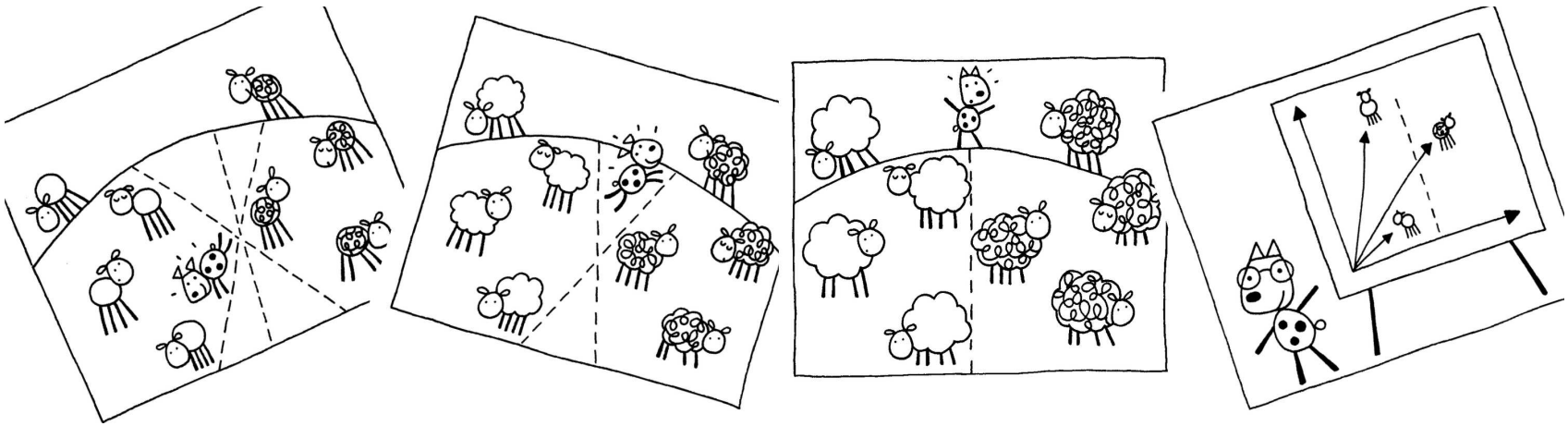
Learning with kernels and SVM

Šámalova chata, 23. května, 2006

Petra Kudová



Outline

Learning from data

- find a general rule that explains data given only as a sample of limited size
- data may contain measurement errors or noise

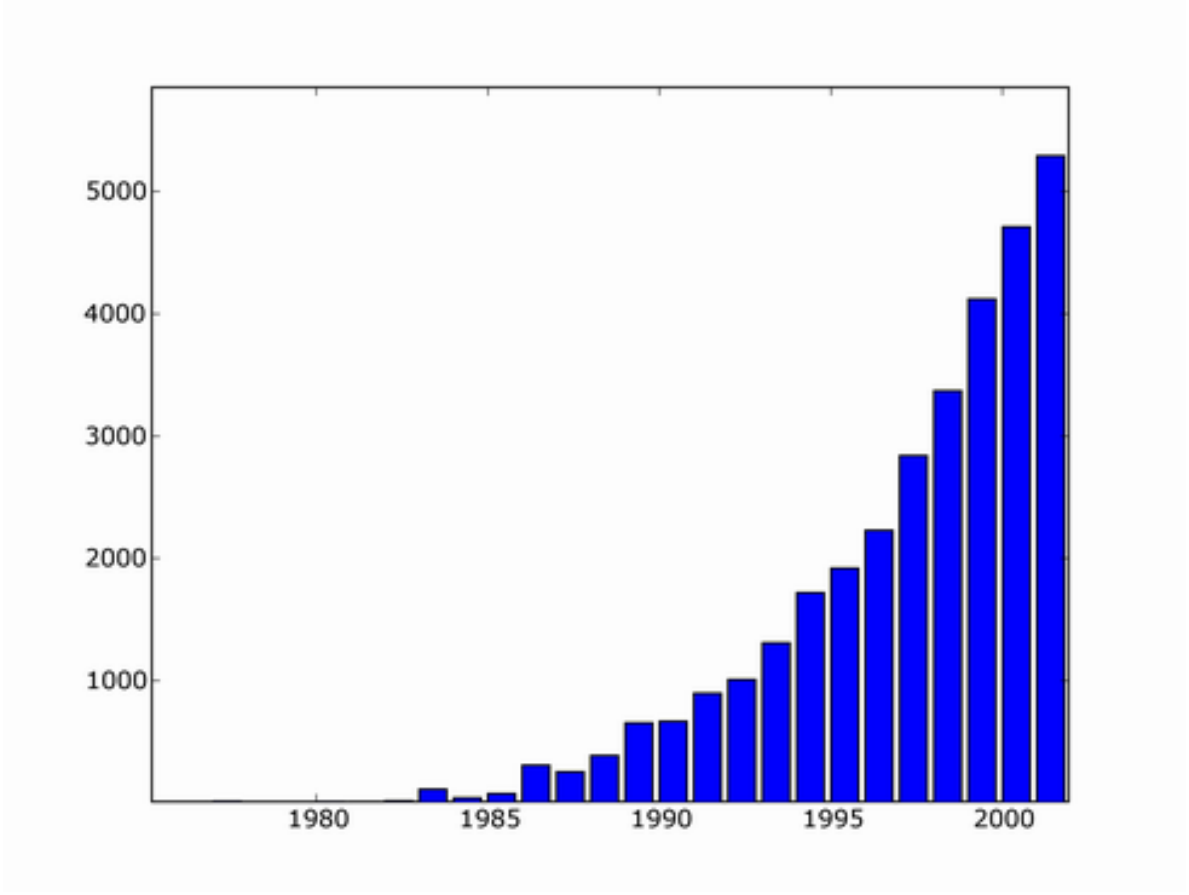
- supervised learning
  - data are sample of input-output pairs
  - find input-output mapping
  - prediction, classification, function approximation, etc.
- unsupervised learning
  - data are sample of objects
  - find some structure
  - clustering, etc.

Learning methods

- wide range of methods available
- statistical approaches
- neural networks
  - originally biological motivation
  - Multi-layer perceptrons, RBF networks
  - Kohonen maps
- kernel methods
  - modern and popular
  - SVM

Trends in machine learning

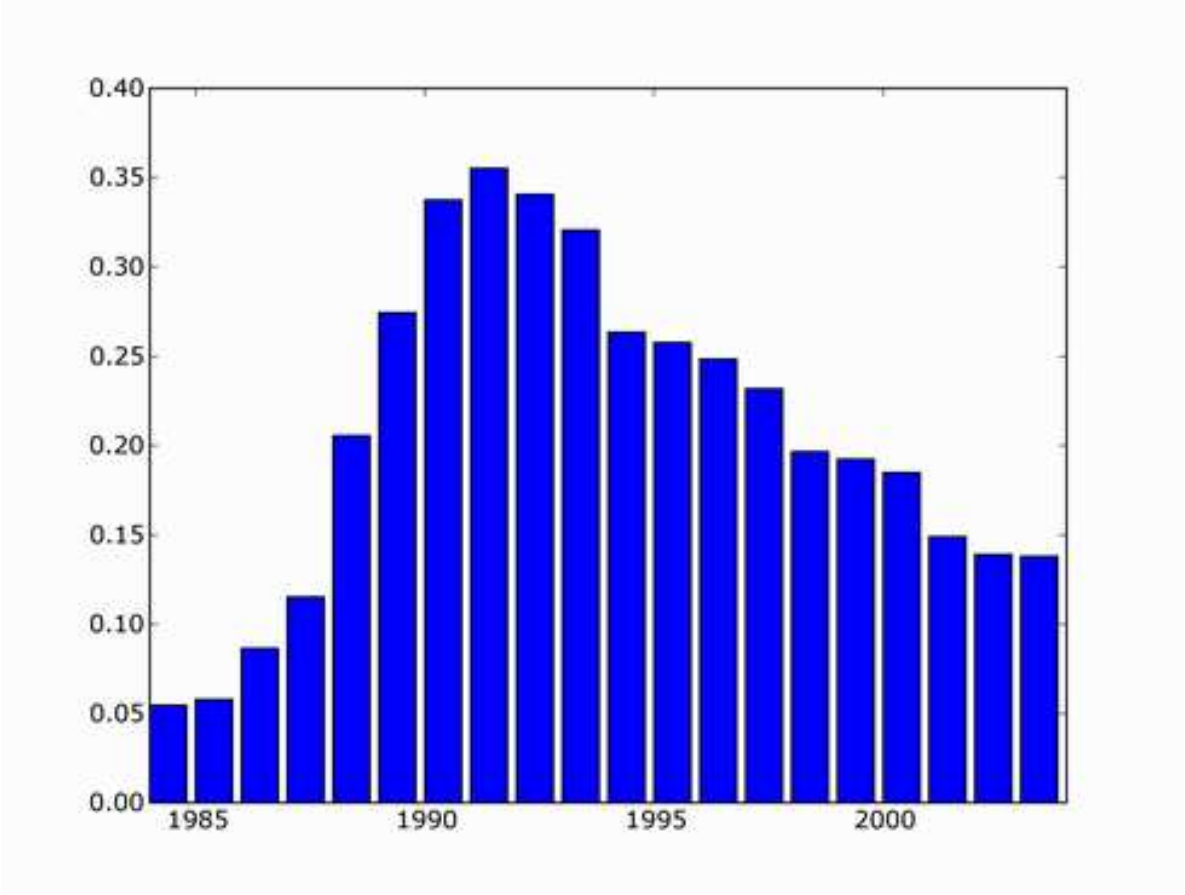
Articles on machine learning found by Google



Source: http://yaroslavvb.blogspot.com/

Trends in machine learning

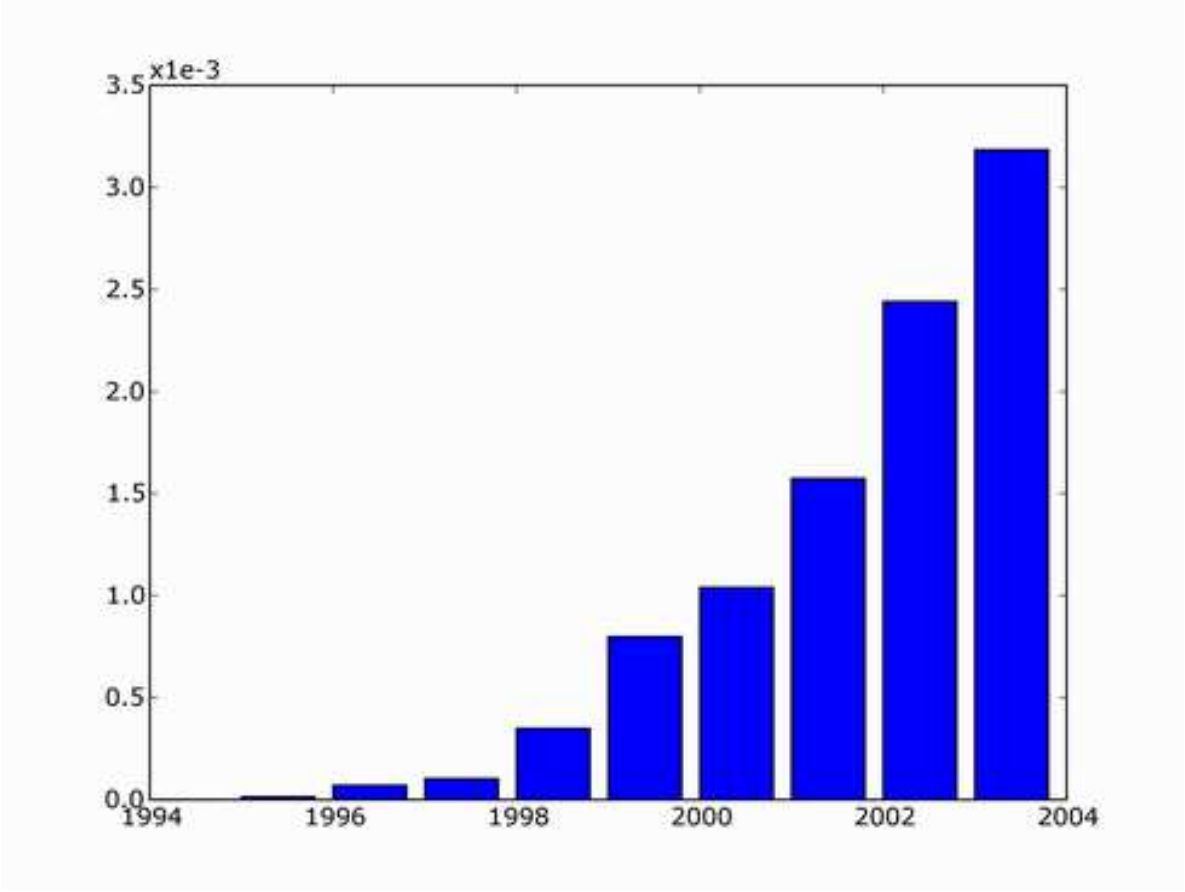
Articles on neural networks found by Google



Source: http://yaroslavvb.blogspot.com/

Trends in machine learning

Articles on suport vector machine found by Google



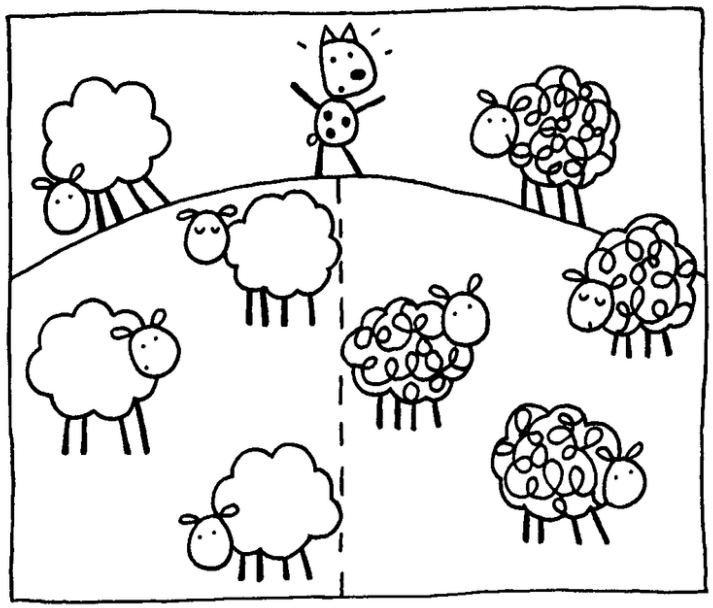
Source: http://yaroslavvb.blogspot.com/

Binary classification

- Training set

$$\{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$$
$$\mathbf{x}_i \in \mathcal{X}$$
$$y_i \in \{-1, 1\}$$

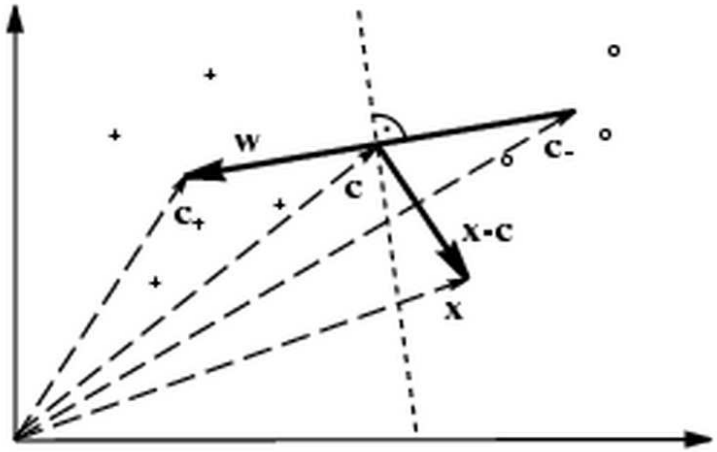
- find classifier
- generalization



Simple Classifier

Suppose: $\mathcal{X} \subset \mathbb{R}^n$, classes linearly separable

- $\mathbf{c}_+ = \frac{1}{m_+} \sum_{\{i|y_i=+1\}} \mathbf{x}_i$
- $\mathbf{c}_- = \frac{1}{m_-} \sum_{\{i|y_i=-1\}} \mathbf{x}_i$
- $\mathbf{c} = \frac{1}{2}(\mathbf{c}_+ + \mathbf{c}_-)$



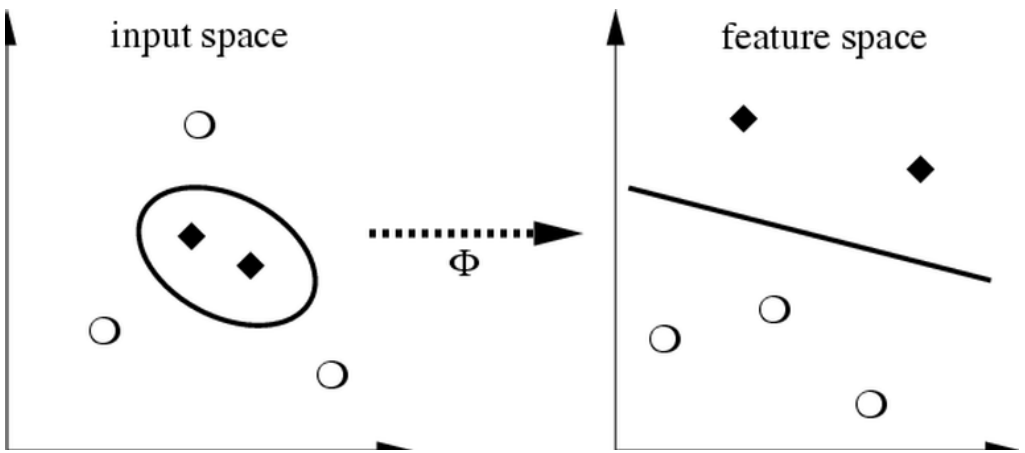
- $y = sgn(\langle(\mathbf{x} - \mathbf{c}), \mathbf{w}\rangle)$
  $= sgn(\langle(\mathbf{x} - (\mathbf{c}_+ + \mathbf{c}_-)/2), ((\mathbf{c}_+ + \mathbf{c}_-)\rangle)$
  $= sgn(\langle\mathbf{x}, \mathbf{c}_+\rangle - \langle\mathbf{x}, \mathbf{c}_-\rangle + b)$
- $b = \frac{1}{2}(||\mathbf{c}_-||^2 - ||\mathbf{c}_+||^2)$

Mapping to the feature space

- life is not so easy, not all problems are linearly separable
- what to do if $\mathcal{X}$ is not dot-product space?

- choose a mapping to some (high dimensional) dot-product space - *feature space*

  $\Phi : \mathcal{X} \to \mathcal{H}$



Mercer's condition and Kernels

If a symmetric function $K(\mathbf{x}, \mathbf{y})$ satisfies

$$\sum_{i,j=1}^{M} a_i a_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0$$

for all $M \in \mathbb{N}, \mathbf{x}_i$, and $a_i \in \mathbb{R}$, there exists a mapping function Φ that maps $\mathbf{x}$ into the dot-product feature space and

$$K(\mathbf{x}, \mathbf{y}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle$$

and vice versa.

Function K is called kernel.

Examples of kernels

- Linear Kernels

$$K(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$$

- Polynomial Kernels

$$K(\mathbf{x}, \mathbf{y}) = (\langle \mathbf{x}, \mathbf{y} \rangle + 1)^d$$

for $d = 2$ and 2-dimensional inputs

$$\begin{aligned} K(\mathbf{x}, \mathbf{y}) &= 1 + 2x_1y_1 + 2x_2y_2 + 2x_1y_1x_2y_2 + x_1^2y_1^2 + x_2^2y_2^2 \\ &= \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}) \rangle \\ \Phi(\mathbf{x}) &= (1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1x_2, x_1^2x_2^2)^T \end{aligned}$$

Examples of kernels

- RBF Kernels

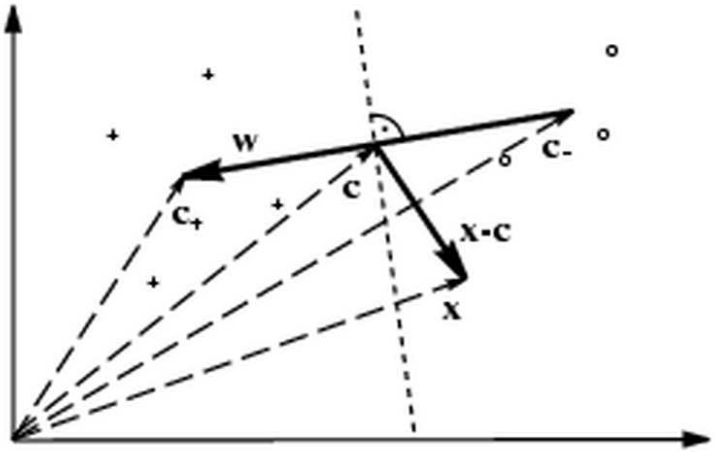
$$K(\mathbf{x}, \mathbf{y}) = exp(-\frac{||\mathbf{x} - \mathbf{y}||^2}{d^2})$$

- Other kernels
  - kernels on various objects, such as graphs, strings, texts, etc.
  - enable us to use dot-product algorithms
  - measure of similarity

Simple Classifier - kernel version

Suppose: $\mathcal{X} \subset \mathbb{R}^n$, classes linearly separable

- $\mathbf{c}_+ = \frac{1}{m_+} \sum_{\{i|y_i=+1\}} \mathbf{x}_i$
- $\mathbf{c}_- = \frac{1}{m_-} \sum_{\{i|y_i=-1\}} \mathbf{x}_i$
- $\mathbf{c} = \frac{1}{2}(\mathbf{c}_+ + \mathbf{c}_-)$

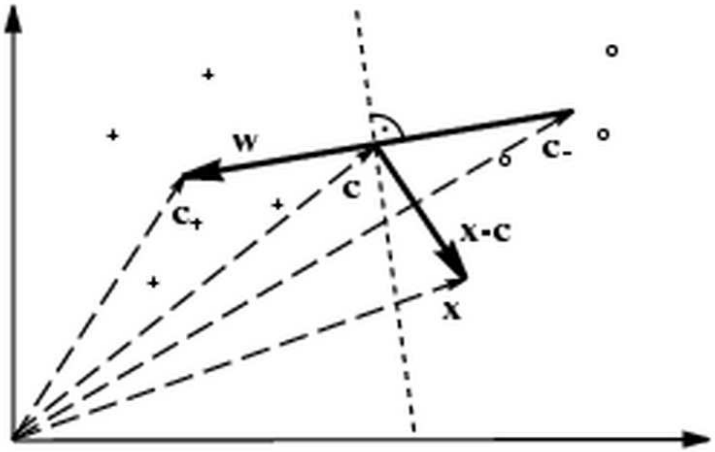


- $y = sgn(\langle(\mathbf{x} - \mathbf{c}), \mathbf{w}\rangle)$
  $= sgn(\langle(\mathbf{x} - (\mathbf{c}_+ + \mathbf{c}_-)/2), ((\mathbf{c}_+ + \mathbf{c}_-)\rangle)$
  $= sgn(\langle \mathbf{x}, \mathbf{c}_+ \rangle - \langle \mathbf{x}, \mathbf{c}_- \rangle + b)$
- $b = \frac{1}{2}(||\mathbf{c}_-||^2 - ||\mathbf{c}_+||^2)$

Simple Classifier - kernel version

Suppose: $\mathcal{X}$ is any set, $\Phi : \mathcal{X} \to \mathcal{H}$ corresponding to kernel K

- $\mathbf{c}_+ = \frac{1}{m_+} \sum_{\{i|y_i=+1\}} \mathbf{x}_i$
- $\mathbf{c}_- = \frac{1}{m_-} \sum_{\{i|y_i=-1\}} \mathbf{x}_i$
- $\mathbf{c} = \frac{1}{2}(\mathbf{c}_+ + \mathbf{c}_-)$



- $y = sgn(\langle(\mathbf{x} - \mathbf{c}), \mathbf{w}\rangle)$
  $= sgn(\langle(\mathbf{x} - (\mathbf{c}_+ + \mathbf{c}_-)/2), ((\mathbf{c}_+ + \mathbf{c}_-)\rangle)$
  $= sgn(\langle\mathbf{x}, \mathbf{c}_+\rangle - \langle\mathbf{x}, \mathbf{c}_-\rangle + b)$
- $b = \frac{1}{2}(||\mathbf{c}_-||^2 - ||\mathbf{c}_+||^2)$

Simple Classifier - kernel version

Suppose: $\mathcal{X}$ is any set, $\Phi : \mathcal{X} \to \mathcal{H}$ corresponding to kernel K

$$y = sgn(\frac{1}{m_+} \sum_{\{i|y_i=+1\}} K(\mathbf{x}, \mathbf{x}_i) - \frac{1}{m_-} \sum_{\{i|y_i=-1\}} K(\mathbf{x}, \mathbf{x}_i) + b)$$

$$b = \frac{1}{2}(\frac{1}{m_-^2} \sum_{\{i,j|y_i=y_j=-1\}} K(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{m_+^2} \sum_{\{i,j|y_i=y_j=+1\}} K(\mathbf{x}_i, \mathbf{x}_j))$$

Statistical approach Bayes classifier - special case

$$\int_{\mathcal{X}} K(\mathbf{x}, \mathbf{y}) d\mathbf{x} = 1 \qquad \forall \mathbf{y} \in \mathcal{X}; \quad b = 0$$

Simple Classifier - kernel version

Suppose: $\mathcal{X}$ is any set, $\Phi : \mathcal{X} \to \mathcal{H}$ corresponding to kernel K

$$y = sgn(\frac{1}{m_+} \sum_{\{i|y_i=+1\}} K(\mathbf{x}, \mathbf{x}_i) - \frac{1}{m_-} \sum_{\{i|y_i=-1\}} K(\mathbf{x}, \mathbf{x}_i) + 0)$$

$$= p_+(\mathbf{x}) \qquad\qquad = p_-(\mathbf{x})$$

Parzen windows

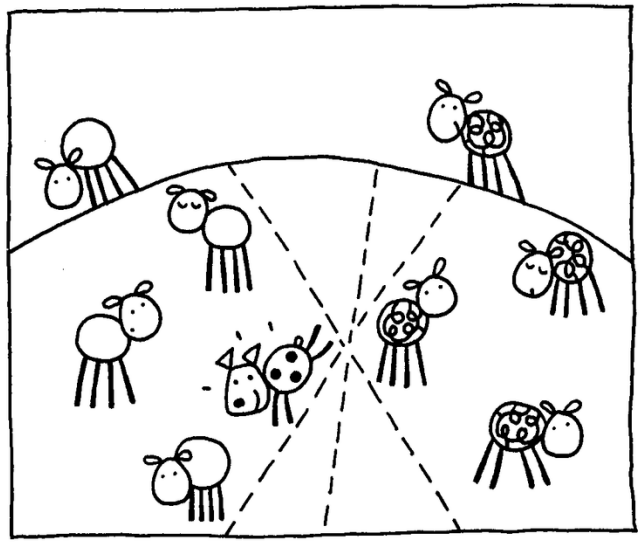
Statistical approach Bayes classifier - special case

$$\int_{\mathcal{X}} K(\mathbf{x}, \mathbf{y}) d\mathbf{x} = 1 \qquad \forall \mathbf{y} \in \mathcal{X}; \quad b = 0$$

Separating hyperplane

- classifier in a form
  $y(\mathbf{x}) = sgn(\langle \mathbf{w}, \mathbf{x} \rangle + b)$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \begin{cases} > 0 & \text{for } y_i = 1 \\ < 0 & \text{for } y_i = -1 \end{cases}$$



- each hyperplane

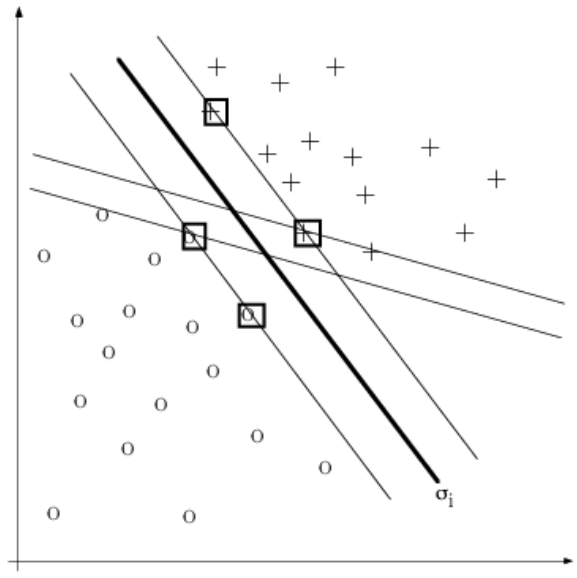
  $D(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b = c, \quad -1 < c < 1$ is separating

- optimal separating hyperplane - the one with the maximal margin

Separating hyperplane

- classifier in a form
  $y(\mathbf{x}) = sgn(\langle \mathbf{w}, \mathbf{x} \rangle + b)$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \begin{cases} \geq 1 & \text{for } y_i = 1 \\ \leq -1 & \text{for } y_i = -1 \end{cases}$$



- each hyperplane

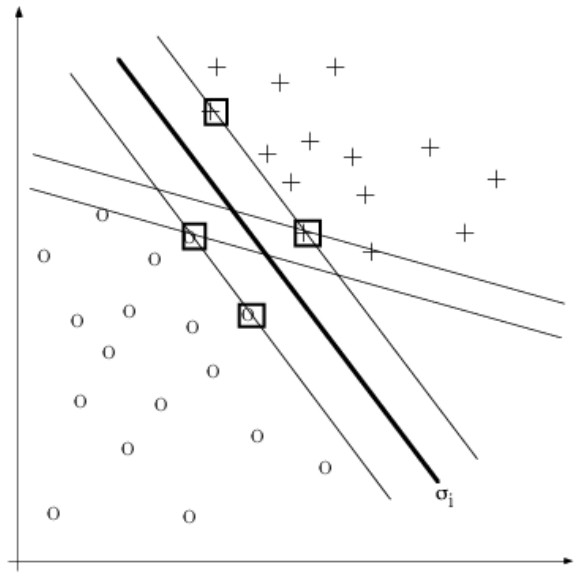
  $D(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b = c, \quad -1 < c < 1$ is separating

- optimal separating hyperplane - the one with the maximal margin

Separating hyperplane

- classifier in a form
  $y(\mathbf{x}) = sgn(\langle \mathbf{w}, \mathbf{x} \rangle + b)$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \begin{cases} \geq 1 & \text{for } y_i = 1 \\ \leq -1 & \text{for } y_i = -1 \end{cases}$$

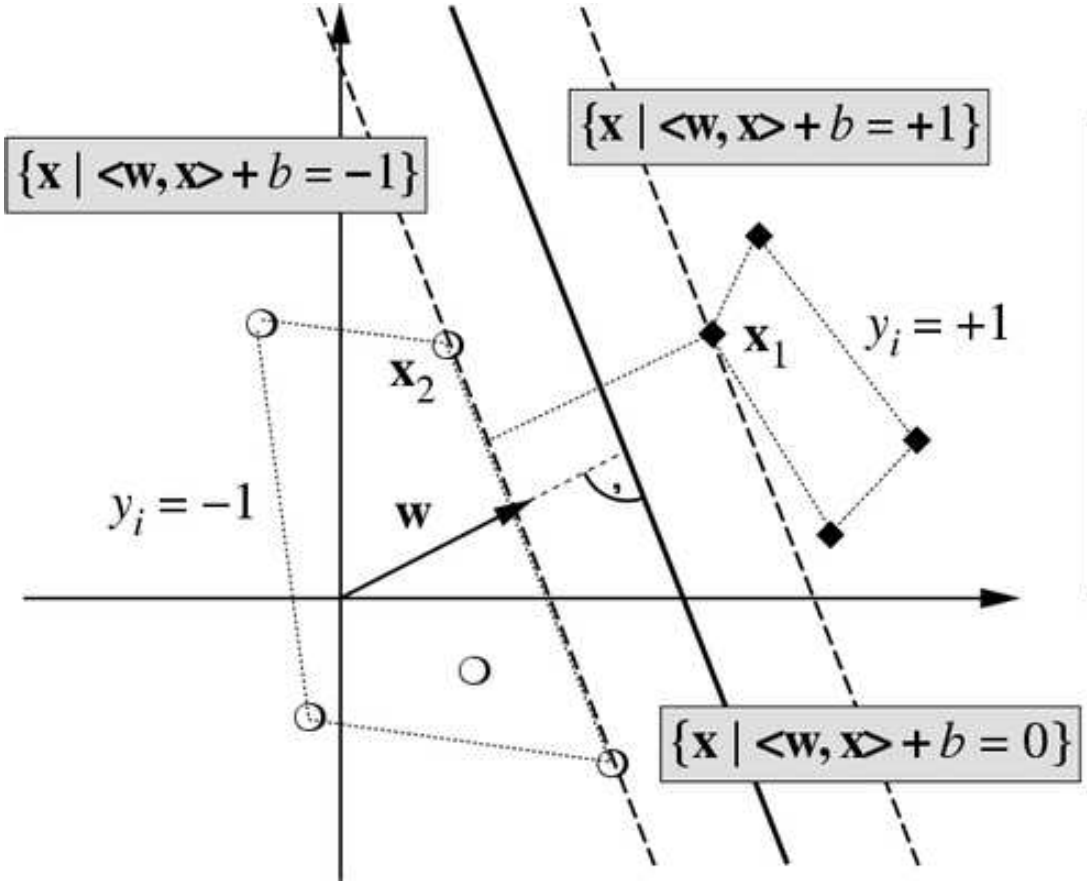


- each hyperplane

$$D(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b = c, \quad -1 < c < 1 \text{ is separating}$$

- optimal separating hyperplane - the one with the maximal margin

Classifier with maximal margin



Note:

$$\langle \mathbf{w}, \mathbf{x}_1 \rangle + b = +1$$
$$\langle \mathbf{w}, \mathbf{x}_2 \rangle + b = -1$$

$$\Rightarrow \quad \langle \mathbf{w}, (\mathbf{x}_1 - \mathbf{x}_2) \rangle = 2$$

$$\Rightarrow \left\langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, (\mathbf{x}_1 - \mathbf{x}_2) \right\rangle = \frac{2}{\|\mathbf{w}\|}$$

Classifier with maximal margin

$$\boxed{y(\mathbf{x}) = sgn(\langle \mathbf{w}, \mathbf{x} \rangle + b)}$$

where $\mathbf{w}$ and b are solution of

$$\min Q(\mathbf{w}), \qquad Q(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2$$

with respect to constraints

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1, \quad \text{for } i = 1, \ldots, M$$

- quadratic programming problem
- linear separability $\rightarrow$ solution exists
- no local minima

Classifier with maximal margin

- constrained optimization problem

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 \text{ subject to } y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1$$

- can be handled by introducing Lagrange multipliers $\alpha_i \geq 0$

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \alpha_i (y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1)$$

- minimize with respect to $\mathbf{w}$ and b
- maximize with respect to α_i

Classifier with maximal margin

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \alpha_i (y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1)$$

- minimize with respect to $\mathbf{w}$, b; maximize with respect to α
- Karush-Kuhn-Tucker (KKT) conditions

$$\frac{\delta L(\mathbf{w}, b, \alpha)}{\delta \mathbf{w}} = 0 \qquad \frac{\delta L(\mathbf{w}, b, \alpha)}{\delta b} = 0$$

- we get $\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i$ $\sum_{i=1}^{m} \alpha_i y_i = 0$

- $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) > 1 \rightarrow \alpha_i = 0$ $\mathbf{x}_i$ irrelevant
- $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) = 1 \rightarrow \alpha_i \neq 0$ $\mathbf{x}_i$ support vector

Dual problem

- by substitution to L we get

$$\max_{\alpha \in \mathbb{R}^n} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

subject to

$$\alpha_i \geq 0 \qquad \sum_{i=1}^{m} \alpha_i y_i = 0$$

- resulting classifier - (hard margin) support vector machine (SVM)

$$f(\mathbf{x}) = sgn(\sum_{i=1}^{m} y_i \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle + b),$$

$$b = y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle$$

Support Vector Machine

- work in feature space and use kernels
- classifier

$$f(\mathbf{x}) = sgn(\sum_{i=1}^{m} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b)$$

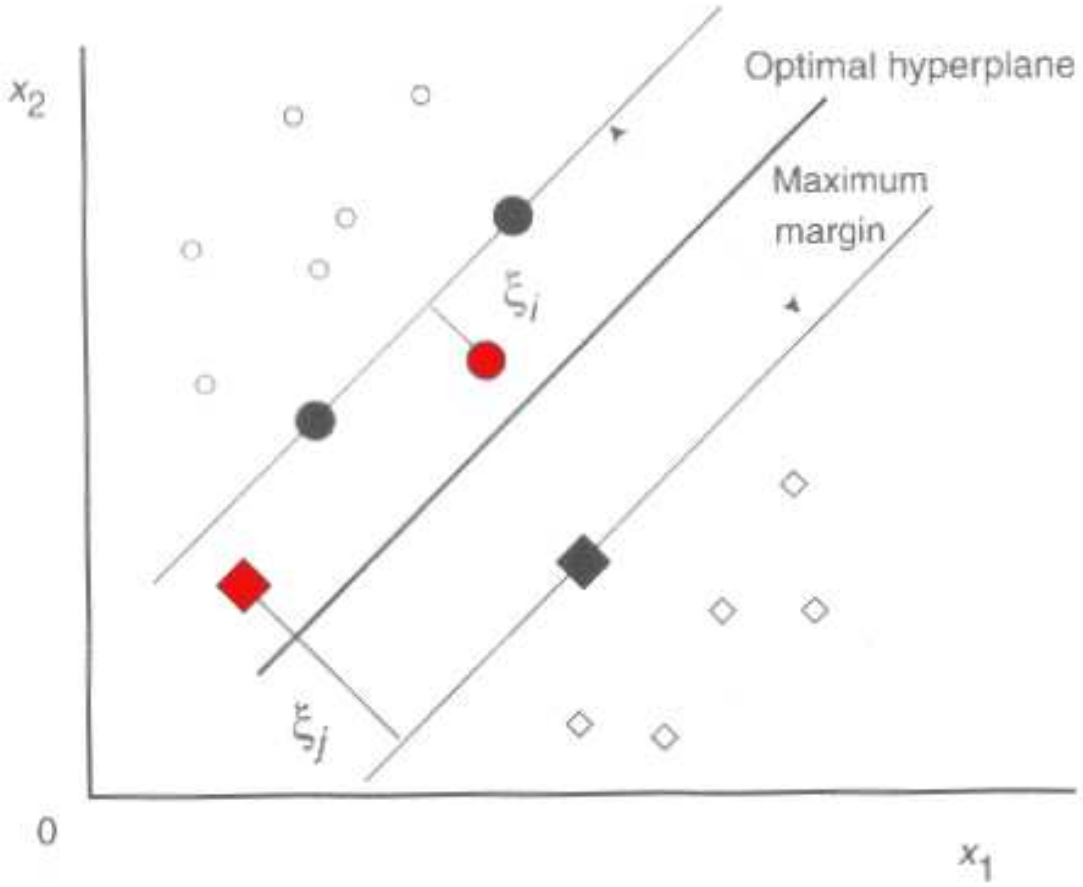
$$\max_{\alpha \in \mathbb{R}^n} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

subject to

$$\alpha_i \geq 0 \qquad \sum_{i=1}^{m} \alpha_i y_i = 0$$

Soft margin SVM

- separating hyperplane may not exists (high level of noise, overlap of classes, etc.)



Soft margin SVM

- separating hyperplane may not exists (high level of noise, overlap of classes, etc.)
- introduce slack variables ξ_i

$$y_i(\langle \mathbf{w}, \Phi(\mathbf{x})_i \rangle + b) \geq 1 - \xi_i$$

- minimize

$$Q(\mathbf{w}, \xi) = \frac{1}{2}||\mathbf{w}||^2 + C \sum_{i=1}^{m} \xi_i^2$$

- $C > 0$ trade-off between maximization of margin and minimization of training error (depands on noise level)

Soft margin SVM

- solution has a form

$$f(\mathbf{x}) = sgn(\sum_{i=1}^{m} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b)$$

$$\max_{\alpha \in \mathbb{R}^n} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \left(K(\mathbf{x}_i, \mathbf{x}_j) + \frac{\delta_{i,j}}{C} \right)$$

subject to

$$\alpha_i \geq 0 \qquad \sum_{i=1}^{m} \alpha_i y_i = 0$$

SVM - summary

- input points are mapped to the feature space
- dot product computed by means of kernel function
- classification via separating hyperplane with maximal margin
- such hyperplane is determined by support vectors
- other training samples are irrelevant
- data not separable in feature space (noise, etc.) - use soft margin
- control trade-of between maximal margin and minimum training error (C)

SVM vs. Neural Networks

+ maximization of generalization ability
+ no local minima

- extension to multiclass problems
- long training time
  - number of variables same as number of data points
  - not necessarily true - many techniques to reduce time exists
- selection of parameters
  - kernel function
  - C

References

- **Support Vector Machines for Pattern Classification**
  Shigoe Abe, Springer 2005
- **Learning with Kernels**
  Bernhard Schölkopf and Alex Smola
  MIT Press, Cambridge, MA, 2002
  Source of “sheep vectors” illustrations.
- **Learning kernel classifiers**
  Ralf Herbrich
  MIT Press, Cambridge, MA, 2002
- `http://www.kernel-machines.org/`

Software

- SVMlib (by Chih-Chung Chang and Chih-Jen Lin)
  `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`
- Matlab toolbox (by S. Gunn)
  `http://www.isis.ecs.soton.ac.uk/resources/svminfo/`

Questions?