SVM and Kernel machine
Lecture 1: Linear SVM

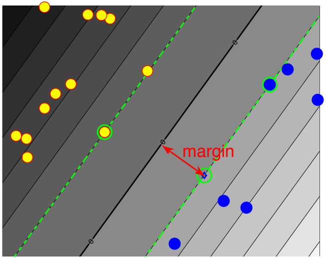
Stéphane Canu
stephane.canu@litislab.eu

Sao Paulo 2014

March 12, 2014

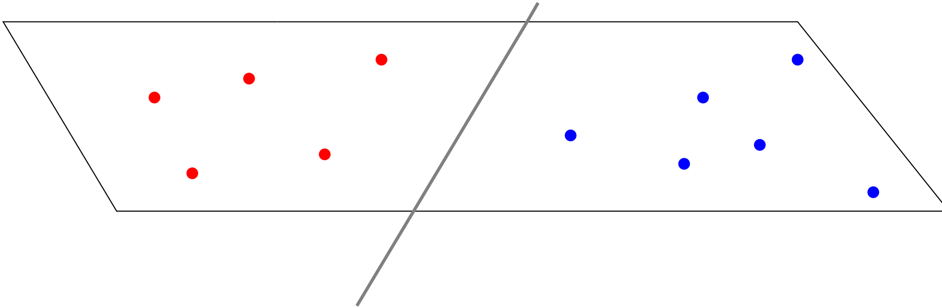
Road map

1. Linear SVM
   - Separating hyperplanes
   - The margin
   - Linear SVM: the problem
   - Linear programming SVM



"The algorithms for constructing the separating hyperplane considered above will be utilized for developing a battery of programs for pattern recognition." in Learning with kernels, 2002 - from V .Vapnik, 1982

Hyperplanes in 2d: intuition



It's a line!

Hyperplanes: formal definition

Given vector $\mathbf{v} \in \mathbb{R}^d$ and bias $a \in \mathbb{R}$

Hyperplane as a function h,

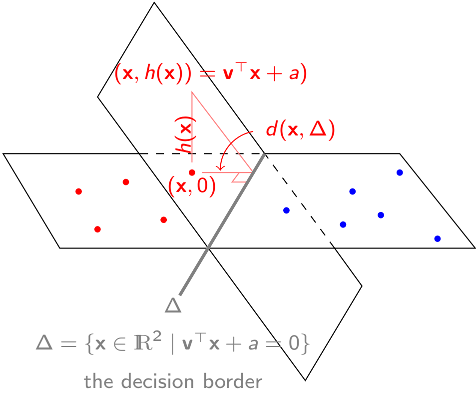
$$h: \begin{array}{ccc} \mathbb{R}^d & \longrightarrow & \mathbb{R} \\ x & \longmapsto & h(x) = \mathbf{v}^\top \mathbf{x} + a \end{array}$$

Hyperplane as a border in $\mathbb{R}^d$
(and an implicit function)

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

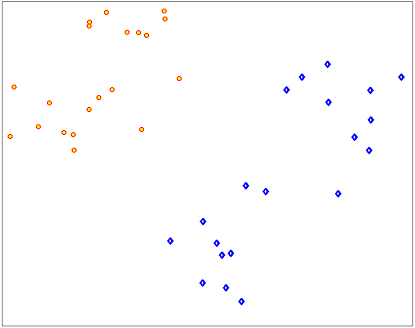
The border invariance property

$$\forall k \in \mathbb{R}, \quad \Delta(k\mathbf{v}, ka) = \Delta(\mathbf{v}, a)$$



Separating hyperplanes

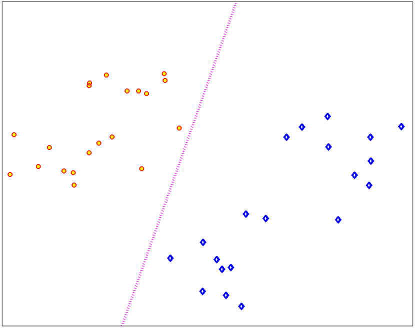
Find a line to separate (classify) blue from red



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

Separating hyperplanes

Find a line to separate (classify) blue from red



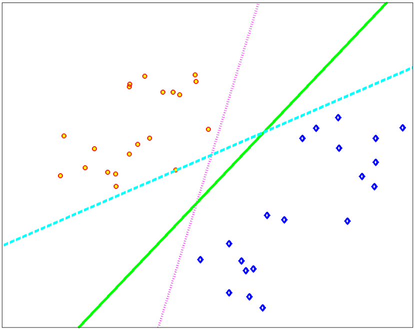
$$D(x) = \text{sign}\left(\mathbf{v}^\top \mathbf{x} + a\right)$$

the decision border:

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

Separating hyperplanes

Find a line to separate (classify) blue from red



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

the decision border:

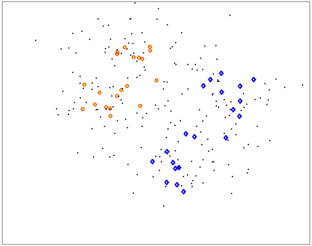
$$\mathbf{v}^\top \mathbf{x} + a = 0$$

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

How to choose a solution?

This is not the problem we want to solve

$\{(\mathbf{x}_i, y_i);\ i = 1 : n\}$ a training sample, i.i.d. drawn according to $\mathbb{P}(\mathbf{x}, y)$
unknown

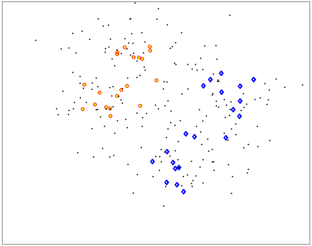


we want to be able to classify new observations: minimize $\mathbb{P}(\text{error})$

This is not the problem we want to solve

$\{(\mathbf{x}_i, y_i);\ i = 1:n\}$ a training sample, i.i.d. drawn according to $\mathbb{P}(\mathbf{x}, y)$

unknown



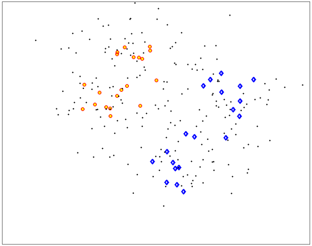
we want to be able to classify new observations: minimize $\mathbb{P}(\text{error})$

Looking for a universal approach

- use training data: (a **few** errors)
- prove $\mathbb{P}(error)$ remains small
- scalable - algorithmic complexity

This is not the problem we want to solve

$\{(\mathbf{x}_i, y_i);\ i = 1 : n\}$ a training sample, i.i.d. drawn according to $\mathbb{P}(\mathbf{x}, y)$ unknown



we want to be able to classify new observations: minimize $\mathbb{P}(\text{error})$

Looking for a universal approach

- use training data: (a **few** errors)
- prove $\mathbb{P}(\mathit{error})$ remains small
- scalable - algorithmic complexity

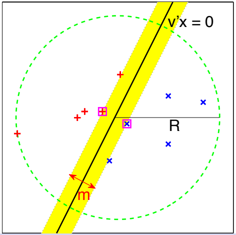
with high probability (for the canonical hyperplane):

$$\mathbb{P}(\text{error}) < \underbrace{\widehat{\mathbb{P}}(\text{error})}_{=0 \text{ here}} + \varphi\Big(\frac{1}{\underbrace{\text{margin}}_{=\|\mathbf{v}\|}} \Big)$$

Vapnik's Book, 1982

Margin guarantees

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



Theorem (Margin Error Bound)

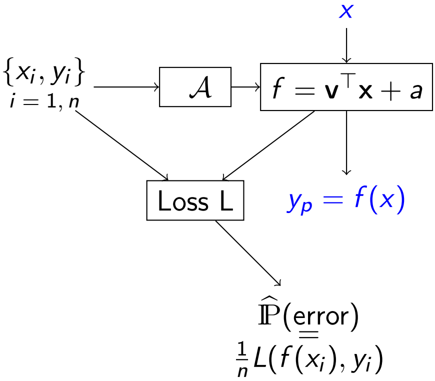
Let R be the radius of the smallest ball $B_R(a) = \{x \in \mathbb{R}^d \mid \|\mathbf{x} - \mathbf{c}\| < R\}$, containing the points $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ i.i.d from some unknown distribution $\mathbb{P}$. Consider a decision function $D(\mathbf{x}) = sign(\mathbf{v}^\top \mathbf{x})$ associated with a separating hyperplane $\mathbf{v}$ of margin m (no training error).

Then, with probability at least $1-\delta$ for any $\delta > 0$, the generalization error of this hyperplane is bounded by

$$\mathbb{P}(error) \ \leq 2\sqrt{\frac{R^2}{n\ m^2}} + 3\sqrt{\frac{\ln(2/\delta)}{2n}}$$

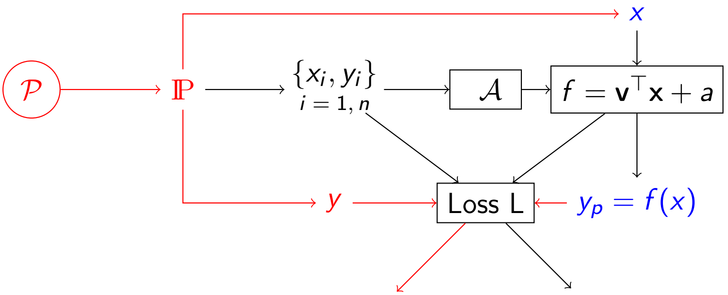
theorem 4.17 p 102 in J Shawe-Taylor, N Cristianini Kernel methods for pattern analysis, Cambridge 2004

Statistical machine learning – Computation learning theory (COLT)



Vapnik's Book, 1982

Statistical machine learning – Computation learning theory (COLT)

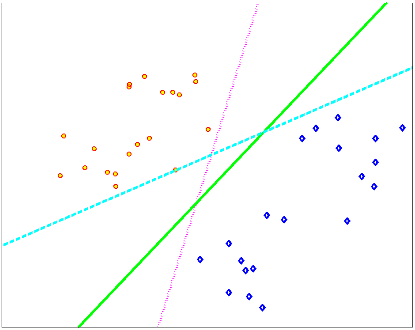


$$\forall \mathbb{P} \in \mathcal{P} \qquad \text{Prob}\Bigg(\underset{\mathbb{E}(L)}{\underset{=}{\mathbb{P}(\text{error})}} \quad \le \quad \underset{\frac{1}{n} L(f(x_i), y_i)}{\underset{=}{\widehat{\mathbb{P}}(\text{error})}} \quad + \quad \varphi(\|\mathbf{v}\|) \Bigg) \ge \delta$$

Vapnik's Book, 1982

linear discrimination

Find a line to classify blue and red



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

the decision border:

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

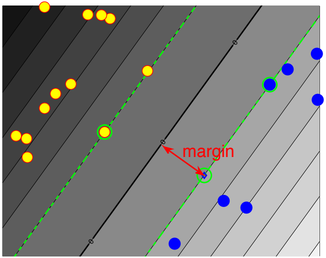
there are many solutions...
The problem is ill posed

How to choose a solution ?

$\Rightarrow$ choose the one with larger margin

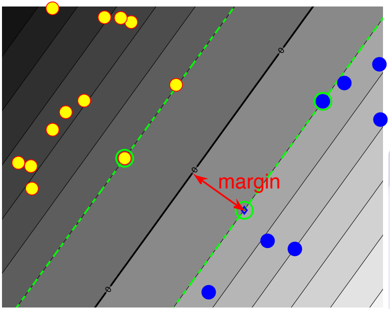
Road map

1. Linear SVM
   - Separating hyperplanes
   - The margin
   - Linear SVM: the problem
   - Linear programming SVM



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

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

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...

Margin and distance: details

Theorem (The geometrical margin)

Let $\mathbf{x}$ *be a vector in* $\mathbb{R}^d$ *and* $\Delta(\mathbf{v}, a) = \{\mathbf{s} \in \mathbb{R}^d \mid \mathbf{v}^\top \mathbf{s} + a = 0\}$ *an hyperplane. The distance between vector* $\mathbf{x}$ *and the hyperplane* $\Delta(\mathbf{v}, a))$ *is*

$$dist(\mathbf{x}_i, \Delta(\mathbf{v}, a)) \quad = \quad \frac{|\mathbf{v}^\top \mathbf{x} + a|}{\|\mathbf{v}\|}$$

Let $\mathbf{s}_x$ be the closest point to $\mathbf{x}$ in Δ , $\mathbf{s}_x = \arg\min_{\mathbf{s} \in \Delta} \|\mathbf{x} - \mathbf{s}\|$. Then

$$\mathbf{x} = \mathbf{s}_x + r \frac{\mathbf{v}}{\|\mathbf{v}\|} \qquad \Leftrightarrow \qquad r \frac{\mathbf{v}}{\|\mathbf{v}\|} = \mathbf{x} - \mathbf{s}_x$$

So that, taking the scalar product with vector $\mathbf{v}$ we have:

$$\mathbf{v}^\top r \frac{\mathbf{v}}{\|\mathbf{v}\|} = \mathbf{v}^\top (\mathbf{x} - \mathbf{s}_x) = \mathbf{v}^\top \mathbf{x} - \mathbf{v}^\top \mathbf{s}_x = \mathbf{v}^\top \mathbf{x} + a - \underbrace{(\mathbf{v}^\top \mathbf{s}_x + a)}_{=0} = \mathbf{v}^\top \mathbf{x} + a$$

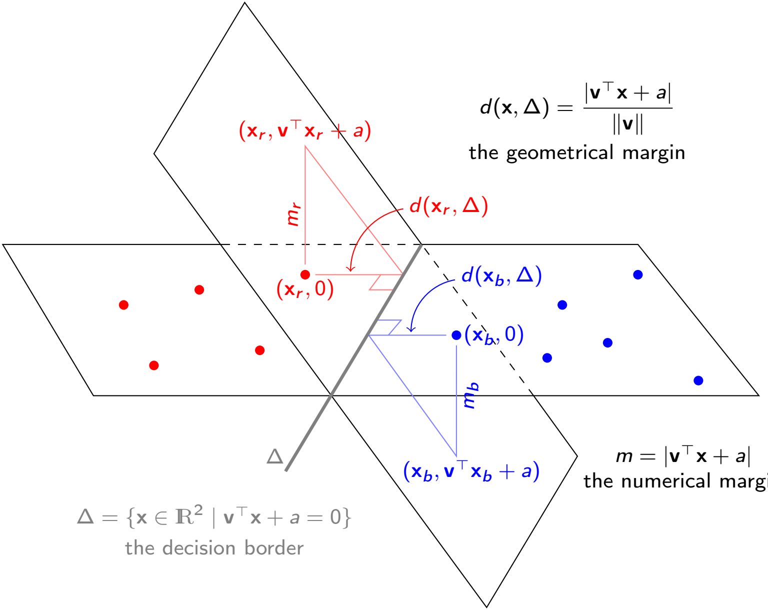
and therefore

$$r = \frac{\mathbf{v}^\top \mathbf{x} + a}{\|\mathbf{v}\|}$$

leading to:

$$\mathrm{dist}(\mathbf{x}_i, \Delta(\mathbf{v}, a)) = \min_{\mathbf{s} \in \Delta} \|\mathbf{x} - \mathbf{s}\| = r = \frac{|\mathbf{v}^\top \mathbf{x} + a|}{\|\mathbf{v}\|}$$

Geometrical and numerical margin



$$d(\mathbf{x}, \Delta) = \frac{|\mathbf{v}^\top \mathbf{x} + a|}{\|\mathbf{v}\|}$$

the geometrical margin

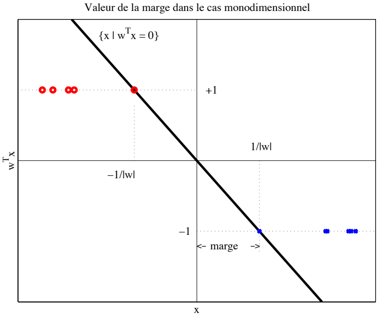
$$m = |\mathbf{v}^\top \mathbf{x} + a|$$

the numerical margin

$$\Delta = \{\mathbf{x} \in \mathbb{R}^2 \mid \mathbf{v}^\top \mathbf{x} + a = 0\}$$

the decision border

From the geometrical to the numerical margin



Maximize the (geometrical) margin

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

if the min is greater, everybody is greater ($y_i \in \{-1, 1\}$)

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

change variable: $\mathbf{w} = \frac{\mathbf{v}}{m\|\mathbf{v}\|}$ and $b = \frac{a}{m\|\mathbf{v}\|} \implies \|\mathbf{w}\| = \frac{1}{m}$

$$\begin{cases} \max\limits_{\mathbf{w},b} & m \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \quad ; \; i = 1, n \\ \text{and} & m = \frac{1}{\|\mathbf{w}\|} \end{cases}$$

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

The canonical hyperplane

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

Definition (The canonical hyperplane)

An hyperplane $(\mathbf{w}, b)$ in $\mathbb{R}^d$ is said to be canonical with respect the set of vectors $\{\mathbf{x}_i \in \mathbb{R}^d, i = 1, n\}$ if

$$\min_{i=1,n} |\mathbf{w}^\top \mathbf{x}_i + b| = 1$$

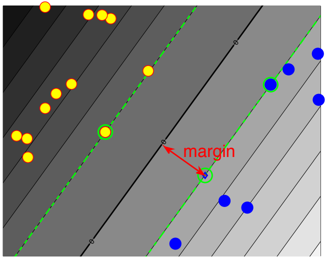
so that the distance

$$\min_{i=1,n} \text{dist}(\mathbf{x}_i, \Delta(\mathbf{w}, b)) = \frac{|\mathbf{w}^\top \mathbf{x} + b|}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|}$$

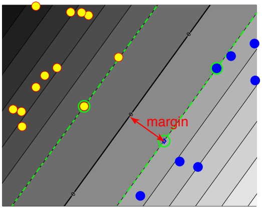
The maximal margin (=minimal norm) canonical hyperplane

Road map

1. Linear SVM
   - Separating hyperplanes
   - The margin
   - Linear SVM: the problem
   - Linear programming SVM



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:

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

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

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

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

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

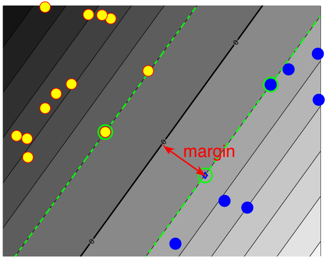
```
% QUADPROG Quadratic programming.
%    X = QUADPROG(H,f,A,b) attempts to solve the quadratic programming problem:
%
%              min 0.5*x'*H*x + f'*x   subject to:  A*x <= b
%               x
%  so that the solution is in the range LB <= X <= UB
```

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

Road map

1. Linear SVM
   - Separating hyperplanes
   - The margin
   - Linear SVM: the problem
   - Linear programming SVM



Other SVMs: Equivalence between norms

- L_1 norm
- variable selection (especially with redundant noisy features)
- Mangassarian, 1965

$$\begin{cases} \max\limits_{m,\mathbf{v},a} & m \\ \text{with} & y_i(\mathbf{v}^\top \mathbf{x}_i + a) \geq m \,\|\mathbf{v}\|_2 \geq m \,\frac{1}{\sqrt{d}}\|\mathbf{v}\|_1 \\ & i = 1, n \end{cases}$$

1-norm or Linear Programming-SVM (LP SVM)

$$\begin{cases} \min\limits_{\mathbf{w},b} & \|\mathbf{w}\|_1 = \sum_{j=1}^{p} |w_j| \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \; ; \quad i = 1, n \end{cases}$$

Generalized SVM (Bradley and Mangasarian, 1998)

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

p = 2: SVM, p = 1: LPSVM (also with $p = \infty$), p = 0: L_0 SVM,
p= 1 and 2: doubly regularized SVM (DrSVM)

Linear support vector support (LP SVM)

$$\begin{cases} \min\limits_{\mathbf{w},b} & \|\mathbf{w}\|_1 = \sum_{j=1}^{p} w_j^+ + w_j^- \\ \text{with} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \; ; \qquad\qquad i = 1, n \end{cases}$$

$$\mathbf{w} = \mathbf{w}^+ - \mathbf{w}^- \qquad \text{with} \qquad \mathbf{w}^+ \geq 0 \text{ and } \mathbf{w}^- \geq 0$$

The Standart LP formulation

$$\begin{cases} \min\limits_{\mathbf{x}} & \mathbf{f}^\top \mathbf{x} \\ \text{with} & A\mathbf{x} \leq \mathbf{d} \\ \text{and} & 0 \leq \mathbf{x} \end{cases}$$

$$\mathbf{x} = [\mathbf{w}^+; \mathbf{w}^-; b] \quad f = [1 \ldots 1; 0] \quad \mathbf{d} = -[1 \ldots 1]^\top \quad A = [-yiXi \; yiXi \; -yi]$$

```
% linprog(f,A,b,Aeq,beq,LB,UB)
%       attempts to solve the linear programming problem:
%             min f'*x    subject to:   A*x <= b
%              x
%  so that the solution is in the range LB <= X <= UB
```

An example of linear discrimination: SVM and LPSVM

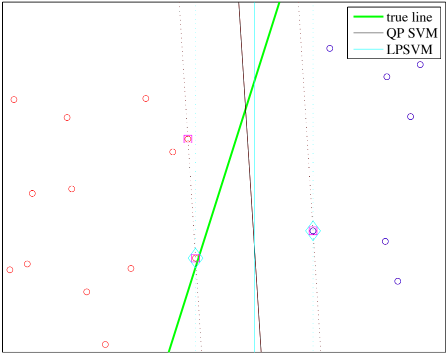
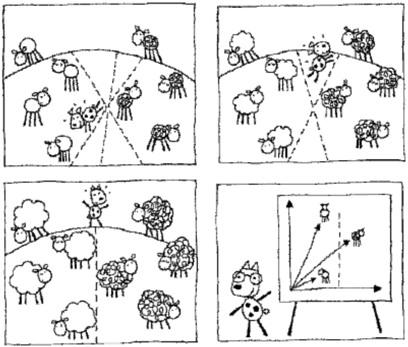


Figure: SVM and LP SVM

The linear discrimination problem



...the story of the sheep dog who was herding his sheep, and serendipitously invented the large margin classification and Sheep Vectors ...

(drawing by Ana Martin Larranaga)

from *Learning with Kernels*, B. Schölkopf and A. Smolla, MIT Press, 2002.

Conclusion

SVM =

- Separating hyperplane (to begin with the simpler)
- + Margin, Norm and statistical learning
- + Quadratic and Linear programming (and associated rewriting issues)
- + Support vectors (sparsity)

SVM preforms the selection of the most relevant data points

Bibliography

- V. Vapnik, *the generalized portrait method* p 355 in Estimation of dependences based on empirical data, Springer, 1982
- B. Boser, I. Guyon & V. Vapnik, A training algorithm for optimal margin classifiers. COLT, 1992
- P. S. Bradley & O. L. Mangasarian. Feature selection via concave minimization and support vector machines. ICML 1998
- B. Schölkopf & A. Smolla, *Learning with Kernels*, MIT Press, 2002
- M. Mohri, A. Rostamizadeh & A. Talwalkar, *Foundations of Machine Learning*, MIT press 2012

- http://agbs.kyb.tuebingen.mpg.de/lwk/sections/section72.pdf
- http://www.cs.nyu.edu/~mohri/mls/lecture_4.pdf
- http://en.wikipedia.org/wiki/Quadratic_programming