  - learn classier and combination weights
  - can be cast as a convex optimization problem

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m \, k_m(\mathbf{x}, \mathbf{x}_i) + b \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m$$

$$=$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

from SVM → to Multiple Kernel Learning (MKL)

- SVM: single kernel k
- MKL: set of M kernels $k_1, \ldots, k_m, \ldots, k_M$
  - learn classier and combination weights
  - can be cast as a convex optimization problem

$$
\begin{aligned}
f(\mathbf{x}) &= \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m \; k_m(\mathbf{x}, \mathbf{x}_i) + b \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m \\
&= \sum_{i=1}^{n} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \qquad \text{with} \qquad K(\mathbf{x}, \mathbf{x}_i) = \sum_{m=1}^{M} d_m \; k_m(\mathbf{x}, \mathbf{x}_i)
\end{aligned}
$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

Multiple Kernel

The model

$$f(x) = \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m k_m(x, x_i) + b, \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m$$

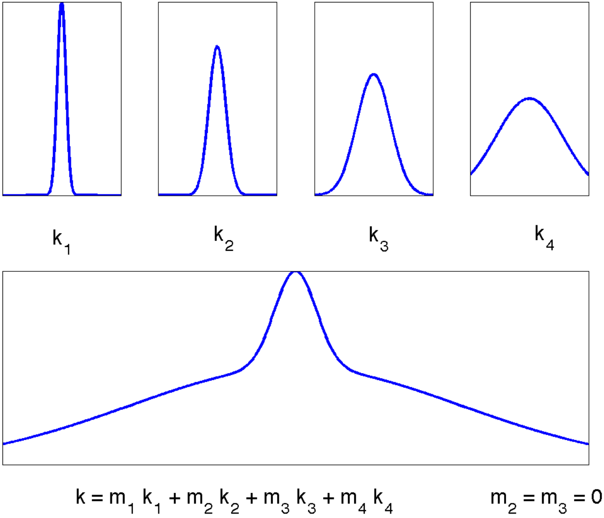
Given M kernel functions $k_1, \ldots, k_M$ that are potentially well suited for a given problem, find a positive linear combination of these kernels such that the resulting kernel k is "optimal"

$$k(\mathbf{x}, \mathbf{x}') = \sum_{m=1}^{M} d_m k_m(\mathbf{x}, \mathbf{x}'), \text{ with } d_m \geq 0, \sum_m d_m = 1$$

Learning together

The kernel coefficients d_m and the SVM parameters α_i, b.

Multiple Kernel: illustration



Multiple Kernel Strategies

- Wrapper method (Weston et al., 2000; Chapelle et al., 2002)
  - solve SVM
  - gradient descent on d_m on criterion:
    - margin criterion
    - span criterion
- Kernel Learning & Feature Selection
  - use Kernels as dictionary
- Embedded Multi Kernel Learning (MKL)

Multiple Kernel functional Learning

The problem (for given C)

$$\begin{aligned} \min_{f\in\mathcal{H},b,\xi,d} \quad & \frac{1}{2}\|f\|^2_{\mathcal{H}} + C\sum_i \xi_i \\ \text{with} \quad & y_i\big(f(x_i)+b\big) \geq 1+\xi_i \; ; \quad \xi_i \geq 0 \quad \forall i \\ & \sum_{m=1}^{M} d_m = 1 \; , \quad d_m \geq 0 \quad \forall m \; , \end{aligned}$$

$$f = \sum_m f_m \qquad \text{and} \qquad k(\mathbf{x},\mathbf{x}') = \sum_{m=1}^{M} d_m k_m(\mathbf{x},\mathbf{x}'), \text{ with } d_m \geq 0$$

The functional framework

$$\mathcal{H} = \bigoplus_{m=1}^{M} \mathcal{H}'_m \qquad\qquad \langle f, g\rangle_{\mathcal{H}'_m} = \frac{1}{d_m}\langle f, g\rangle_{\mathcal{H}_m}$$

Multiple Kernel functional Learning

The problem (for given C)

$$\min_{\{f_m\},b,\xi,d} \quad \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i$$
$$\text{with} \quad y_i\Big(\sum_m f_m(x_i) + b\Big) \geq 1 + \xi_i \; ; \quad \xi_i \geq 0 \quad \forall i$$
$$\sum_m d_m = 1 \; , \quad d_m \geq 0 \quad \forall m \; ,$$

Treated as a bi-level optimization task

$$\min_{d \in \mathbb{R}^M} \left\{ \begin{array}{ll} \displaystyle\min_{\{f_m\},b,\xi} & \displaystyle\frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i \\ \text{with} & y_i\Big(\displaystyle\sum_m f_m(x_i) + b\Big) \geq 1 + \xi_i \; ; \quad \xi_i \geq 0 \quad \forall i \end{array} \right.$$
$$\text{s.t.} \quad \sum_m d_m = 1 \; , \quad d_m \geq 0 \quad \forall m \; ,$$

Multiple Kernel representer theorem and dual

The Lagrangian:

$$\mathcal{L} = \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i - \sum_i \alpha_i\Big(y_i\big(\sum_m f_m(x_i) + b\big) - 1 - \xi_i\Big) - \sum_i \beta_i\xi_i$$

Associated KKT stationarity conditions:

$$\nabla_m\mathcal{L} = 0 \quad \Leftrightarrow \quad \frac{1}{d_m} f_m(\bullet) = \sum_{i=1}^n \alpha_i y_i k_m(\bullet, \mathbf{x}_i) \qquad m = 1, M$$

Representer theorem

$$f(\bullet) = \sum_m f_m(\bullet) = \sum_{i=1}^n \alpha_i y_i \underbrace{\sum_m d_m k_m(\bullet, \mathbf{x}_i)}_{K(\bullet, \mathbf{x}_i)}$$

We have a standard SVM problem with respect to function f and kernel K.

Multiple Kernel Algorithm

Use a Reduced Gradient Algorithm[1]

$$\begin{array}{ll} \min_{d \in \mathbb{R}^M} & J(d) \\ \text{s.t.} & \sum_m d_m = 1 \;, \quad d_m \geq 0 \quad \forall m \;, \end{array}$$

SimpleMKL algorithm

set $d_m = \frac{1}{M}$ for $m = 1, \dots, M$
while stopping criterion not met **do**
  compute $J(d)$ using an QP solver with $K = \sum_m d_m K_m$
  compute $\frac{\partial J}{\partial d_m}$, and projected gradient as a descent direction D
  $\gamma \leftarrow$ compute optimal stepsize
  $d \leftarrow d + \gamma D$
end while

$\longrightarrow$ Improvement reported using the Hessian

[1] Rakotomamonjy et al. JMLR 08

Complexity

For each iteration:

- SVM training: $O(nn_{sv} + n_{sv}^3)$.
- Inverting $K_{sv,sv}$ is $O(n_{sv}^3)$, but might already be available as a by-product of the SVM training.
- Computing H: $O(Mn_{sv}^2)$
- Finding d: $O(M^3)$.

The number of iterations is usually less than 10.

$\longrightarrow$ When $M < n_{sv}$, computing d is not more expensive than QP.

MKL on the 101-caltech dataset

Performance of recent methods applied to Caltech-101. Note that (*) combines [Gehler et al. ICCV'09] and our features.

Method	15 train	30 train
LP-beta(*) P. Gehler and S. Nowozin, ICCV'09.	74.6 ± 1.0	82.1 ± 0.3
Group-sensitive multiple kernel learning for object categorization. J. Yang, Y. Li, Y. Tian, L. Duan, and W. In Proc. ICCV, 2009.	73.2	84.3
Bayesian localized multiple kernel learning. M. Christoudias, R. Urtasun, and T. Darrell. *Technical report, UC Berkeley,* 2009.	73.0 ± 1.3	NA
In defense of nearest-neighbor based image classification. O. Boiman, E. Shechtman, and M. Irani. In *Proc. CVPR*, 2008.	72.8	≈79
This method.	71.1 ± 0.6	78.2 ± 0.4
On feature combination for multiclass object classification. P. Gehler and S. Nowozin. In *Proc. ICCV*, 2009.	70.4 ± 0.8	77.7 ± 0.3
Recognition using regions. C. Gu, J. J. Lim, P. Arbelàez, and J. Malik. In *Proc. CVPR*, 2009.	65.0	73.1
SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. H. Zhang, A. C. Berg, M. Maire, and J. Malik. In *Proc. CVPR*, 2006.	59.06 ± 0.56	66.23 ± 0.48

http://www.robots.ox.ac.uk/~vgg/software/MKL/

Conclusion on multiple kernel (MKL)

- MKL: Kernel tuning, variable selection...
  - extention to classification and one class SVM
- SVM KM: an efficient Matlab toolbox (available at MLOSS)[2]
- Multiple Kernels for Image Classification: Software and Experiments on Caltech-101[3]
- new trend: Multi kernel, Multi task and ∞ number of kernels

---

[2] http://mloss.org/software/view/33/

[3] http://www.robots.ox.ac.uk/~vgg/software/MKL/

Bibliography

- A. Rakotomamonjy, F. Bach, S. Canu & Y. Grandvalet. SimpleMKL. J. Mach. Learn. Res. 2008, 9:2491–2521.
- M. Gönen & E. Alpaydin Multiple kernel learning algorithms. J. Mach. Learn. Res. 2008;12:2211-2268.

- http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf
- http://www.robots.ox.ac.uk/~vgg/software/MKL/
- http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf