Machine learning in computer vision

Lesson 7

Linear classifier

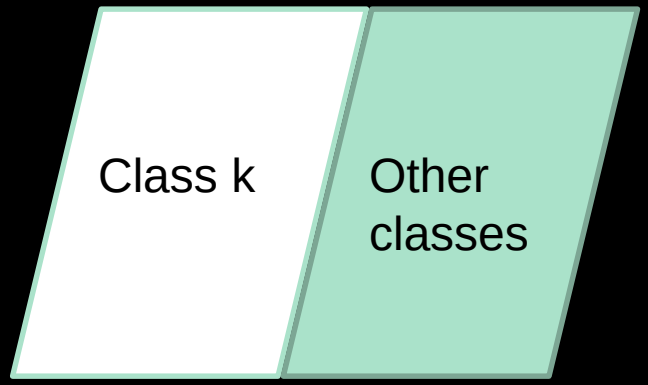
~~Assumption 1: there are two classes of data~~

Assumption 2: Classes are linearly separable, i.e. ∃ hyperplane that separates the space s.t. data from the two classes lie in different subspaces

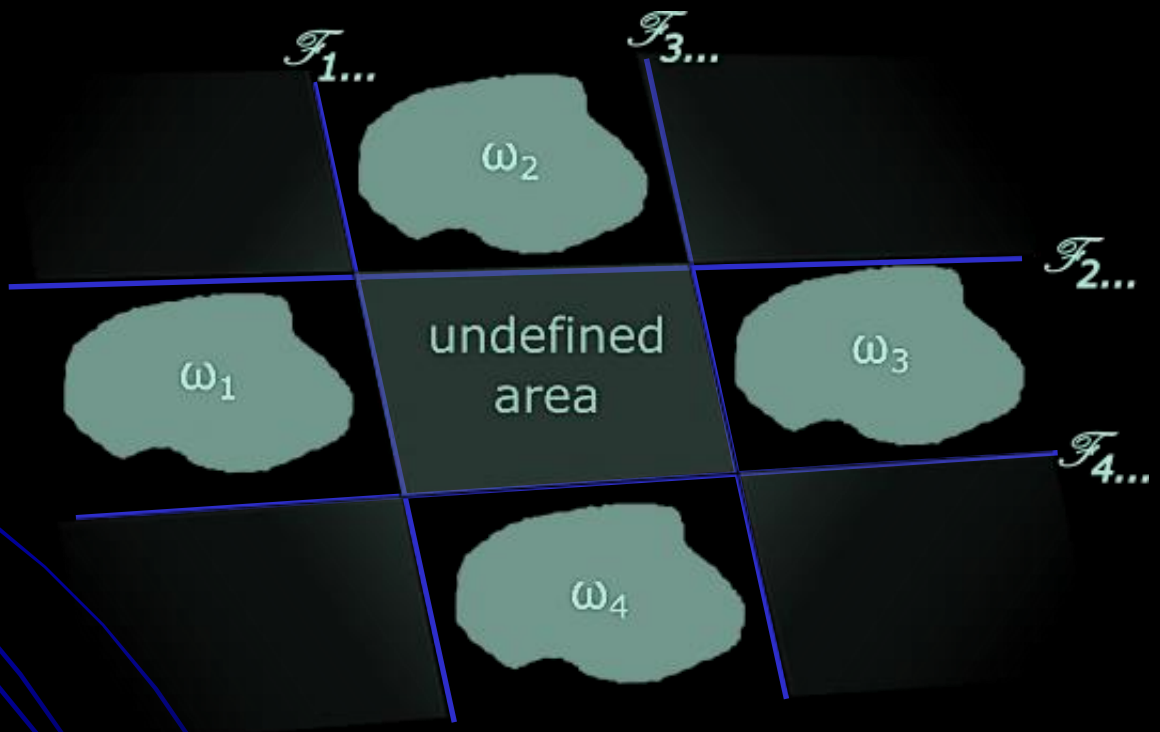
K classes (1)

For each class one decision function separating the class from the rest of the world

K decision functions



K classes (1)


$\mathcal{F}_{1...}$
$\mathcal{F}_{3...}$
$\mathcal{F}_{2...}$
$\mathcal{F}_{4...}$
ω_1
ω_2
ω_3
ω_4
undefined area

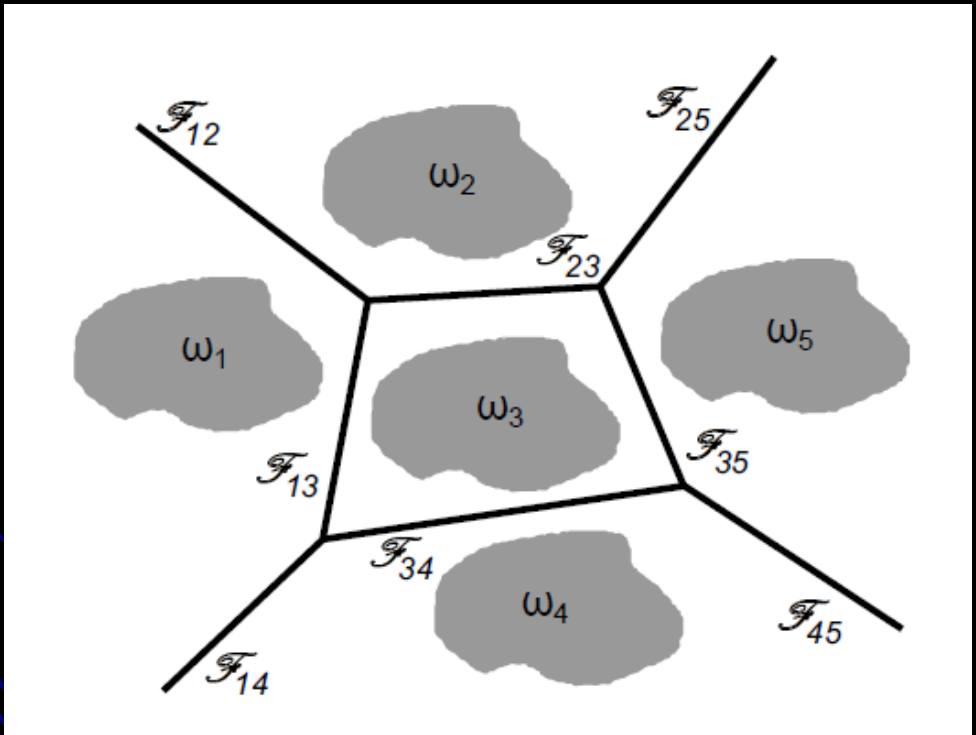
K classes (2)

For each couple of classes one decision function (not taking into account the rest of classes)

$K(K-1)/2$ max number of decision functions

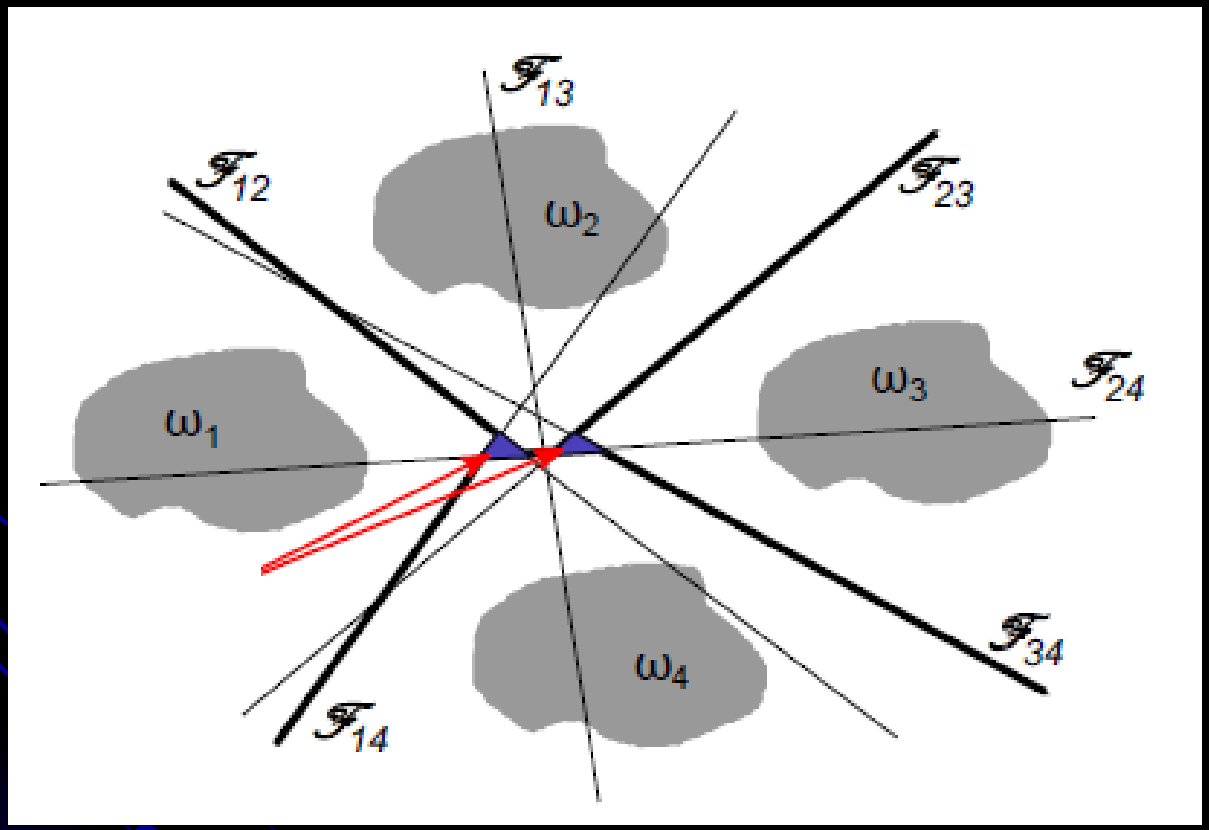
K classes (2)

$K(K-1)/2$ max number of decision functions



K classes (2)

Undefined areas



Linear machine

K decision functions (1 vs all)

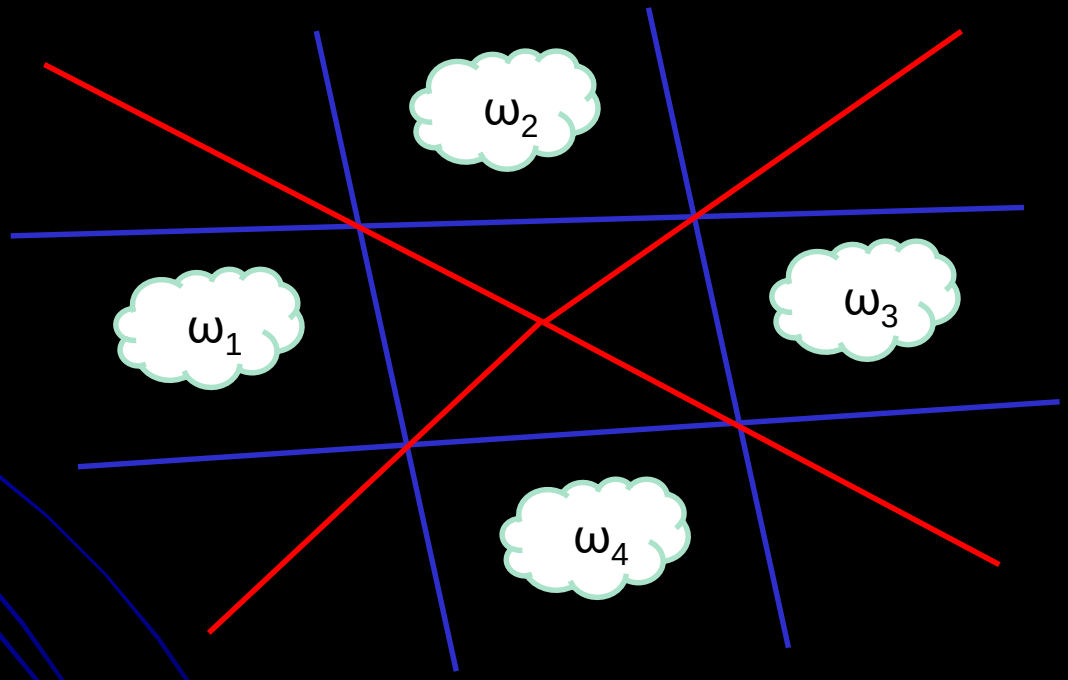
$\{f_k(\mathbf{x})\}_{k=1}^{K}$

x belongs to the class, where the function value is the highest

$$\mathbf{x} \in \omega_i \Leftrightarrow \forall_{j \neq i}: f_i(\mathbf{x}) > f_j(\mathbf{x})$$

K convex decision regions

Linear machine

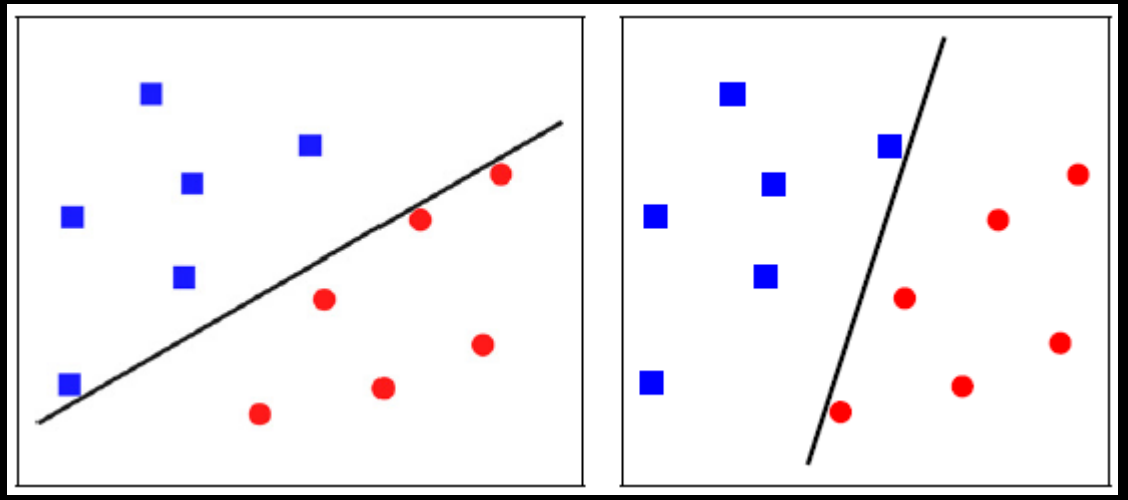


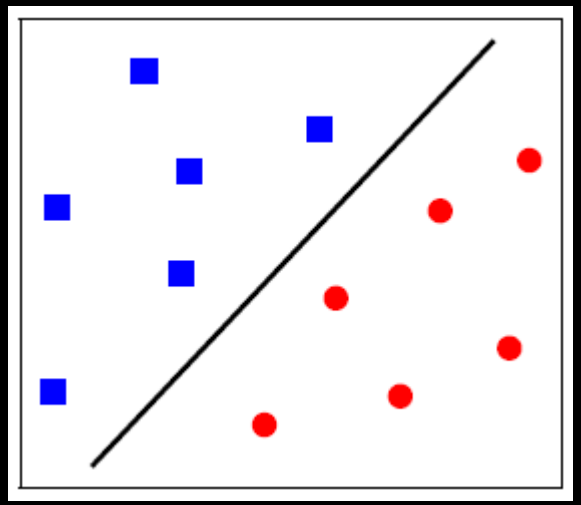
Linear classifier

Assumption 1: there are two classes of data

Assumption 2: Classes are linearly separable, i.e. $\exists$ hyperplane that separates the space s.t. data from the two classes lie in different subspaces

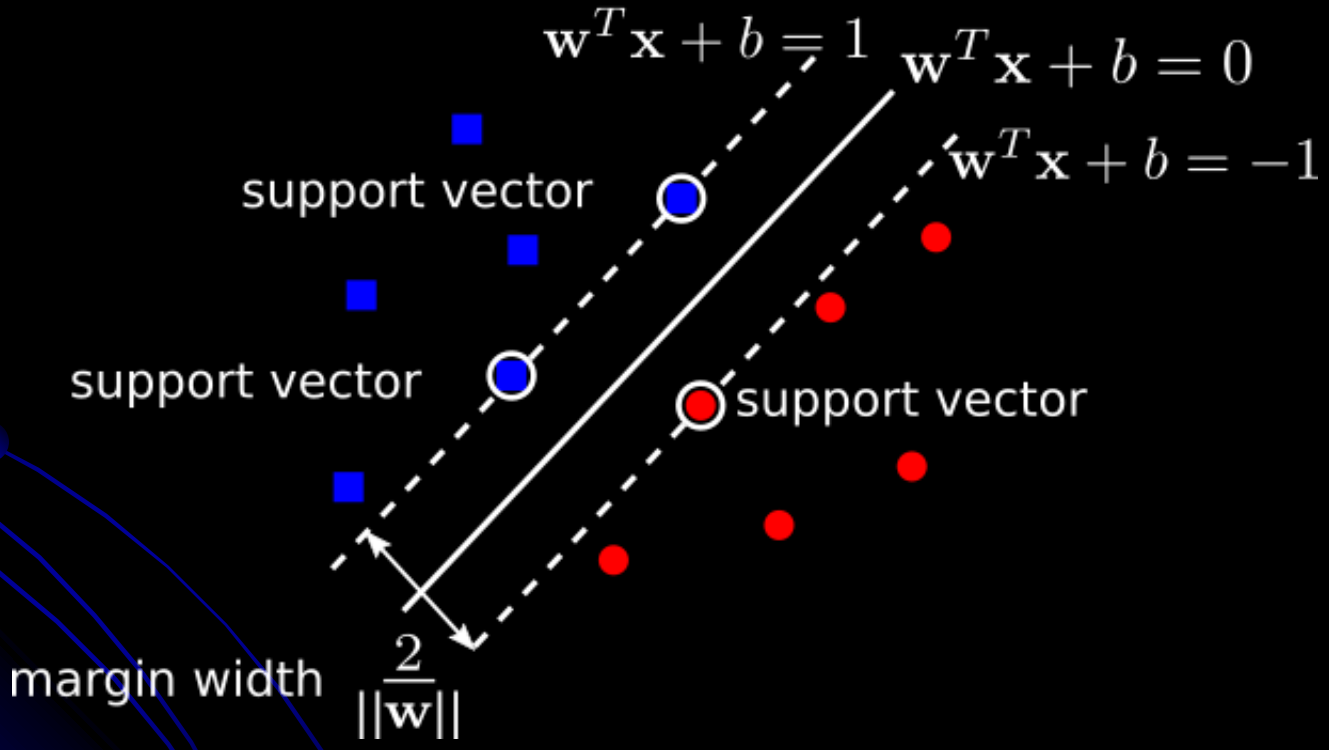
Hyperplane optimality





Support vector machines

SVM



SVM learning

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{2}{\|\mathbf{w}\|}$$

Change the problem:

$$(\mathbf{w}^*, b^*) = \arg\min_{(\mathbf{w},b)} \frac{1}{2}\|\mathbf{w}\|^2$$

s.t.

$\mathbf{w}^T\mathbf{x}_i + b \geq 1$ for $\mathbf{x}_i \in \omega_1$

$\mathbf{w}^T\mathbf{x}_i + b \leq -1$ for $\mathbf{x}_i \in \omega_2$

SVM learning

Change the conditions:

$$(\mathbf{w}^T\mathbf{x}_i + b)k_i \geq 1$$

where

$$k_i = \begin{cases} 1, & \mathbf{x}_i \in \omega_1 \\ -1, & \mathbf{x}_i \in \omega_2 \end{cases}$$

SVM learning

Use Lagrange method

$$L(\mathbf{w}, b, \alpha_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i \left((\mathbf{w}^T \mathbf{x_i} + b)k_i - 1\right)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i (\mathbf{w}^T \mathbf{x_i} + b)k_i + \sum_{i=1}^{N} \alpha_i$$

s.t.

$$\alpha_i \geq 0$$

SVM learning

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N} \alpha_i k_i \mathbf{x}_i \equiv 0 \quad \rightarrow \quad \mathbf{w} = \sum_{i=1}^{N} \alpha_i k_i \mathbf{x}_i$$

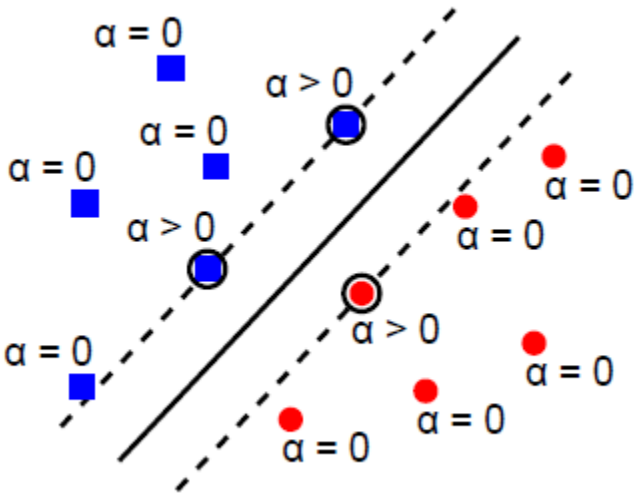
$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{N} \alpha_i k_i \equiv 0 \quad \rightarrow \quad \sum_{i=1}^{N} \alpha_i k_i = 0$$

$$(\mathbf{w}^T \mathbf{x}_i + b) k_i - 1 \geq 0$$

$$\alpha_i \geq 0$$

$$\alpha_i ((\mathbf{w}^T \mathbf{x}_i + b) k_i - 1) = 0$$

Kuhn Tucker conditions



$$\alpha_i = 0 \Leftrightarrow (\mathbf{w}^T \mathbf{x}_i + b) k_i > 1$$

$$\alpha_i > 0 \Leftrightarrow (\mathbf{w}^T \mathbf{x}_i + b) k_i = 1$$

SVM learning

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i k_i \mathbf{x}_i$$

$$\sum_{i=1}^{N} \alpha_i k_i = 0$$

Use results to get dual problem

$$L(\mathbf{w}, b, \alpha_i) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i (\mathbf{w}^T \mathbf{x}_i + b) k_i + \sum_{i=1}^{N} \alpha_i$$

$$\tilde{\alpha}_i = \arg\max \left(\sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j k_i k_j \mathbf{x}_i^T \mathbf{x}_j \right)$$

s.t.

$\alpha_i \geq 0$

$\sum_{i=1}^{N} \alpha_i k_i = 0$

SVM learning

1. Solve the dual problem

   α_i

2. Find the support vectors SV

   $\mathbf{x_i} \in SV \Leftrightarrow \alpha_i > 0$

3. Find b

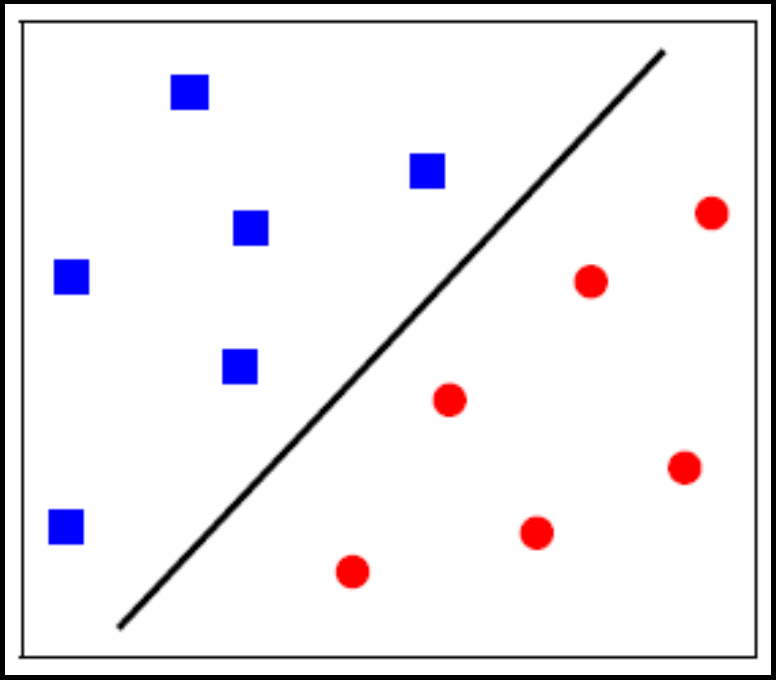
   $b = \frac{1}{|SV|} \sum_{\mathbf{x_j} \in SV} \left(k_j - \sum_{\mathbf{x_i} \in SV} \alpha_i k_i \mathbf{x_i}^T \mathbf{x_j} \right)$

SVM classification

Find the value

$$f(\mathbf{x}) = \sum_{\mathbf{x}_i \in SV} \alpha_i k_i \mathbf{x}_i{}^T \mathbf{x} + b$$

Hyperplane



Linear classifier

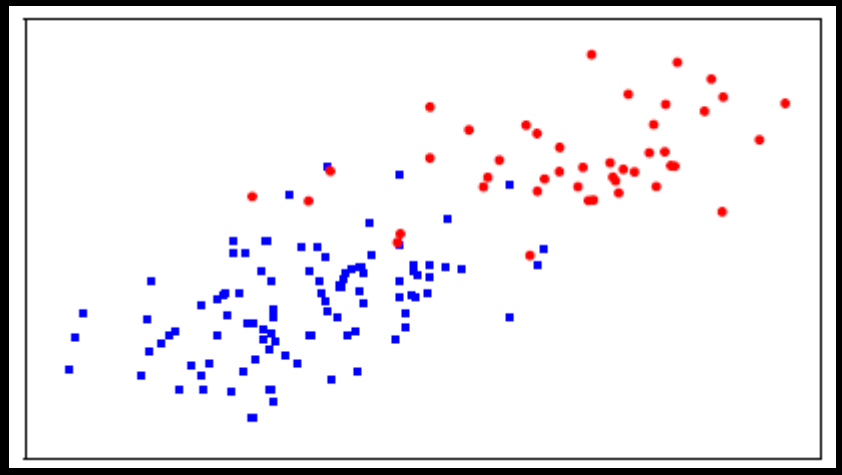
Assumption 1: there are two classes of data

~~Assumption 2: Classes are linearly separable, i.e. ∃ hyperplane that separates the space s.t. data from the two classes lie in different subspaces~~

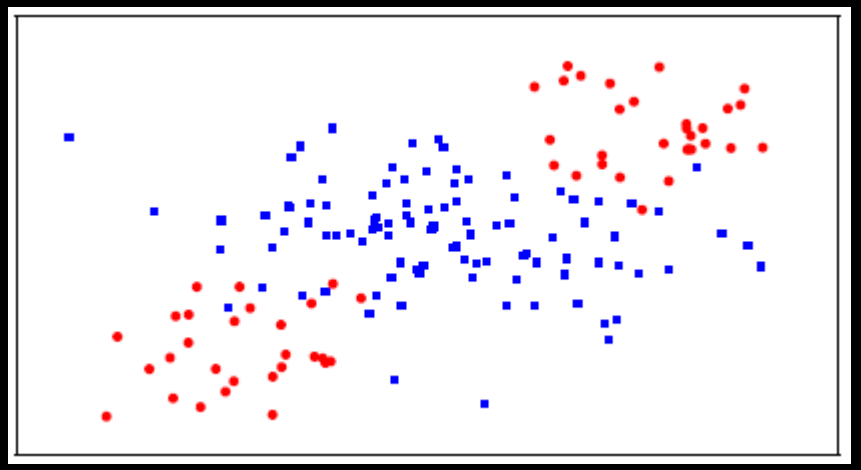
Linearly nonseparable data

2 types:

Nearly separable

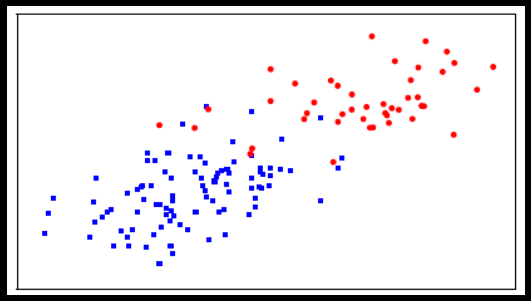


Nonseparable



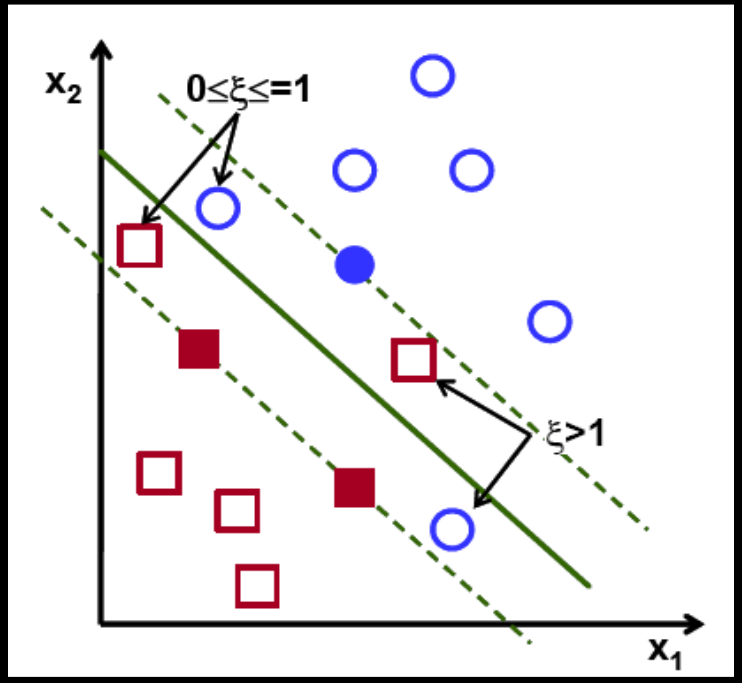
Soft margin SVM

Nearly separable data



Allow small errors

Introduce slack variables ξ_i



Soft margin SVM

C – penalty for misclassification, affects the size of the margin

$$(\mathbf{w}^*, b^*, \xi_i^*) = \arg\min\left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i\right)$$

s.t.

$$(\mathbf{w}^T\mathbf{x}_i + b)k_i \geq 1 - \xi_i$$

Soft margin SVM

Derive dual problem:

$$\tilde{\alpha}_i = \arg\max\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k_i k_j \mathbf{x_i}^T\mathbf{x_j}\right)$$

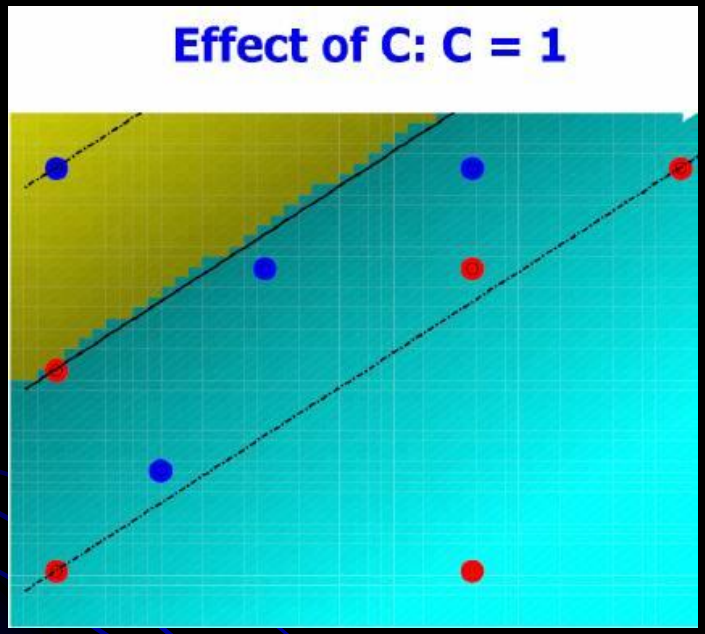
s.t.

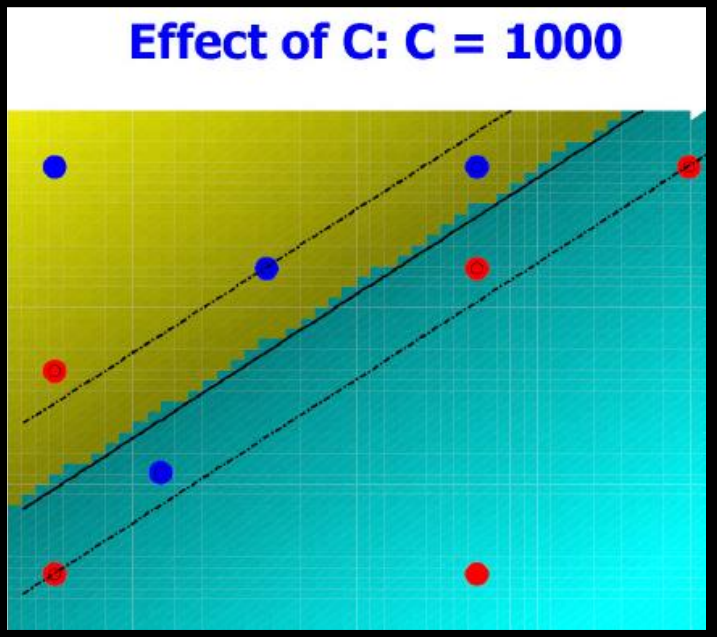
$$C \geq \alpha_i \geq 0$$

$$\sum_{i=1}^{N}\alpha_i k_i = 0$$

Classify as before

Soft margin SVM

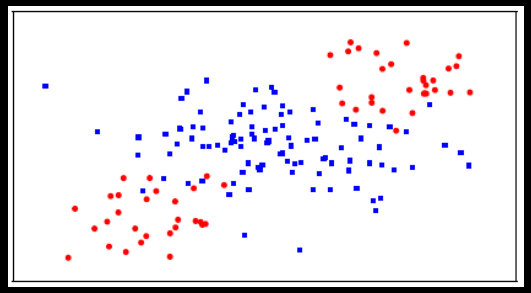




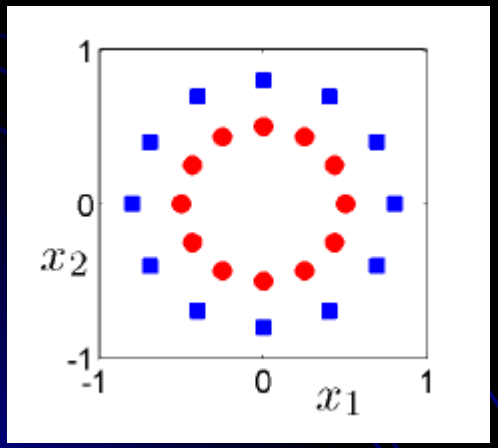
Use cross validation to find optimal value of C

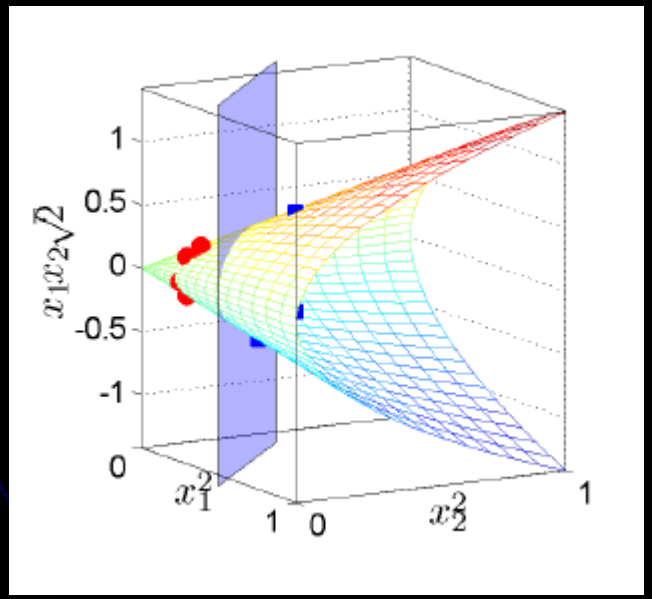
Nonlinear SVM

Linearly nonseparable data



Transform to higher dimensions, where they are separable





$$\phi : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}x_1 x_2 \end{pmatrix}$$

Nonlinear SVM

Recall the separable case

$$\tilde{\alpha}_i = \arg\max\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k_i k_j \mathbf{x_i}^T\mathbf{x_j}\right)$$

$$b = \frac{1}{|SV|}\sum_{\mathbf{x_j}\in SV}\left(k_j - \sum_{\mathbf{x_i}\in SV}\alpha_i k_i \mathbf{x_i}^T\mathbf{x_j}\right)$$

$$f(\mathbf{x}) = \sum_{\mathbf{x_i}\in SV}\alpha_i k_i \mathbf{x_i}^T\mathbf{x} + b$$

In nonseparable case we work with transformed data $\varphi(\mathbf{x_i})$

Nonlinear SVM learning

1. Solve the dual problem

$$\tilde{\alpha}_i = \arg\max\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k_i k_j \varphi(\mathbf{x_i})^T\varphi(\mathbf{x_j})\right)$$

2. Find the support vectors SV: $\mathbf{x_i} \in SV \Leftrightarrow \alpha_i > 0$

3. Find b

$$b = \frac{1}{|SV|}\sum_{\mathbf{x_j}\in SV}\left(k_j - \sum_{\mathbf{x_i}\in SV}\alpha_i k_i \varphi(\mathbf{x_i})^T\varphi(\mathbf{x_j})\right)$$

4. Find the value

$$f(\mathbf{x}) = \sum_{\mathbf{x_i}\in SV}\alpha_i k_i \varphi(\mathbf{x_i})^T\varphi(\mathbf{x}) + b$$

Nonlinear SVM

How to find $\varphi(\mathbf{x}_i)$?

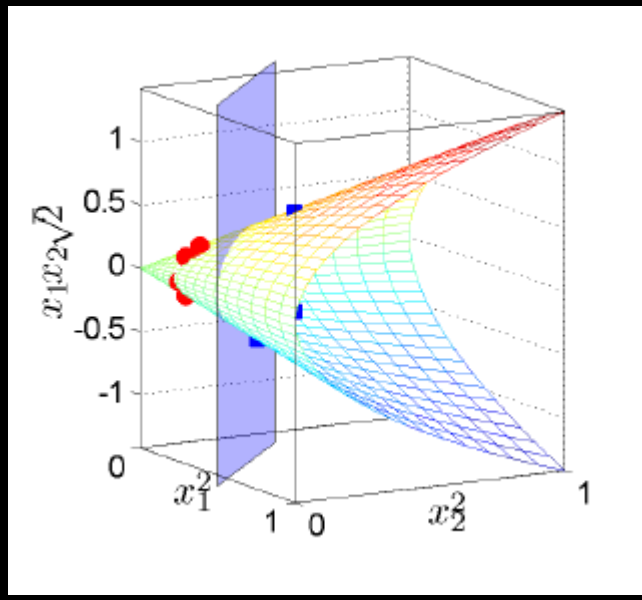
We don't have to.

$$K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)$$

K – kernel (dot product of transformed data)

Kernel trick

$$\varphi: \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}x_1x_2 \end{pmatrix}$$



$$K(\mathbf{x}, \mathbf{y}) = (x_1^2, x_2^2, \sqrt{2}x_1x_2) \begin{pmatrix} y_1^2 \\ y_2^2 \\ \sqrt{2}y_1y_2 \end{pmatrix}$$

$$= x_1^2y_1^2 + x_2^2y_2^2 + 2x_1y_1x_2y_2$$
$$= (x_1y_1 + x_2y_2)^2$$
$$= (\mathbf{x}^T\mathbf{y})^2$$

Kernel example

$$K(\mathbf{x},\mathbf{y}) = (\mathbf{x}^T\mathbf{y})^2 = x_1^2y_1^2 + x_2^2y_2^2 + 2x_1y_1x_2y_2$$

$$\varphi(\mathbf{x}) \rightarrow \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix}$$

$$\varphi(\mathbf{x}) \rightarrow \begin{pmatrix} x_1^2 \\ x_1x_2 \\ x_1x_2 \\ x_2^2 \end{pmatrix}$$

$$\varphi(\mathbf{x}) \rightarrow \frac{1}{\sqrt{2}}\begin{pmatrix} x_1^2 - x_2^2 \\ 2x_1x_2 \\ x_1^2 + x_2^2 \end{pmatrix}$$

Mercer's Theorem

A symmetric function $K(\mathbf{x}, \mathbf{y})$ can be expressed as a dot product

$K(\mathbf{x}, \mathbf{y}) = \varphi(\mathbf{x})^T \varphi(\mathbf{y})$

for some φ iff

$K(\mathbf{x}, \mathbf{y})$ is positive semidefinite (psd).

How to check psd?

$K(\mathbf{x}, \mathbf{y})$ is psd, when matrix

$$\begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & \cdots & K(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_n, \mathbf{x}_1) & \cdots & K(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix}$$

is psd for any collection $\{\mathbf{x}_1, \ldots \mathbf{x}_n\}$

Matrix $\mathbf{M}_{[n \times n]}$ is psd iff $\forall_{\mathbf{a} \in \mathbb{R}^n}: \mathbf{a}^T \mathbf{M} \mathbf{a} \geq 0$

Kernel properties

If K_1 and K_2 are kernels, so are

$K(\mathbf{x}, \mathbf{y}) = K_1(\mathbf{x}, \mathbf{y}) + K_2(\mathbf{x}, \mathbf{y})$

$K(\mathbf{x}, \mathbf{y}) = aK_1(\mathbf{x}, \mathbf{y}), \mathrm{a} > 0$

$K(\mathbf{x}, \mathbf{y}) = K_1(\mathbf{x}, \mathbf{y})K_2(\mathbf{x}, \mathbf{y})$

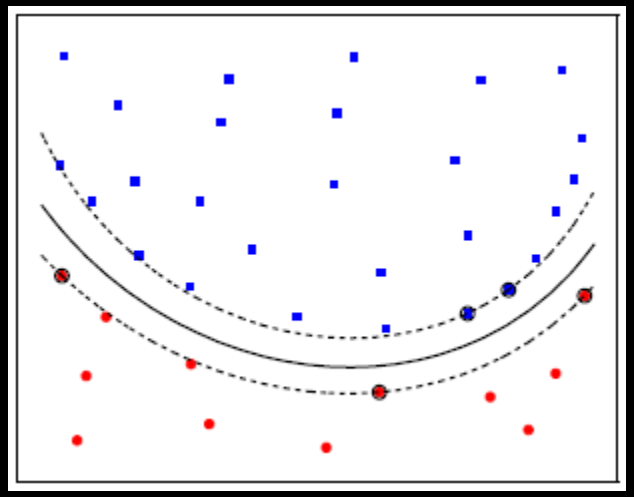
$K(\mathbf{x}, \mathbf{y}) = p\big(K_1(\mathbf{x}, \mathbf{y})\big), p$ polynomial with non-negative coefficients

$K(\mathbf{x}, \mathbf{y}) = \exp(K_1(\mathbf{x}, \mathbf{y}))$

Common kernels

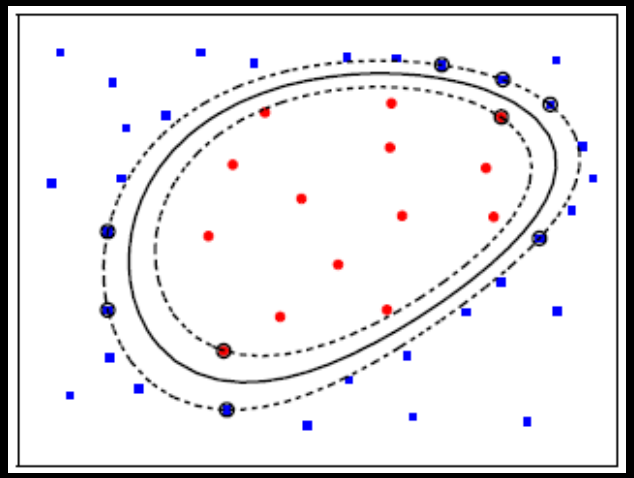
Polynomial

$$K(\mathbf{x}, \mathbf{y}) = (a + \mathbf{x}^T \mathbf{y})^p$$



Gaussian

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



Kernel SVM

1. Solve the dual problem

$$\tilde{\alpha}_i = \arg\max\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k_i k_j K\left(\mathbf{x}_i, \mathbf{x}_j\right)\right)$$

2. Find the support vectors SV: $\mathbf{x}_i \in SV \Leftrightarrow \alpha_i > 0$

3. Find b

$$b = \frac{1}{|SV|}\sum_{\mathbf{x}_j \in SV}\left(k_j - \sum_{\mathbf{x}_i \in SV}\alpha_i k_i K\left(\mathbf{x}_i, \mathbf{x}_j\right)\right)$$

4. Find the value

$$f(\mathbf{x}) = \sum_{\mathbf{x}_i \in SV}\alpha_i k_i K\left(\mathbf{x}_i, \mathbf{x}\right) + b$$

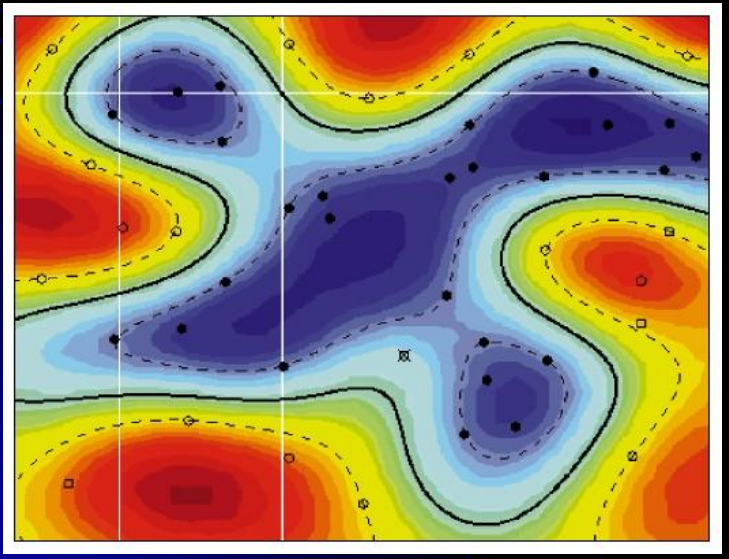
A Tutorial on Support Vector Machines for Pattern Recognition

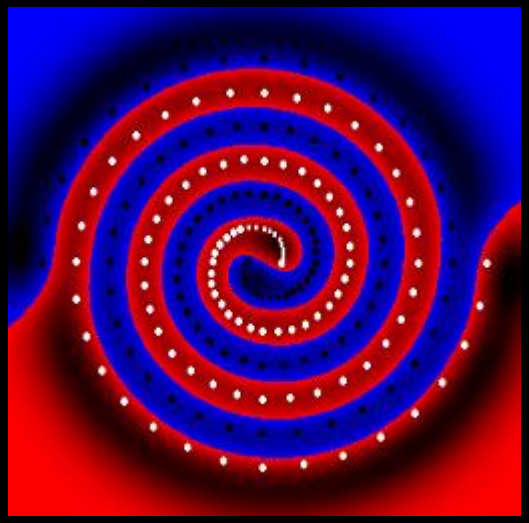
CHRISTOPHER J.C. BURGES burges@lucent.com
Bell Laboratories, Lucent Technologies

IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 12, NO. 2, MARCH 2001

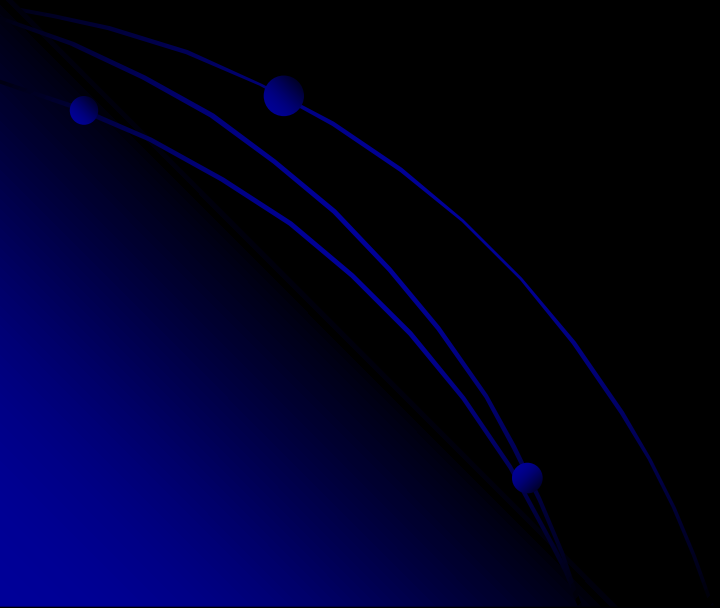
An Introduction to Kernel-Based Learning Algorithms

Klaus-Robert Müller, Sebastian Mika, Gunnar Rätsch, Koji Tsuda, and Bernhard Schölkopf





Ensemble methods



Classifiers

Performance

None of the classifiers is perfect

Complementary

Examples which are not correctly classified by one classifier may be correctly classified by the other classifiers

Idea

Combine the classifiers to improve the performance

Slides adapted from Evgueni Smirnov

Ensemble methods

Independently Constructed Ensembles

- the base classifiers are generated in parallel
- exploit the independence between them

Coordinated Construction of Ensembles

- the base classifiers are generated sequentially
- exploit the dependence between them

Various names: ensemble methods, committee, classifier fusion, combination, aggregation,…

Base classifiers

Homogeneous classifiers – use of the same algorithm over diversified data sets

Heterogeneous classifiers – different learning algorithms over the same data

Ensembles

Dietterich(2002) showed that ensembles overcome three problems:

Statistical

Computational

Representational

Ensembles

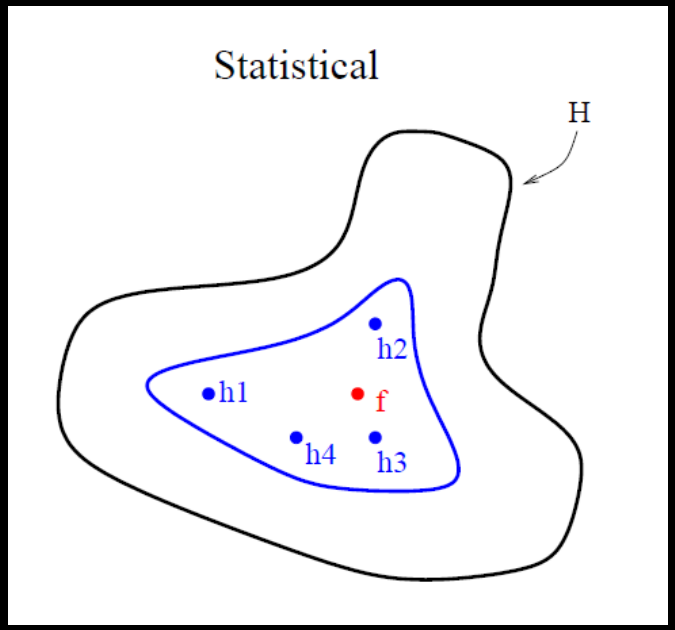
The Statistical Problem:

- The hypothesis space is too large for the amount of available data
- There are many hypotheses with the same accuracy on the data and the learning algorithm chooses only one of them
- There is a risk that the accuracy of the chosen hypothesis is low on unseen data

Ensembles

The Statistical Problem:

By constructing an ensemble out of all of these accurate classifiers, the algorithm can “average" their votes and reduce the risk of choosing the wrong classifier.

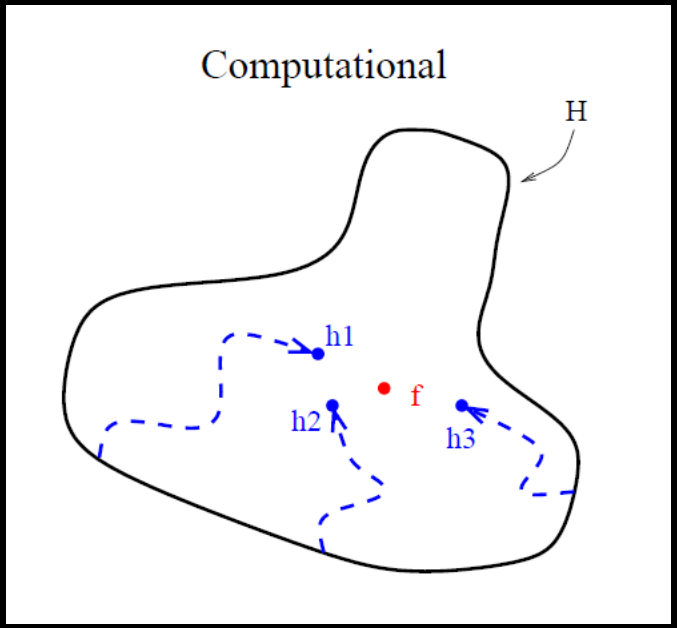


Ensembles

The Computational Problem:

- The learning algorithm cannot guarantee finding the best hypothesis

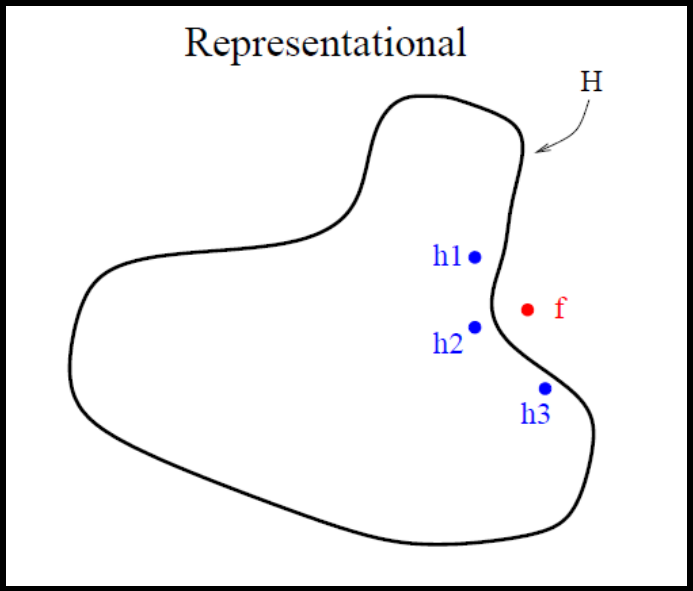
An ensemble constructed by running the local search from many different starting points may provide a better approximation to the true unknown function



Ensembles

The Representational Problem:

The hypothesis space does not contain any good approximation of the target class(es)



Ensembles

- The statistical problem and computational problem result in the variance component of the error of the classifiers
- The representational problem results in the bias component of the error of the classifiers
- Hence, ensemble methods can reduce both the bias and the variance of learning algorithms

Combining classifiers

Majority voting:

Every base classifier makes a prediction (votes) for each test instance and the final output prediction is the one that receives more than half of the votes

C1	1	1	0	0	0	1	0	0	0
C2	0	0	1	1	1	1	0	1	1
C3	1	0	0	1	1	0	0	1	1
C*	1	0	0	1	1	1	0	1	1

Voting

Modifications:

Plurality voting

- the most voted prediction (even if that is less than half of the votes) as the final prediction

Weighted Voting

- increase the importance of one or more models
- In weighted voting you count the prediction of the better models multiple times

Why Majority Voting works?

Suppose there are 25 base classifiers

Each classifier has error rate, $\varepsilon = 0.35$

Assume errors made by the classifiers are uncorrelated

Probability that the ensemble classifier makes a wrong prediction:

$$P(X \geq 13) = \sum_{i=13}^{25} \binom{25}{i} \varepsilon^i (1-\varepsilon)^{25-i} = 0.06$$