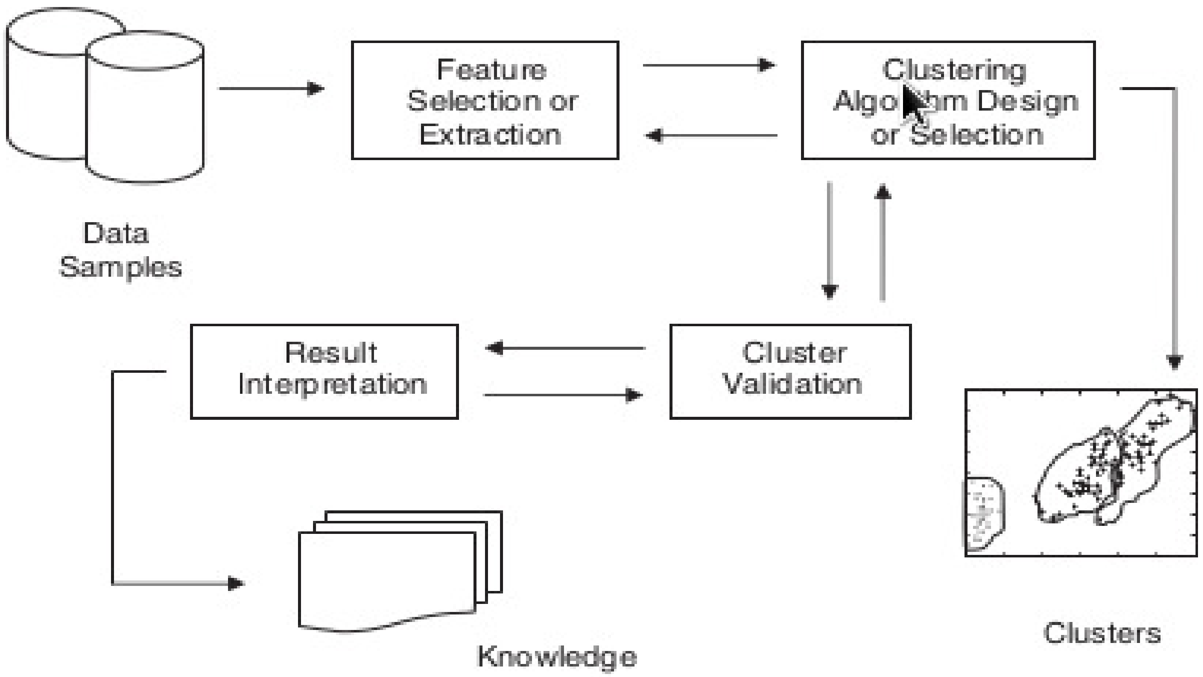
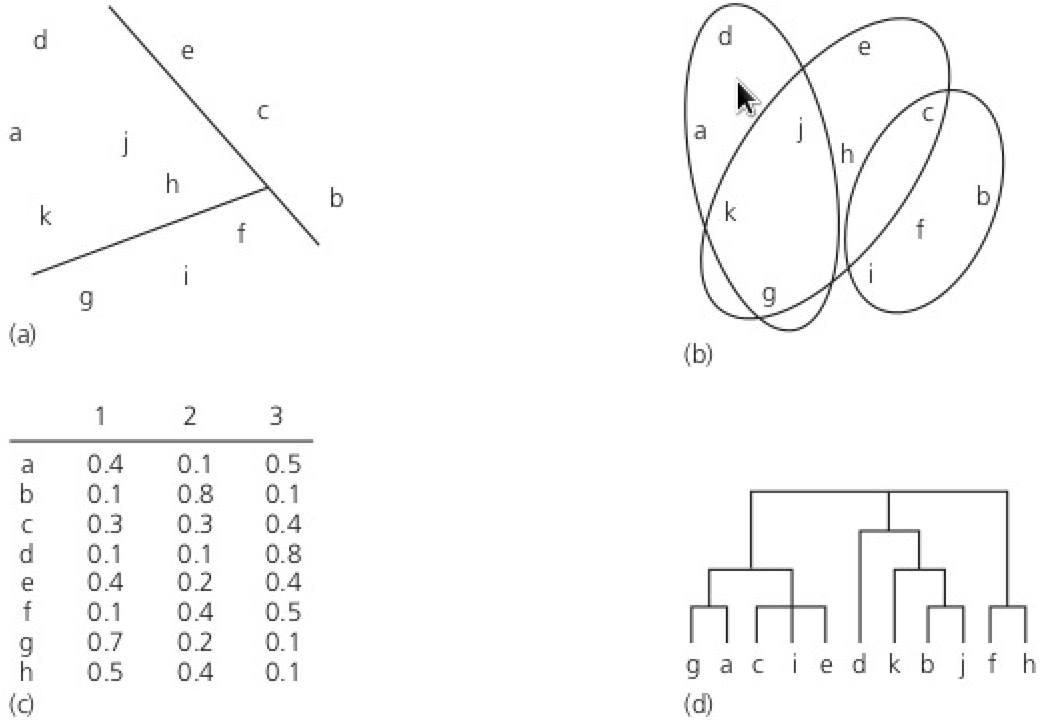
Data Mining

Mining continous and sequential data


Feature Selection or Extraction
Clustering Algorithm Design or Selection
Data Samples
Result Interpretation
Cluster Validation
Knowledge
Clusters

Clustering procedure. The basic process of cluster analysis consists of four steps with a feedback pathway. These steps are closely related to each other and determine the derived clusters.

Clusters



	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8
e	0.4	0.2	0.4
f	0.1	0.4	0.5
g	0.7	0.2	0.1
h	0.5	0.4	0.1

(c)

Different ways of representing clusters.

Proximity measures and their applications.

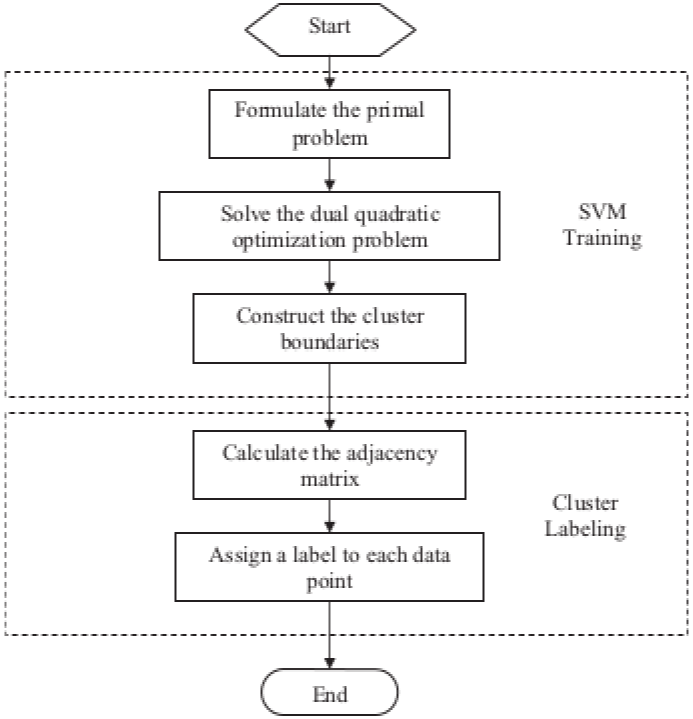
Measure	Metric	Examples and Applications
Minkowski distance	Yes	Fuzzy *c*-means with measures based on Minkowski family (Hathaway et al., 2000)
City block distance	Yes	Fuzzy ART (Carpenter et al., 1991)
Euclidean distance	Yes	*K*-means with its variants (Ball and Hall, 1967; Forgy, 1965; MacQueen, 1967)
Sup distance	Yes	Fuzzy *c*-means with sup norm (Bobrowski and Bezdek, 1991)
Mahalanobis	Yes	Ellipsoidal ART (Anagnostopoulos and M. Georgiopoulos, 2001); Hyperellipsoidal clustering algorithm (Mao and Jain, 1996)
Point symmetry distance	No	Symmetry-based *K*-means (Su and Chou, 2001)
Pearson correlation	No	Widely used as the measure for microarray gene expression data analysis (Eisen et al., 1998)
Cosine similarity	No	The most commonly used measure in document clustering (Steinbach et al., 2000)

KERNEL-BASED CLUSTERING

- Since the 1990s, kernel-based learning algorithms have become increasingly important in pattern recognition and machine learning, particularly in supervised classification and regression analysis, with the introduction of support vector machines

- SVM – Support Vector Machine
- Kernel k-means
- NMF based clustering
- …

SUPPORT VECTOR CLUSTERING


Start
Formulate the primal problem
Solve the dual quadratic optimization problem
Construct the cluster boundaries
SVM Training
Calculate the adjacency matrix
Assign a label to each data point
Cluster Labeling
End

Flowchart of SVC algorithm. The SVC algorithm consists of two main phases: SVM training for generating the cluster boundaries and cluster labeling for determining the cluster membership of each data point.

SVM

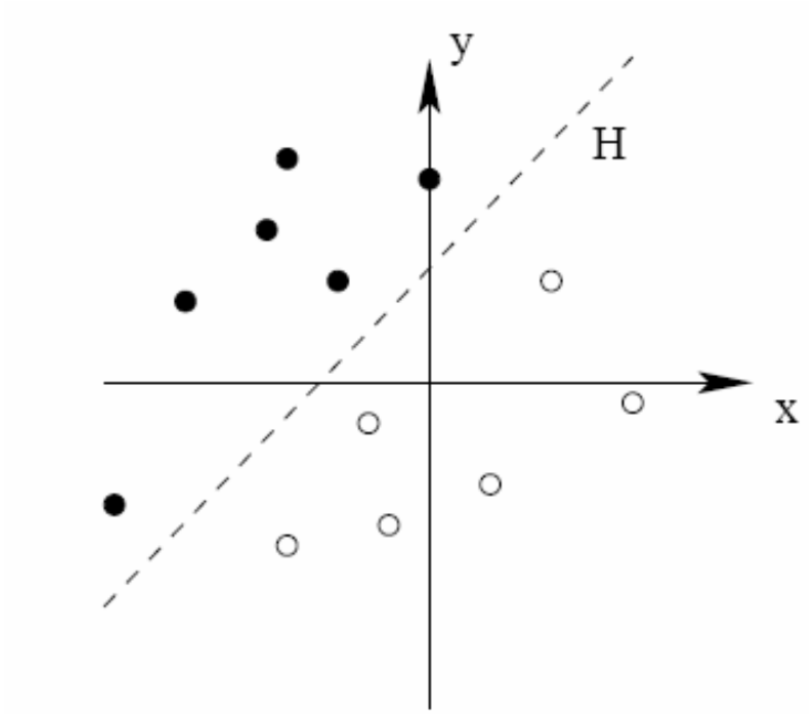
SVM : Suport Vector Machine

SVM is a binary classification supervised learning method introduced by Vladimir Vapnik in 1995.

It is based on the use of kernel functions (kernel) which allows an optimal separation of data.

SVM : principle (1)

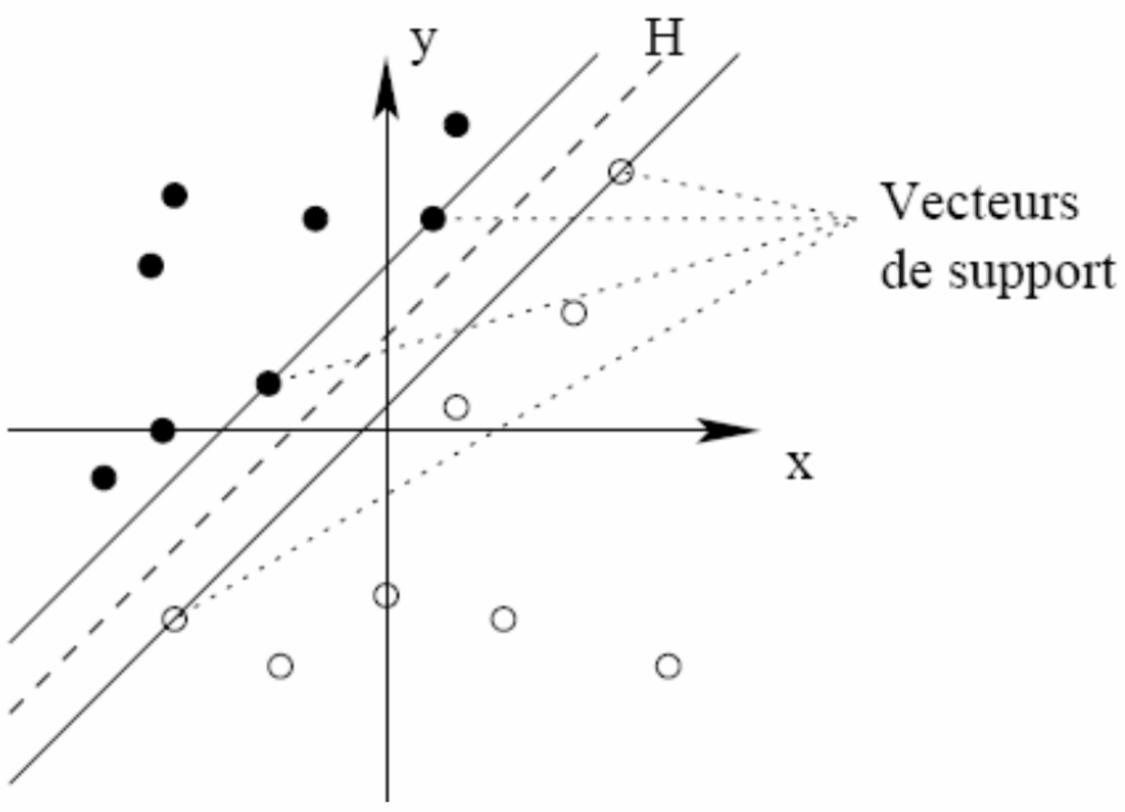
SVM became famous when, using images as input, it gave accuracy comparable to neural-network with hand-designed features in a handwriting recognition task. Currently, SVM is widely used in object detection & recognition, content-based image retrieval, text recognition, biometrics, speech recognition, etc.



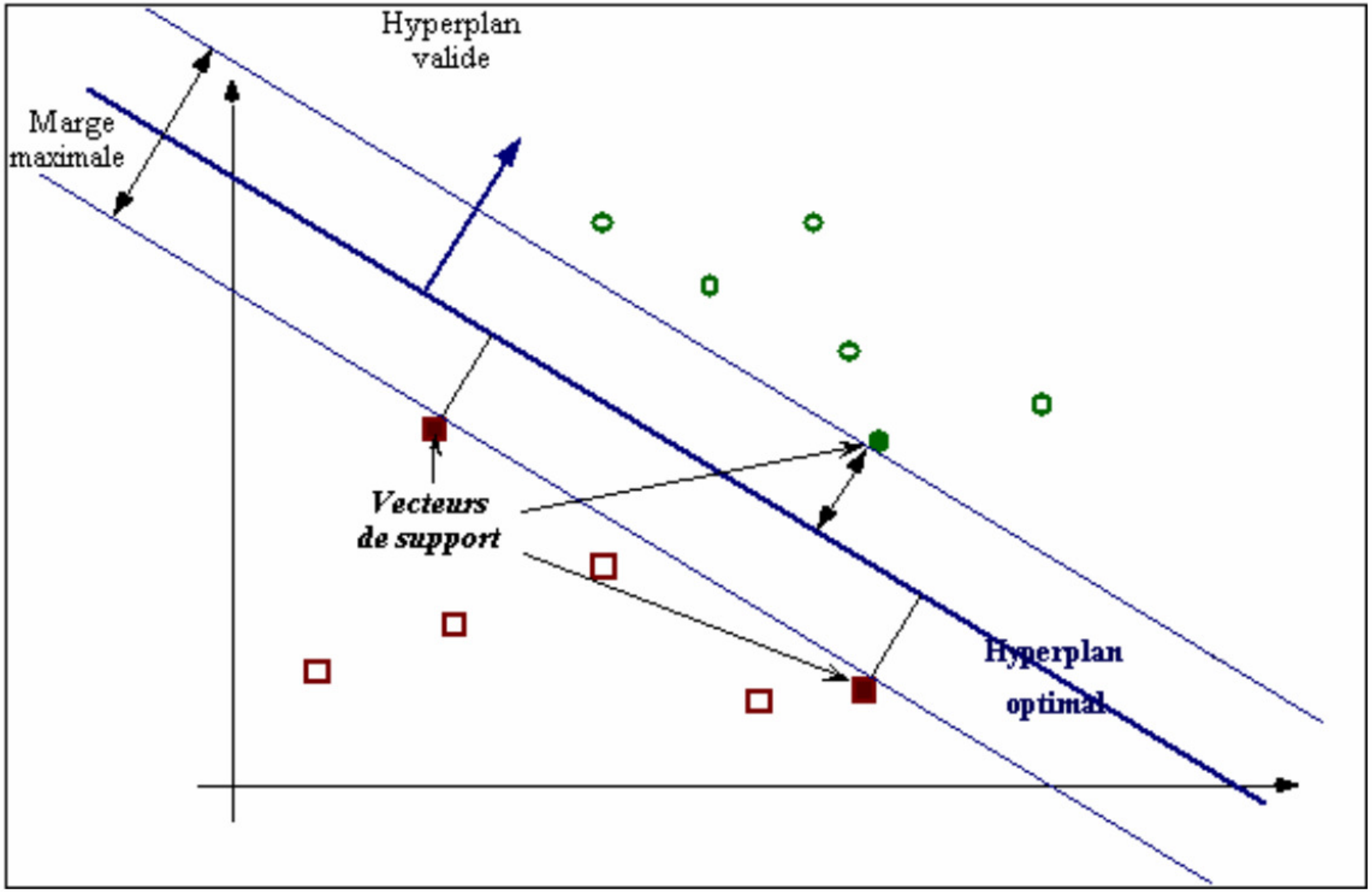
Examples closest to the hyperplane are ***support vectors***.

SVM : principle (2)

Find a hyperplane whose minimum distance to the learning examples is maximum (distance "margin").



SVM : principle (3)



SVM : Hyperplanes

- Classification task
  - Linear separation case

- We seek *h* using a linear function:

$$h(x) = w.x + b$$

- The *separation surface* is the hyperplane :

$$w.\boldsymbol{x} + b = 0$$

- It is valid if $\forall i \quad u_i \, h(\boldsymbol{x}_i) \geq 0$

- The hyperplane is has the canonical form when $\min_i |w.\boldsymbol{x} + b| = 1$

or $\forall i \quad u_i\,(w.\boldsymbol{x}_i + b) \geq 1$

Margin optimization

The distance from a point to the hyperplan is : $d(\boldsymbol{x}) = \dfrac{|\boldsymbol{w}.\boldsymbol{x} + w_0|}{\|\boldsymbol{w}\|}$

The optimal hyperplane is the one for which the distance to the closest points (margin) is maximized. This distance is $\dfrac{2}{\|\boldsymbol{w}\|}$

Maximizing the margin is therefore to minimize ||w|| under the constraints:

$$\begin{cases} \min \frac{1}{2} \|\boldsymbol{w}\|^2 \\ \forall \boldsymbol{i} \quad \boldsymbol{u_i}(\boldsymbol{w}.\boldsymbol{x_i} + w_0) \geq 1 \end{cases}$$

Optimization problem: solution

$$\begin{cases} D(\boldsymbol{x}) & = & (\boldsymbol{w}^{*} .\boldsymbol{x} + w_0^{*}) \\ \boldsymbol{w}^{*} & = & \sum_{i=1}^{m} \alpha_i^{*} u_i \boldsymbol{x}_i \\ w_0^{*} & = & u_s - \sum_{i=1}^{m} \alpha_i^{*} u_i (\boldsymbol{x}_i .\boldsymbol{x}_s) \end{cases}$$

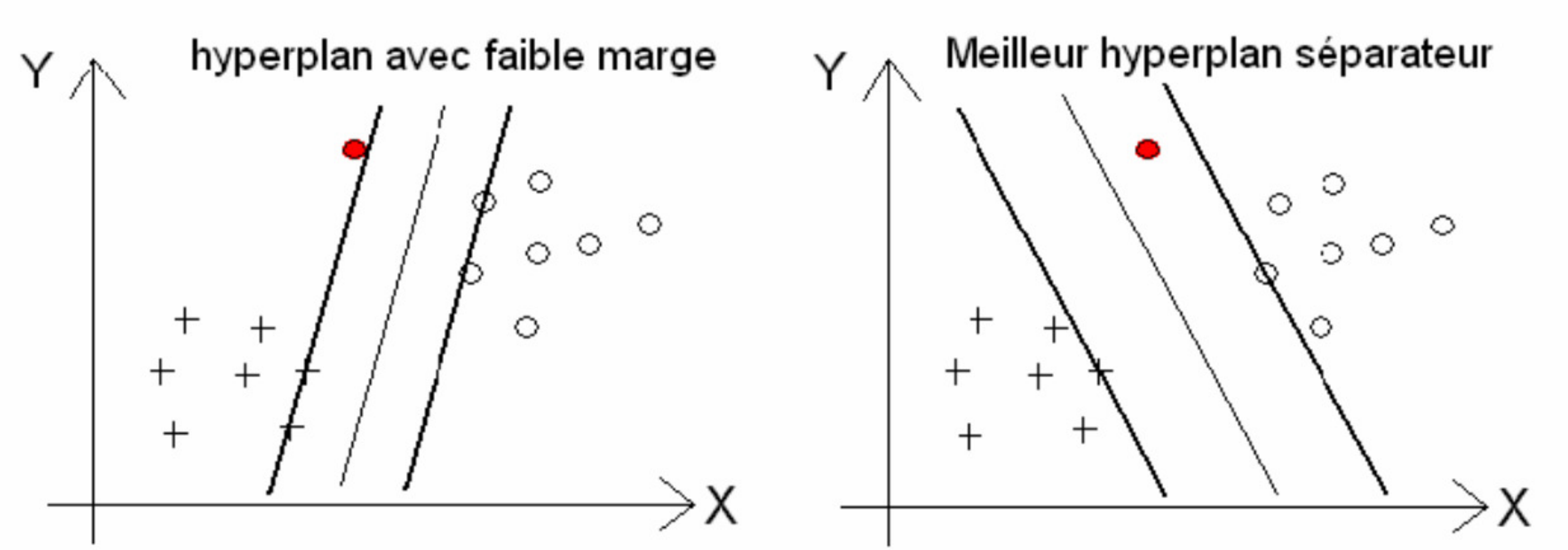
* : estimated

(x_S, u_S) a point of the suport

Property1 : only α_i corresponding to the closest points are non-zero. We speak about support points.

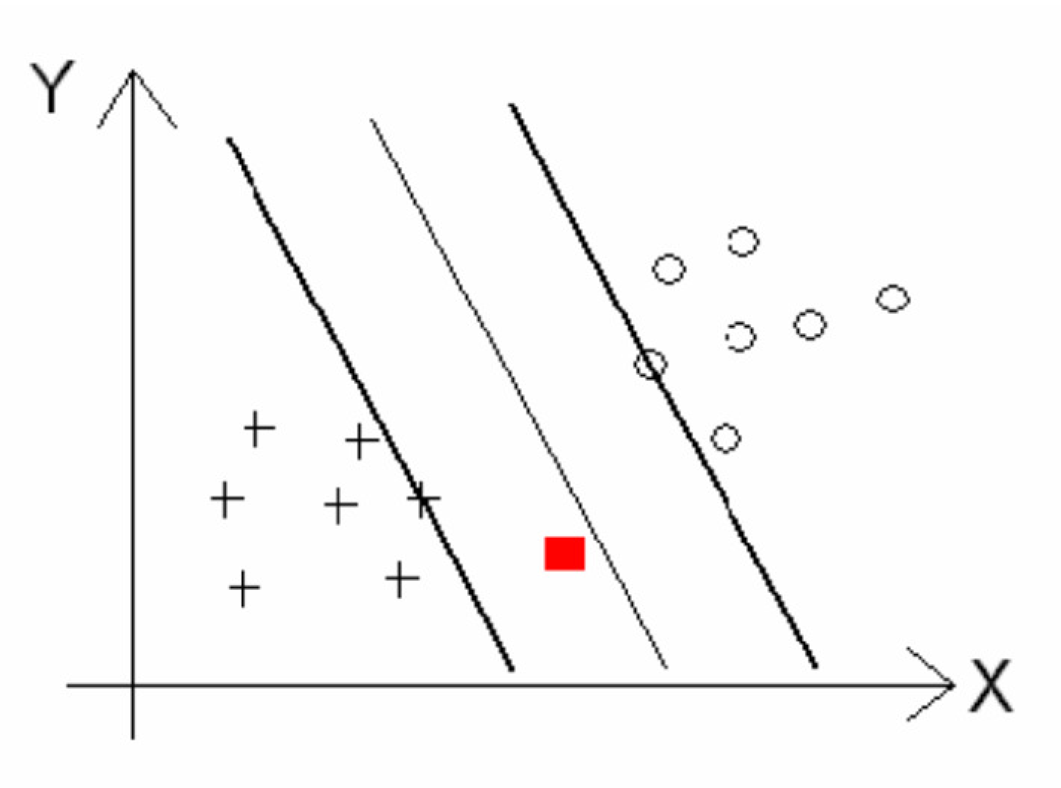
Property2 **:** in the optimization problem are involved only the scalar products between observations x.

Maximizing the margin



Classification of a new data

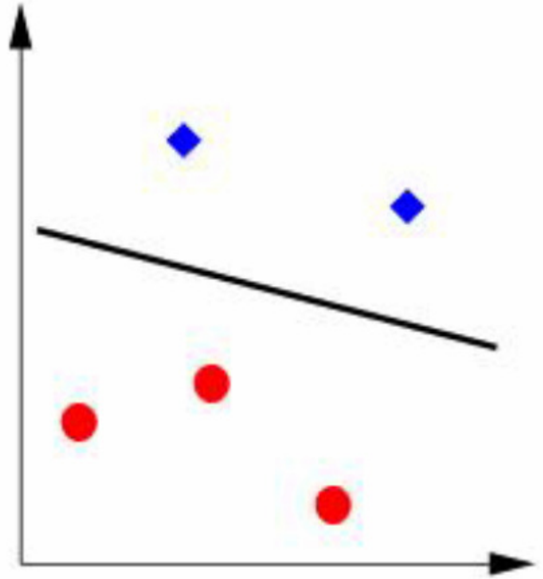
In general, the classification of a new example (assignment) is given by its position relative to the optimal hyperplane.



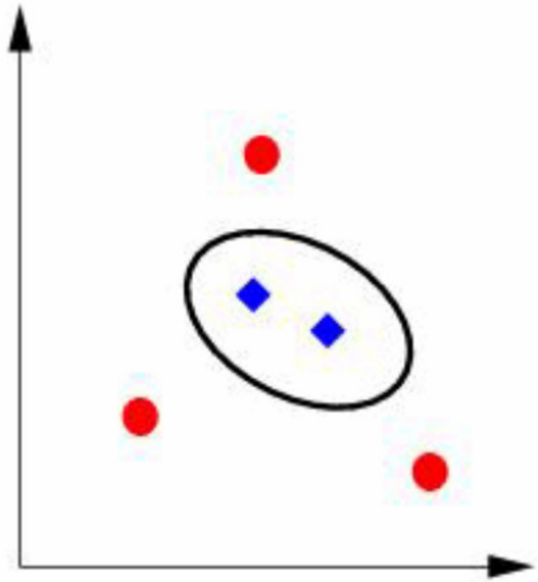
Non-linear SVM

- General idea: the original input space can be mapped to some higher-dimensional feature space where the training set is separable:

Linearly separable case

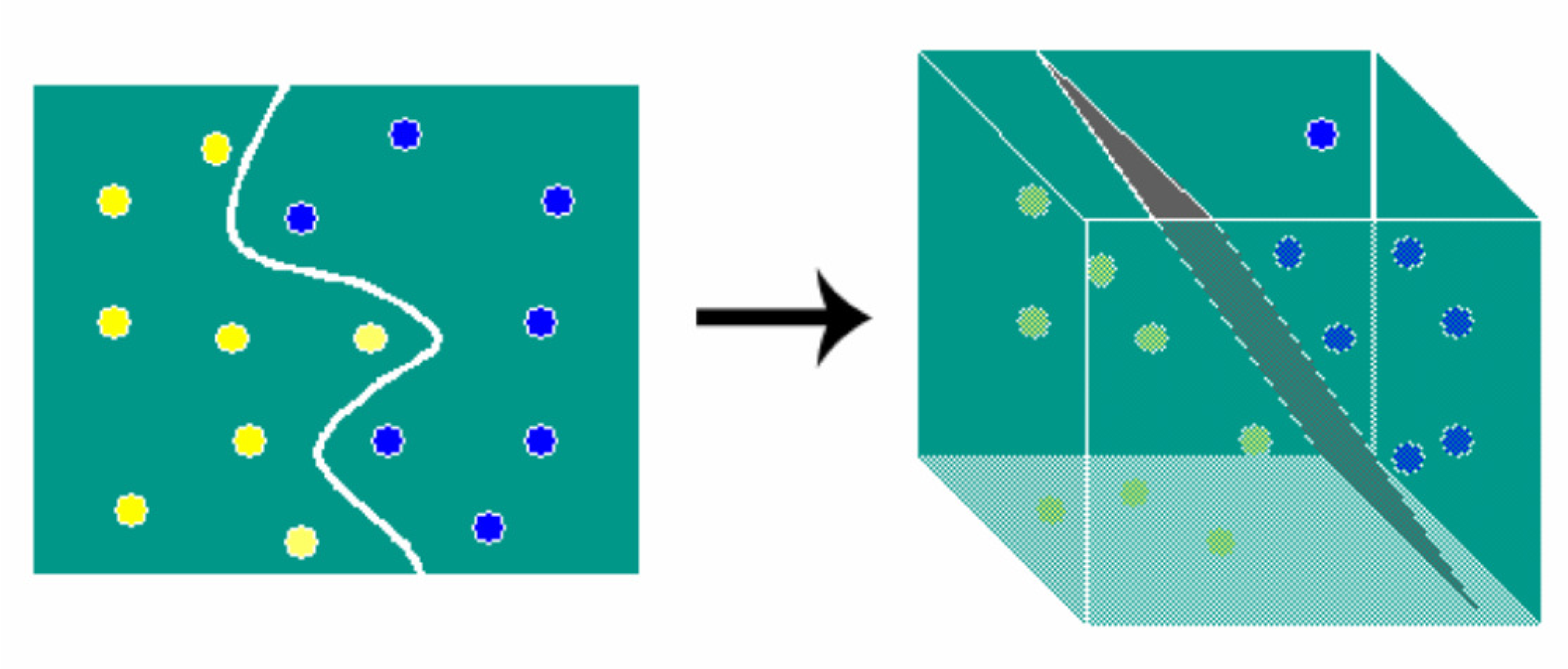


Non-linearly separable case



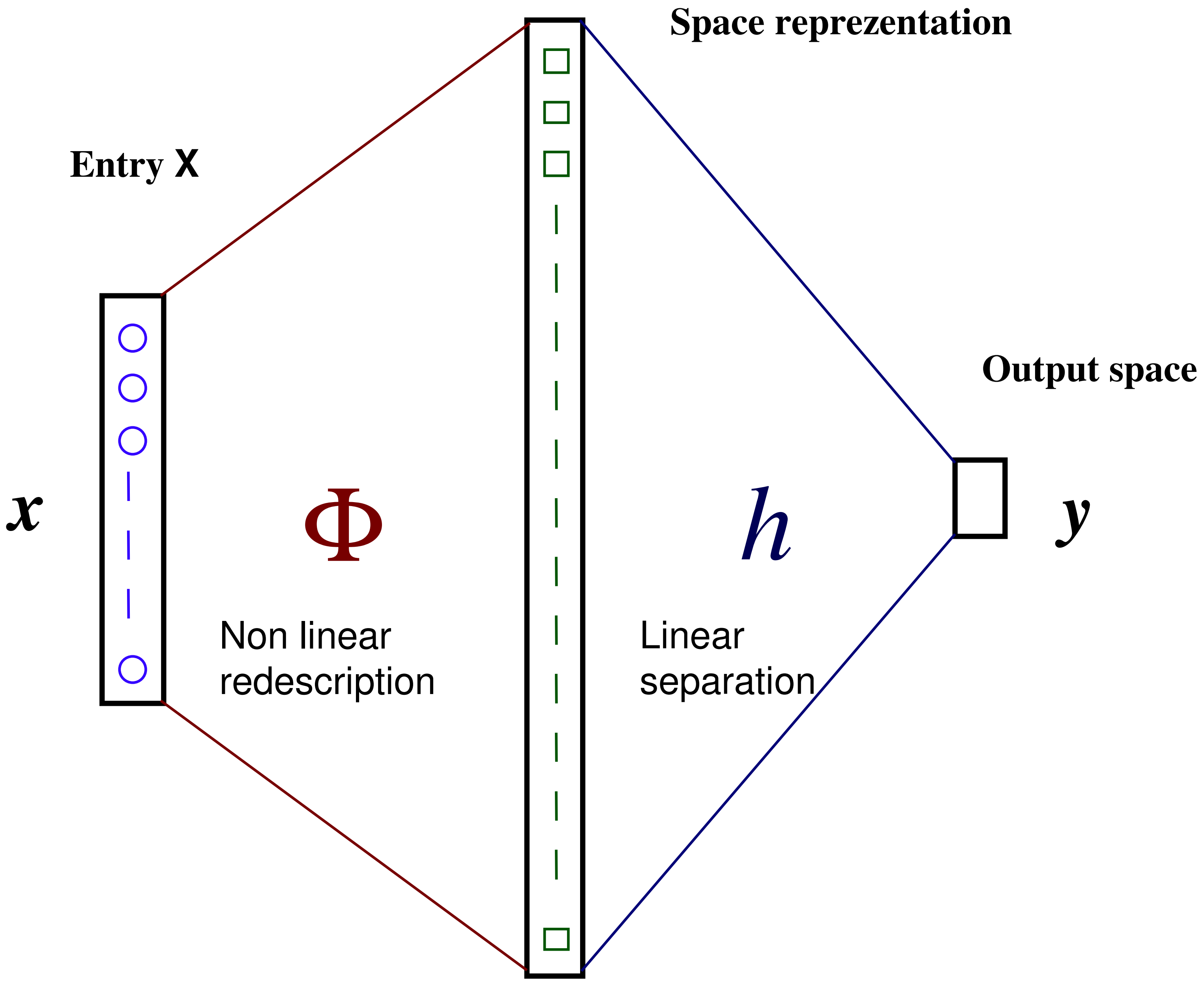
Non-linear SVM

Idea: change the data space, change of dimension ("space re-description").



More the re-description dimension is higher - the probability to find the hyperplane between the objects are higher.

SVM & re-description



The practical use

Choose:

- The type of the kernel function *K*
  - Its shape;
  - Its parameters ;
- The value of the constant *C*;
- The careful selection of these parameters requires an estimate of the Vapnik-Chervonenkis dimension:
- In the separable case, it is possible to determine these parameters;
- In the case of non-separability, it must be tested with empirical methods to make the best choice;

Kernel fonctions

- **Polynomial** :

  Polynomials of degree q have the following associated kernel function: $K(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}.\boldsymbol{x}' + 1)^q$

- **RBF** :

  The radial basis based functions : $h(\boldsymbol{x}) = sign\left(\sum_{i=1}^{n} \alpha_i \exp\left\{ -\frac{|\boldsymbol{x} - \boldsymbol{x}_i|^2}{\sigma^2} \right\} \right)$

  Have the kernel function: $K(\boldsymbol{x}, \boldsymbol{x}') = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}}$

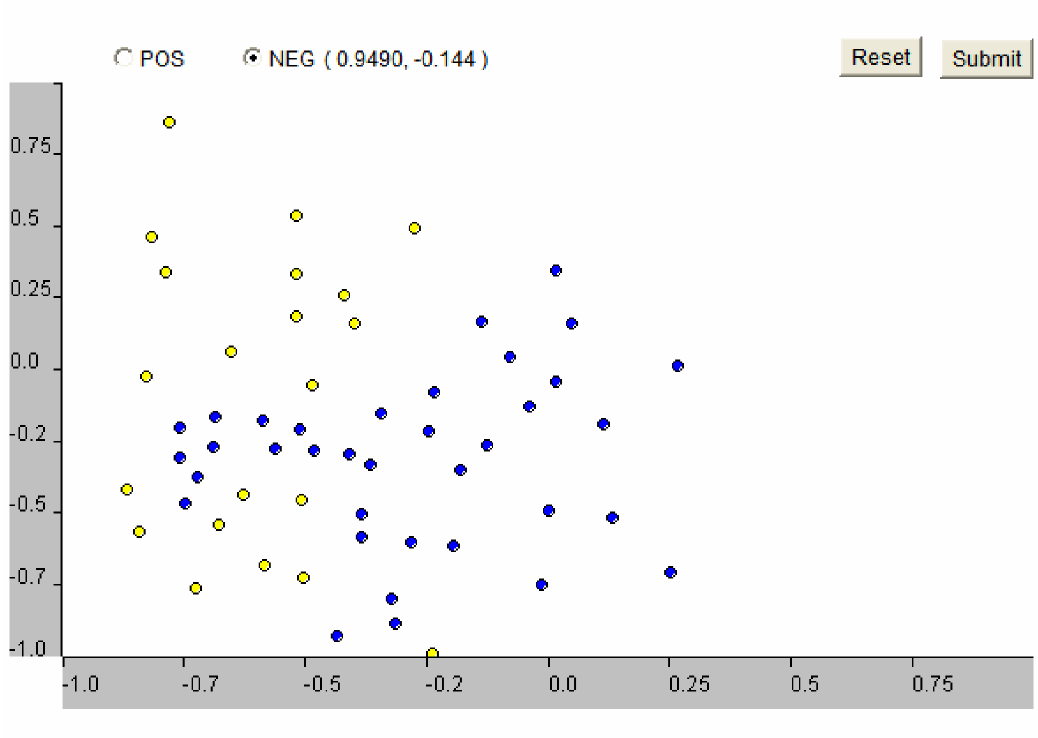
- **Sigmoid**:

  Neural networks based on activation functions:

  Have the kernel function: $h(\boldsymbol{x}) = sign\left(\sum_{i=1}^{n} \alpha_i \tanh\{v(\boldsymbol{x}.\boldsymbol{x}_i) + a\} + b \right)$

$$K(\boldsymbol{x}, \boldsymbol{x}') = \tanh(a\boldsymbol{x}.\boldsymbol{x}' - b)$$

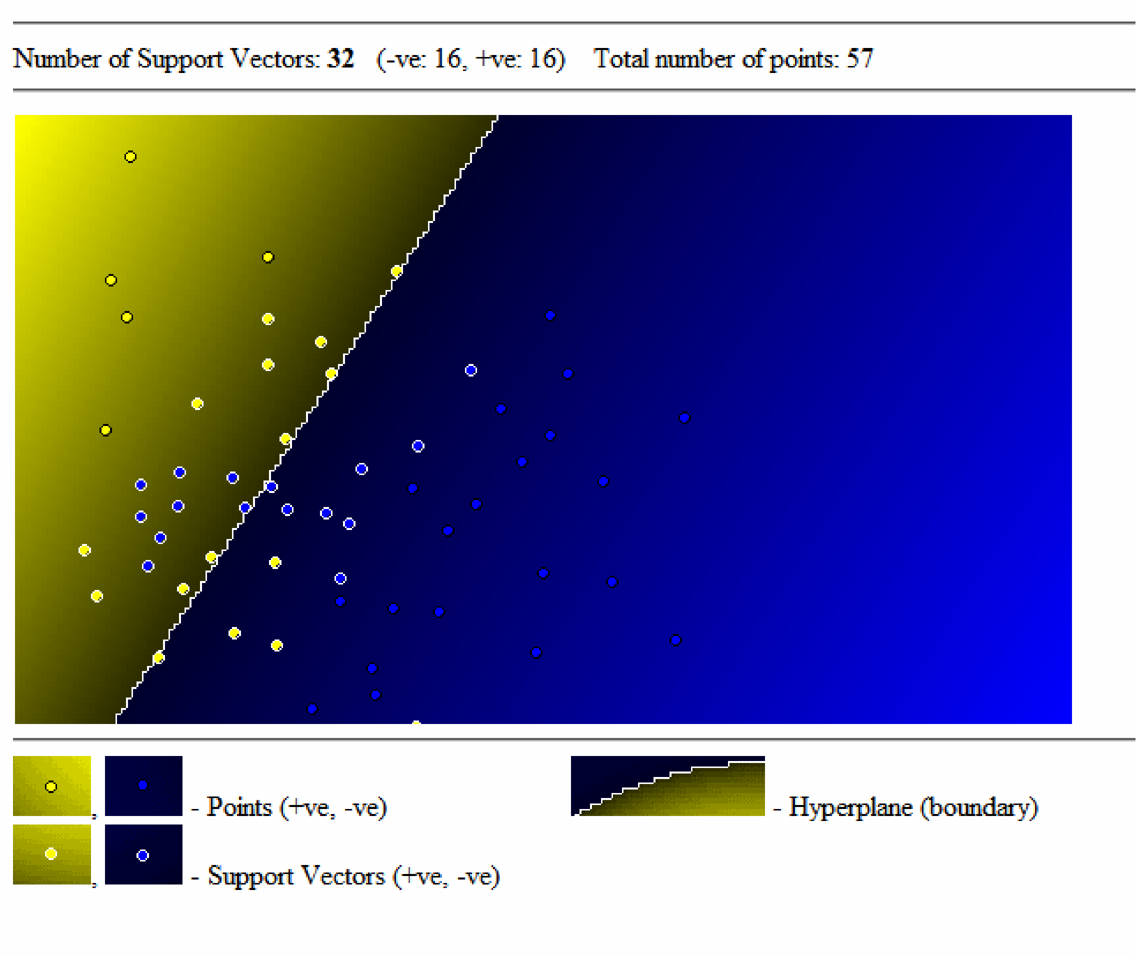
SVM : linear kernel (1)



○ POS ● NEG (0.9490, -0.144)

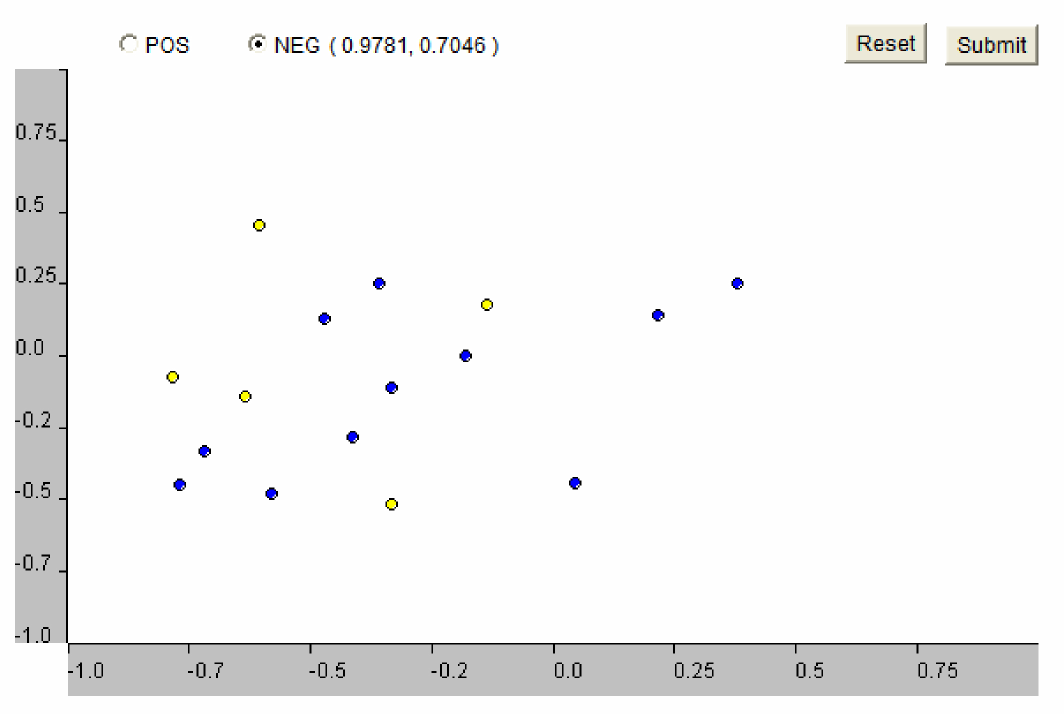
C = 10000 (penality error)

SVM : linear kernel (2)



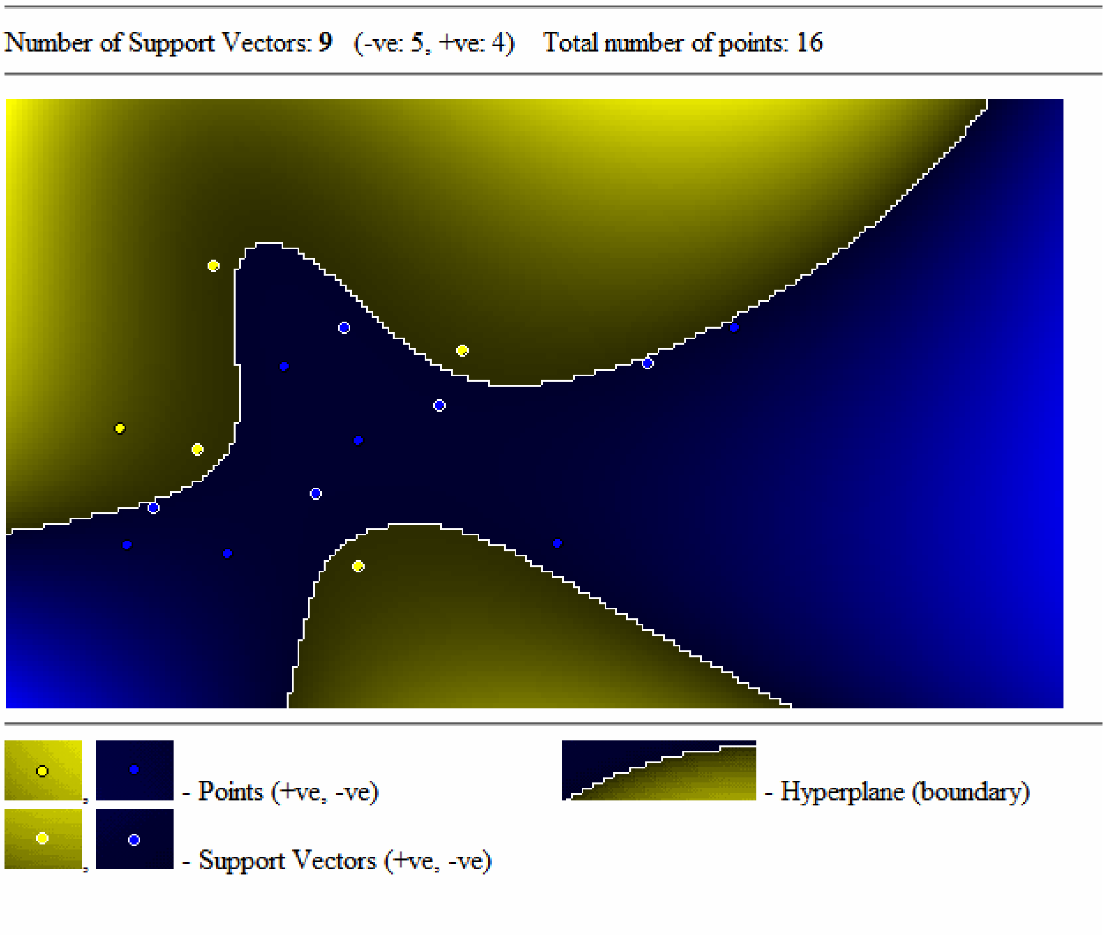
Number of Support Vectors: **32** (-ve: 16, +ve: 16) Total number of points: **57**

SVM : polynomial kernel(1)



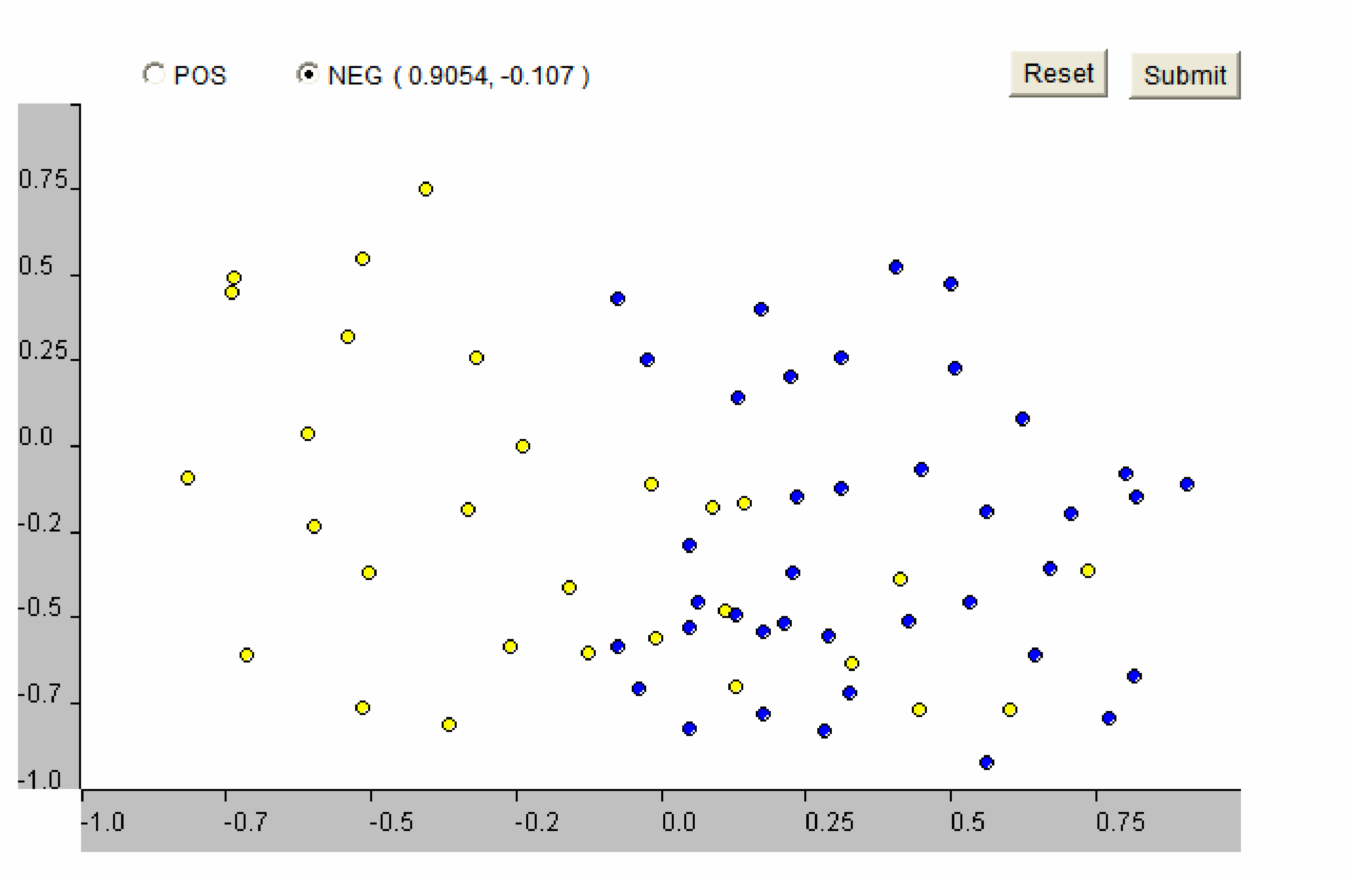
C = 10000 (penality error);
Degree : 5;

SVM : polynomial kernel (2)



Number of Support Vectors: 9 (-ve: 5, +ve: 4) Total number of points: 16

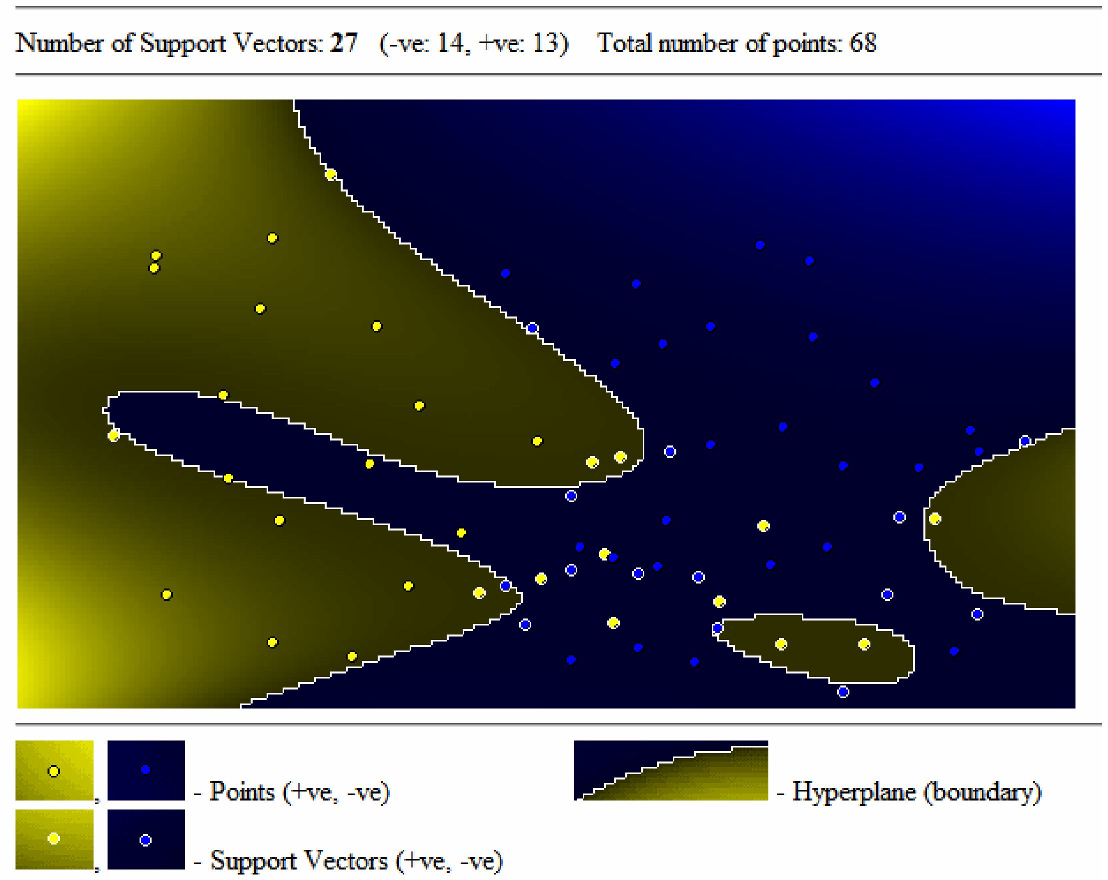
SVM : Gaussian kernel(1)


○ POS ● NEG (0.9054, -0.107)

C = 10000 (penality error);
sigma : 10;

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\,\mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}}.\qquad H = 2\sqrt{2\ \ln(2)}\,\sigma \simeq 2{,}3548\sigma.$$

SVM : Gaussian kernel (2)



Number of Support Vectors: **27** (-ve: 14, +ve: 13) Total number of points: 68

- Points (+ve, -ve)

- Support Vectors (+ve, -ve)

- Hyperplane (boundary)

SVM : demos & software

2D Pattern Recognition :

http://svm.dcs.rhbnc.ac.uk/pagesnew/GPat.shtml

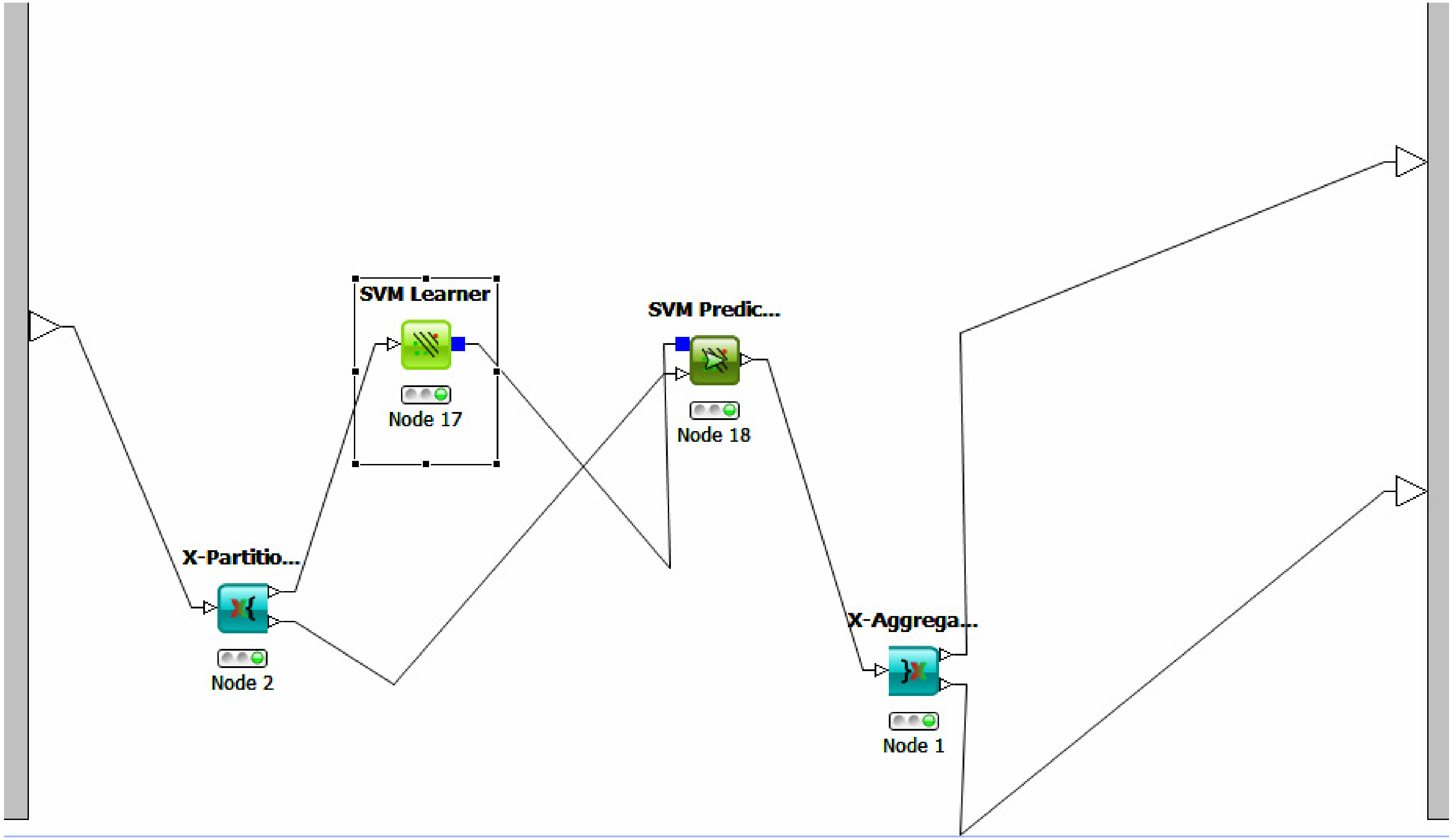
KNIME :

http://www.knime.org/

KNIME

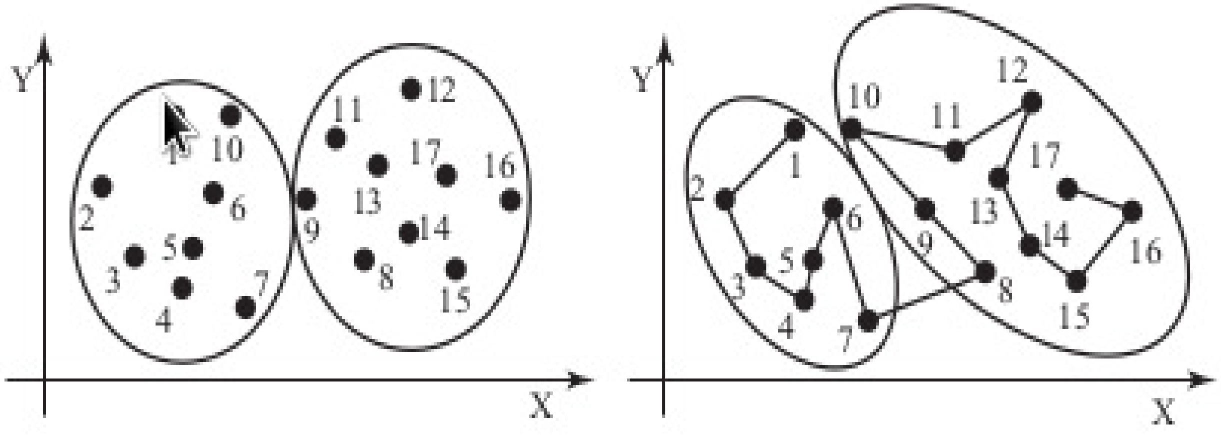
SVM Learner;

SVM Predictor;



SEQUENTIAL DATA CLUSTERING

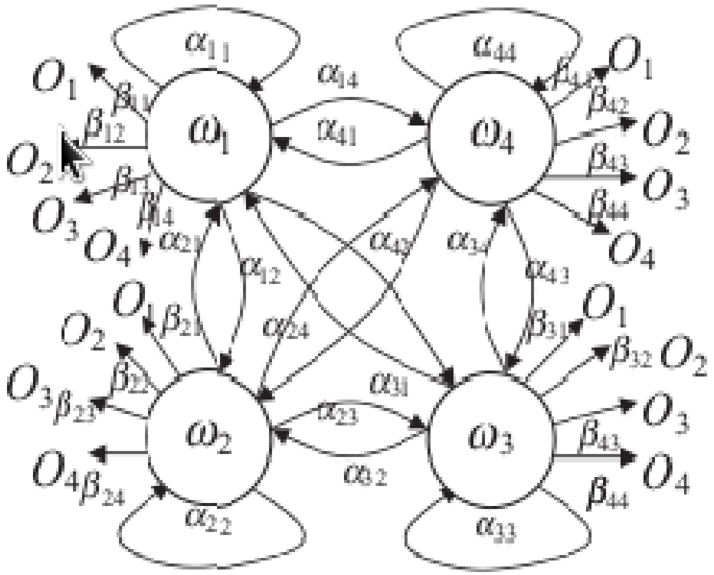
- Sequential data consist of a sequence of sets of units with possibly variable length and other interesting characteristics, such as dynamic behaviors and time constraints
- Sequential data could be generated from a large number of task sources, such as DNA sequencing, speech processing, text mining, medical diagnosis, stock market analysis, customer transactions, web data mining, and robot



(a) A conventional clustering

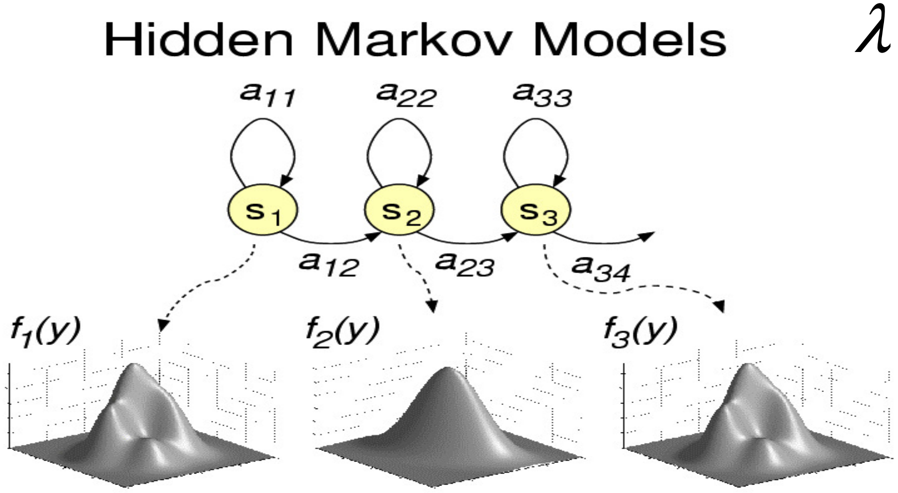
(b) A sequential clustering

Hidden Markov Model



A four-state hidden Markov model. Each hidden state is associated with four visible observations.

HMM



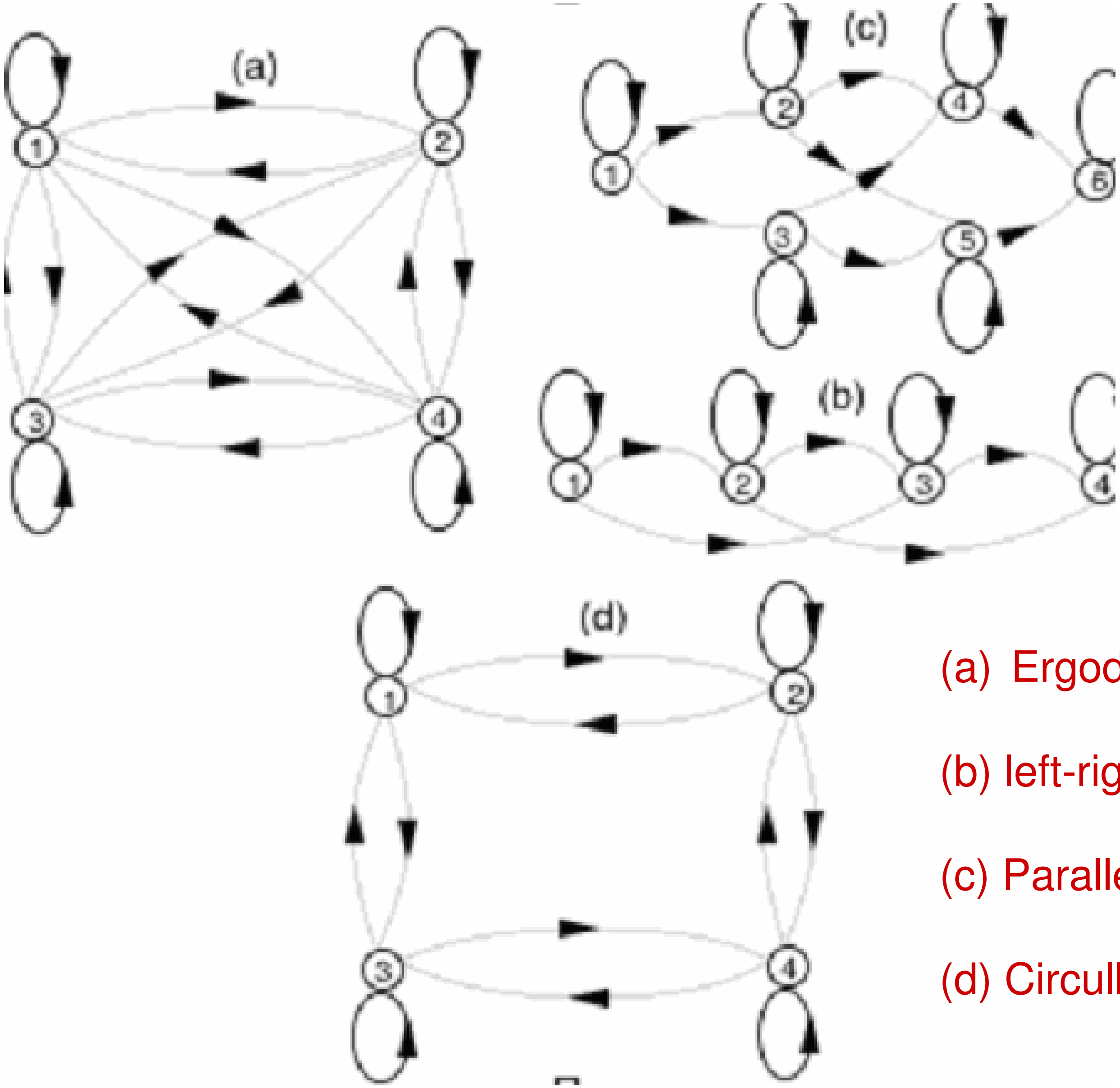
$$\lambda = (A, B, \pi)$$

$A = \{a_{ij}\}$ the probability distribution of the state transitions

$\mathrm{B} = \{\mathrm{b}_j(\mathrm{k})\}$ the emission probability distribution of symbols observed in the states

$\pi = \{\pi_i\}$ initial distribution

Différents modèles de HMM



(a) Ergodic model

(b) left-right model

(c) Parallel left-right model

(d) Circullar model

Cluster Validity

- Cross Validation

In *k*-fold cross-validation, the original sample is randomly partitioned into *k* equal size subsamples. Of the *k* subsamples, a single subsample is retained as the validation data for testing the model, and the remaining *k* – 1 subsamples are used as training data.

The cross-validation process is then repeated *k* times (the *folds*), with each of the *k* subsamples used exactly once as the validation data. The *k* results from the folds then can be averaged (or otherwise combined) to produce a single estimation.

The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used,but in general *k* remains an unfixed parameter

Practical example: Kros Validation workflow in KNIME