Inference on Graphs with Support Vector Machines

Jean-Philippe Vert

Ecole des Mines de Paris, France
Jean-Philippe.Vert@mines.org

INSEAD, Feb. 6, 2004

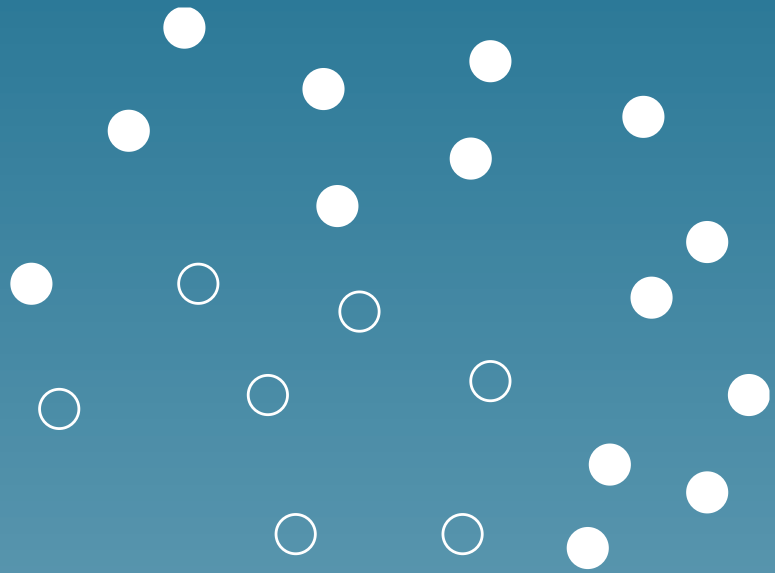
Outline

1. Introduction to SVMs
2. Inference on graphs

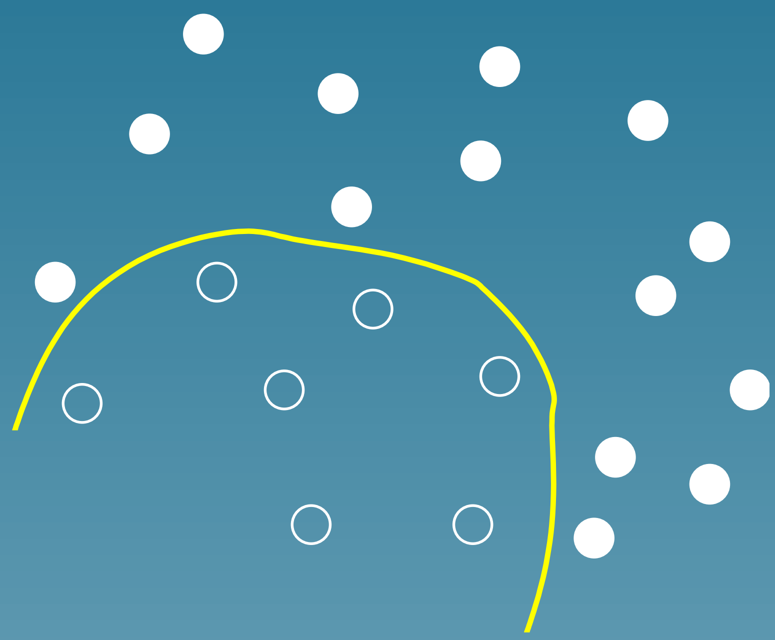
Part 1

Support Vector Machines (SVMs)

The pattern recognition problem

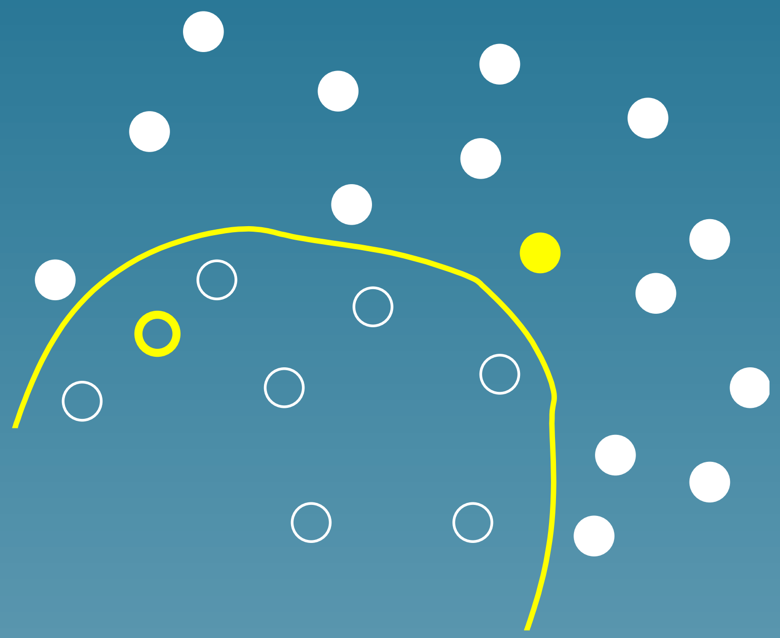


The pattern recognition problem



- Learn from labelled examples a discrimination rule

The pattern recognition problem



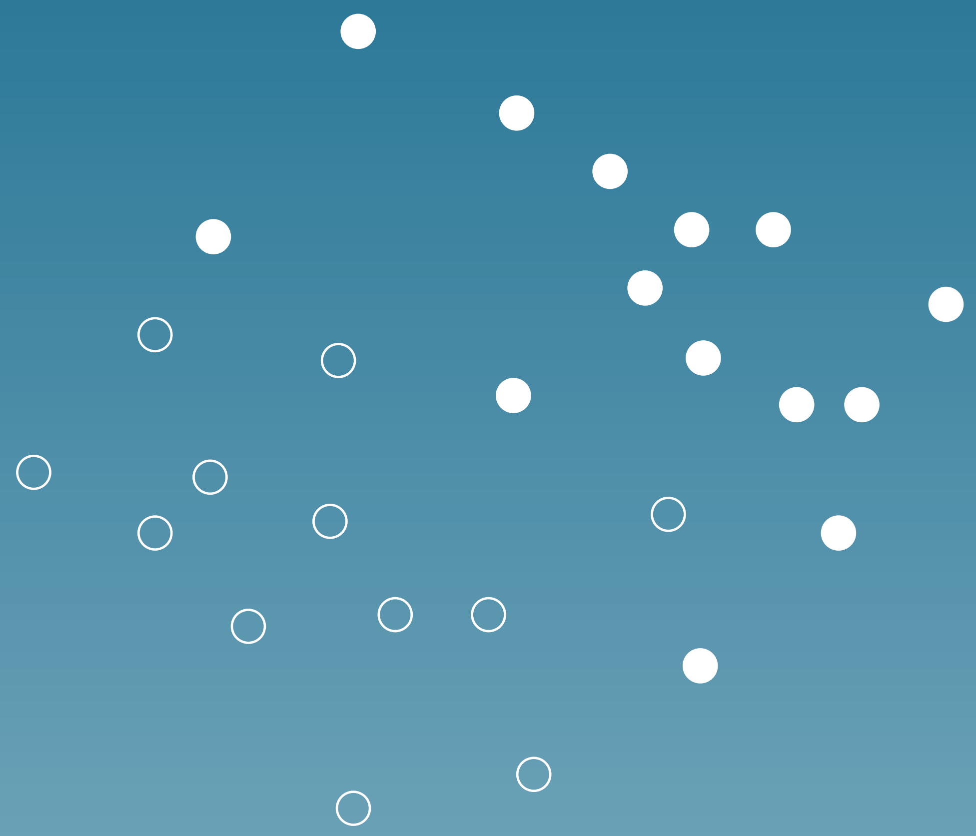
- Learn from labelled examples a discrimination rule
- Use it to predict the class of new points

Pattern recognition examples

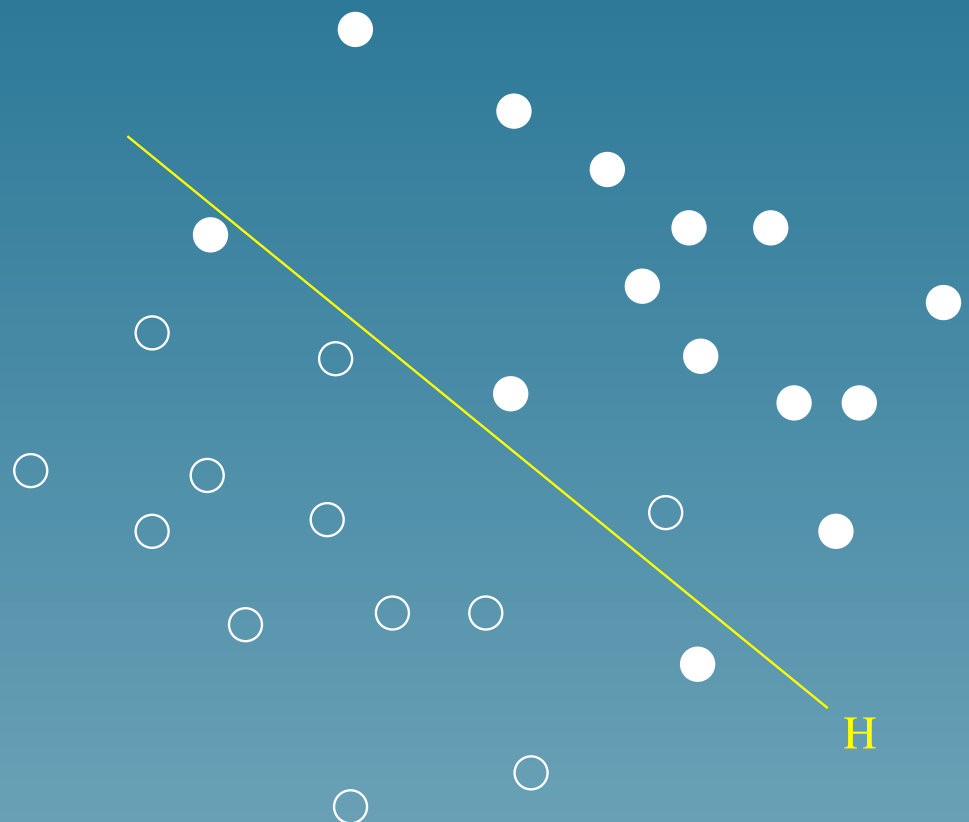
- Hand-written digit recognition
- Medical diagnosis
- Direct marketing
- Predicting the future...

Remark: other problems are possible: multi-class, continuous values, etc...

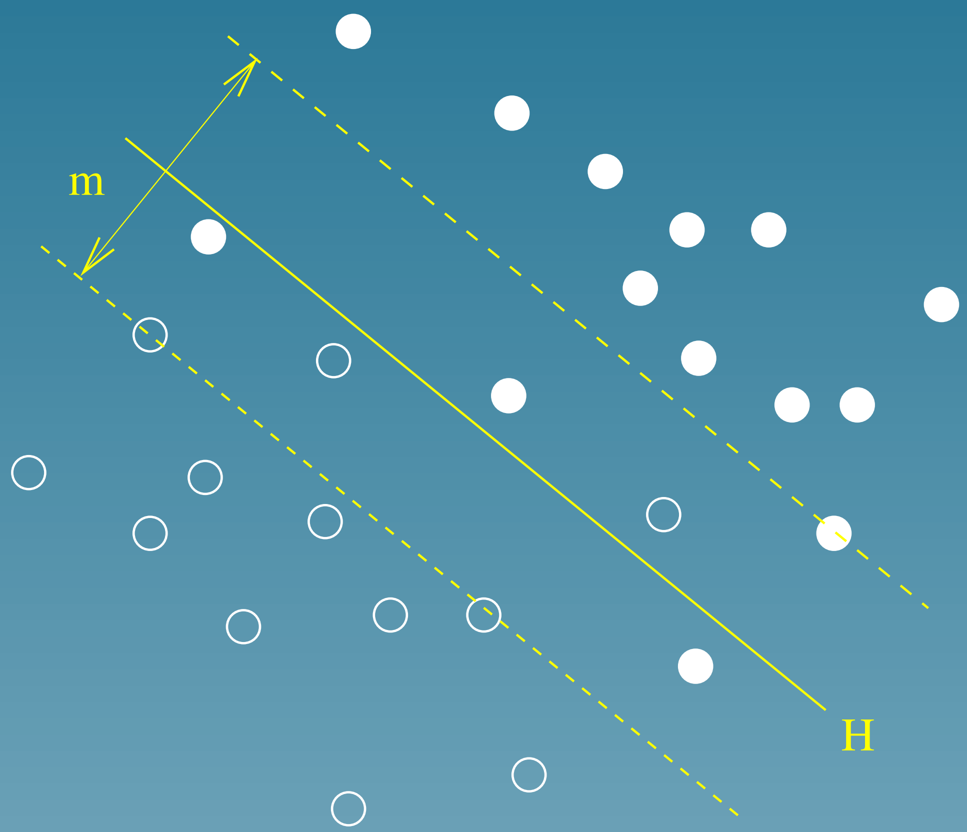
Linear SVM



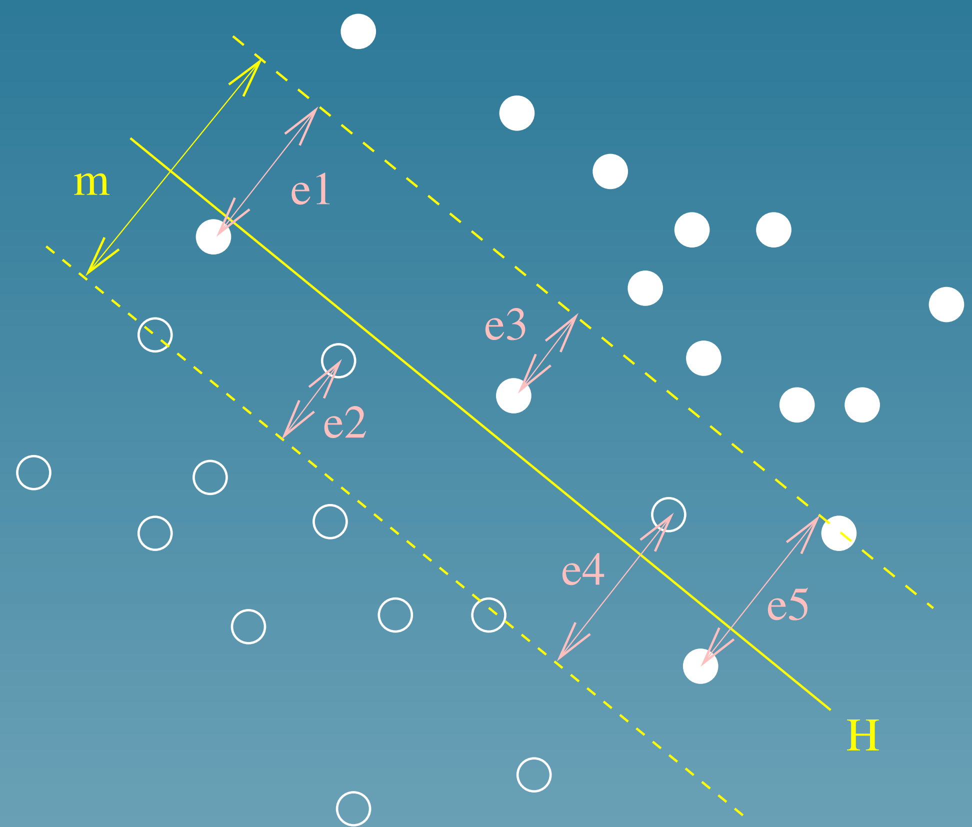
Linear SVM



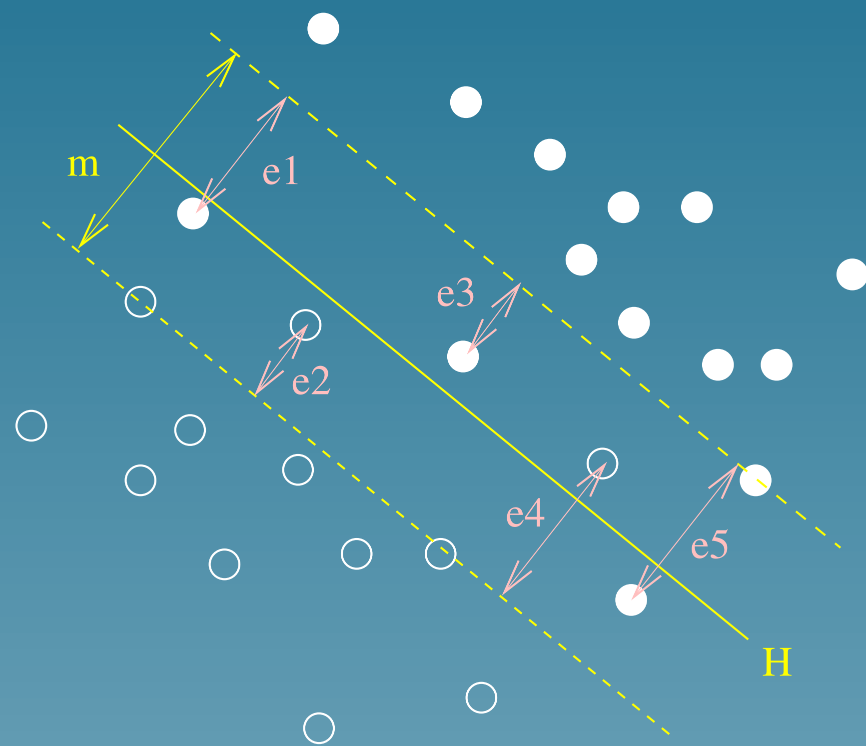
Linear SVM



Linear SVM



Linear SVM



$$\min_{H}\left\{\min_{m}\left[\frac{1}{m^2}+C\sum_{i}e_i\right]\right\}$$

Dual formulation

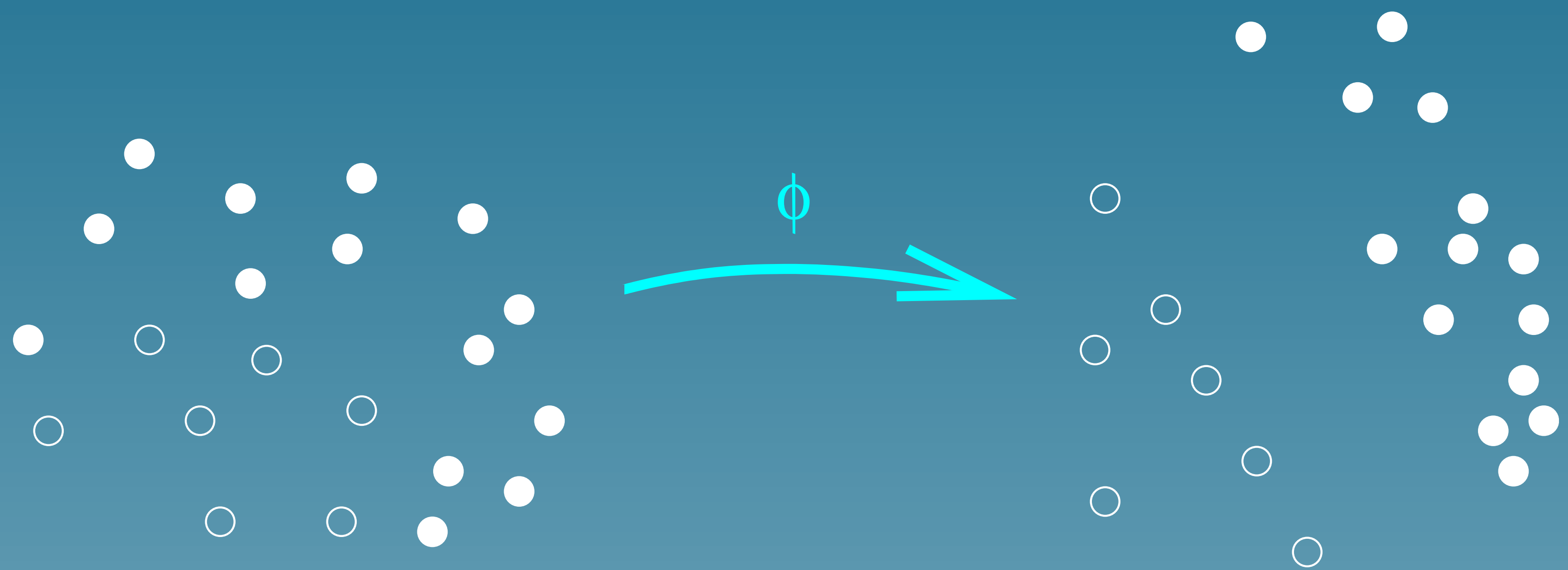
The classification of a new point x is the sign of:

$$f(x) = w.x + b = \left(\sum_i \alpha_i x_i \right) .x + b,$$

where α_i solves:

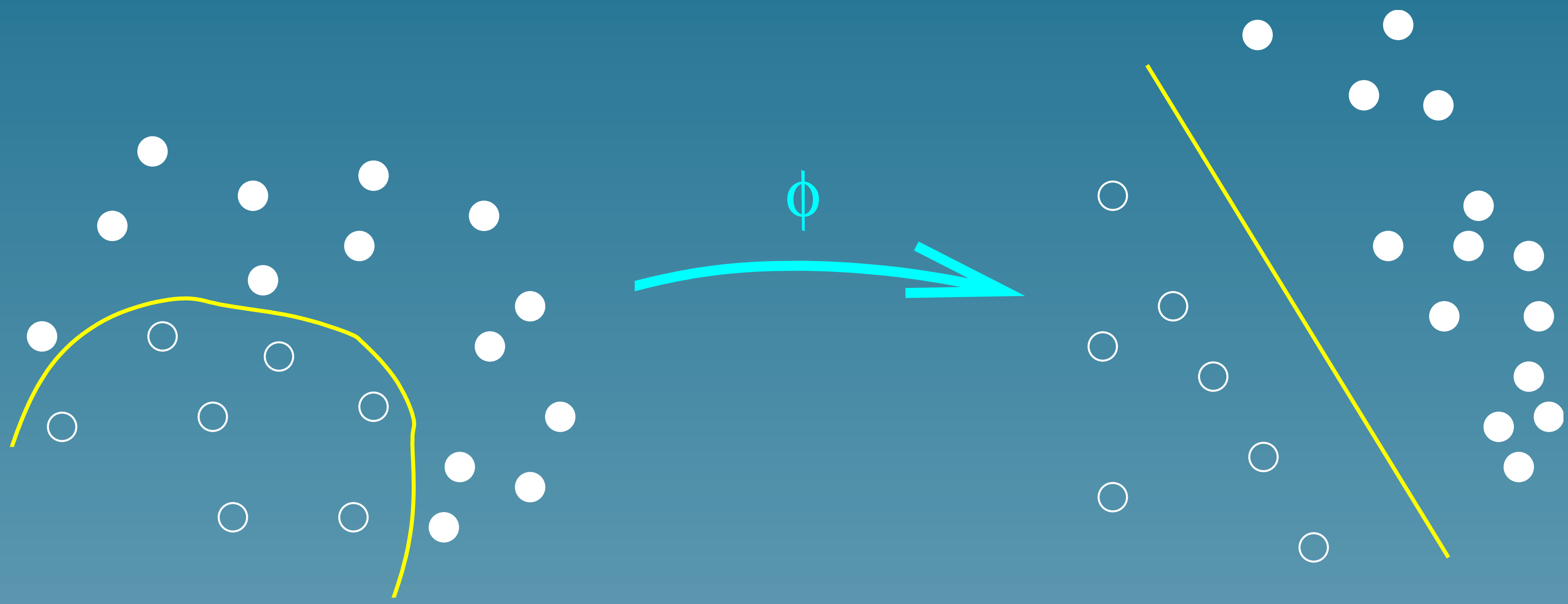
$$\begin{cases} \max_{\vec{\alpha}} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j x_i . x_j \\ \forall i = 1, \ldots, n \quad 0 \leq \alpha_i \leq C \\ \sum_{i=1}^n \alpha_i y_i = 0. \end{cases}$$

General Support Vector Machines



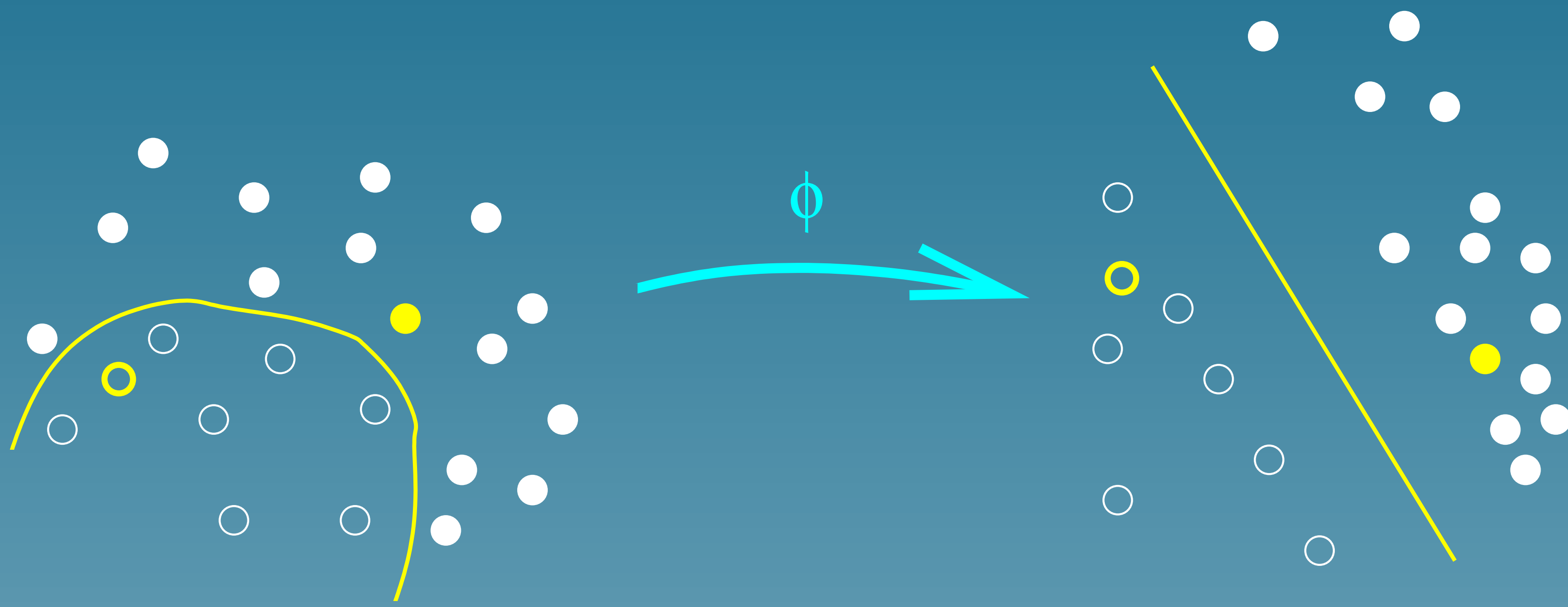
- Object x represented by the vector $\Phi(\vec{x})$ (feature space)

General Support Vector Machines



- Object x represented by the vector $\Phi(\vec{x})$ (feature space)
- Linear SVM in the feature space

General Support Vector Machines



- Object x represented by the vector $\Phi(\vec{x})$ (feature space)
- Linear SVM in the feature space

Dual formulation

The classification of a new point x is the sign of:

$$f(x) = w.\Phi(\vec{x}) + b = \left(\sum_i \alpha_i \Phi(\vec{x}_i) \right) .\Phi(\vec{x}) + b,$$

where α_i solves:

$$\begin{cases} \max_{\vec{\alpha}} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \Phi(\vec{x}_i)_i . \Phi(\vec{x}_j) \\ \forall i = 1, \ldots, n \quad 0 \leq \alpha_i \leq C \\ \sum_{i=1}^n \alpha_i y_i = 0. \end{cases}$$

A useful trick

Let

$$K(x,y) := \Phi(\vec{x}).\Phi(\vec{y})$$

K is called a kernel.

Dual formulation using the kernel

The classification of a new point x is the sign of:

$$f(x) = w.\Phi(\vec{x}) + b = \sum_i \alpha_i K(x_i, x) + b,$$

where α_i solves:

$$\begin{cases} \max_{\vec{\alpha}} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \forall i = 1, \ldots, n \quad 0 \leq \alpha_i \leq C \\ \sum_{i=1}^n \alpha_i y_i = 0. \end{cases}$$

The kernel trick for SVM

- The separation can be found without computing $\Phi(x)$ explicitly. Only the kernel matters:

$$K(x,y) = \Phi(\vec{x}).\Phi(\vec{y})$$

- Simple kernels $K(x,y)$ can correspond to complex $\vec{\Phi}$
- SVM work with any sort of data as soon as a kernel is defined

Kernel examples

- Linear :

$$K(x, x') = x.x'$$

- Polynomial :

$$K(x, x') = (x.x' + c)^d$$

- Gaussian RBf :

$$K(x, x') = \exp\left(-\frac{||x - x'||^2}{2\sigma^2}\right)$$

Kernels

For any set $\mathcal{X}$, a function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel iff:

- it is symmetric :
$$K(x, y) = K(y, x),$$
- it is positive semi-definite:
$$\sum_{i,j} a_i a_j K(x_i, x_j) \geq 0$$
for all $a_i \in \mathbb{R}$ and $x_i \in \mathcal{X}$

Advantages of SVM

- Works well on real-world applications
- Large dimensions, noise OK (?)
- Can be applied to any kind of data as soon as a kernel is available

Part 2

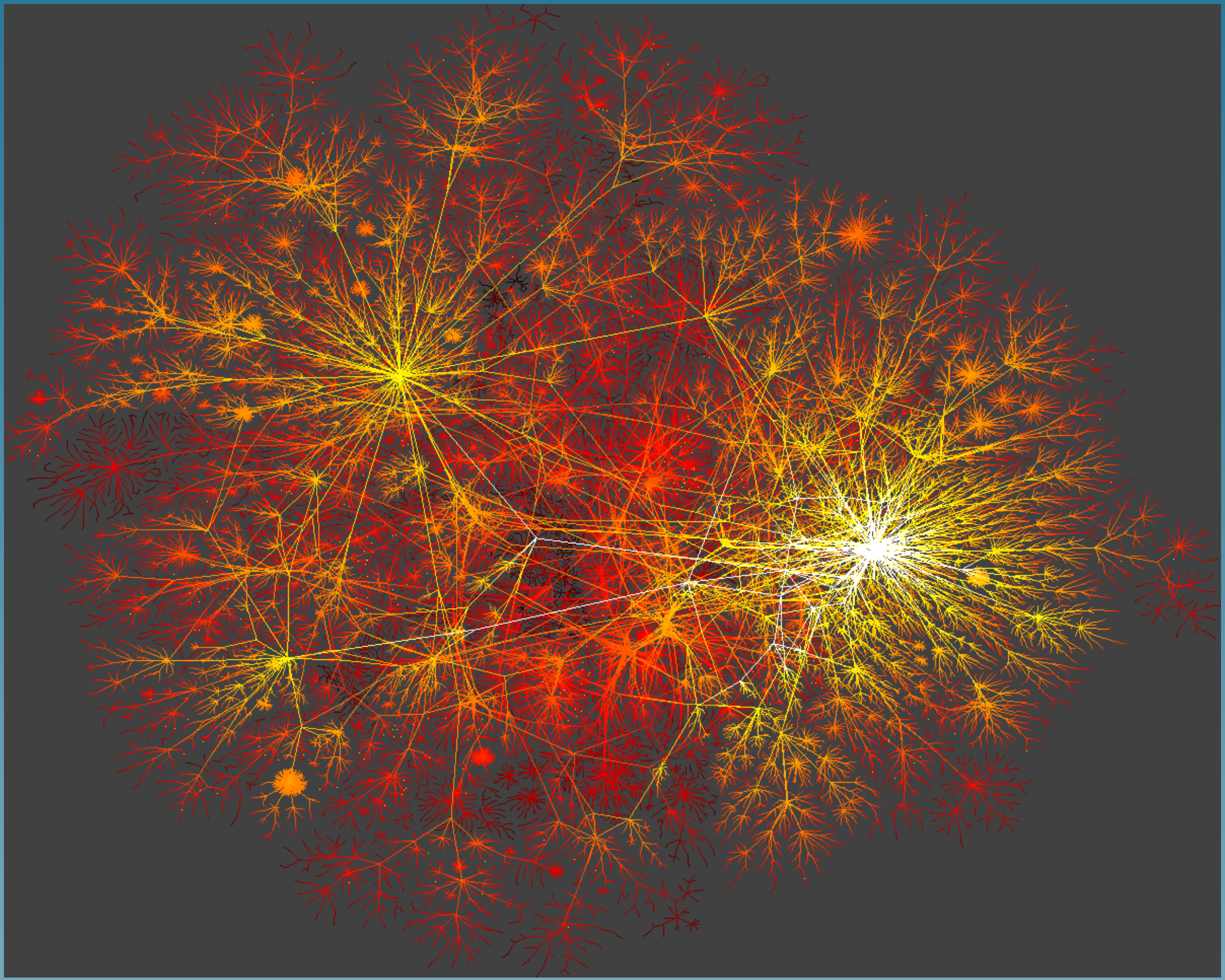
Inference on Graphs

Motivations

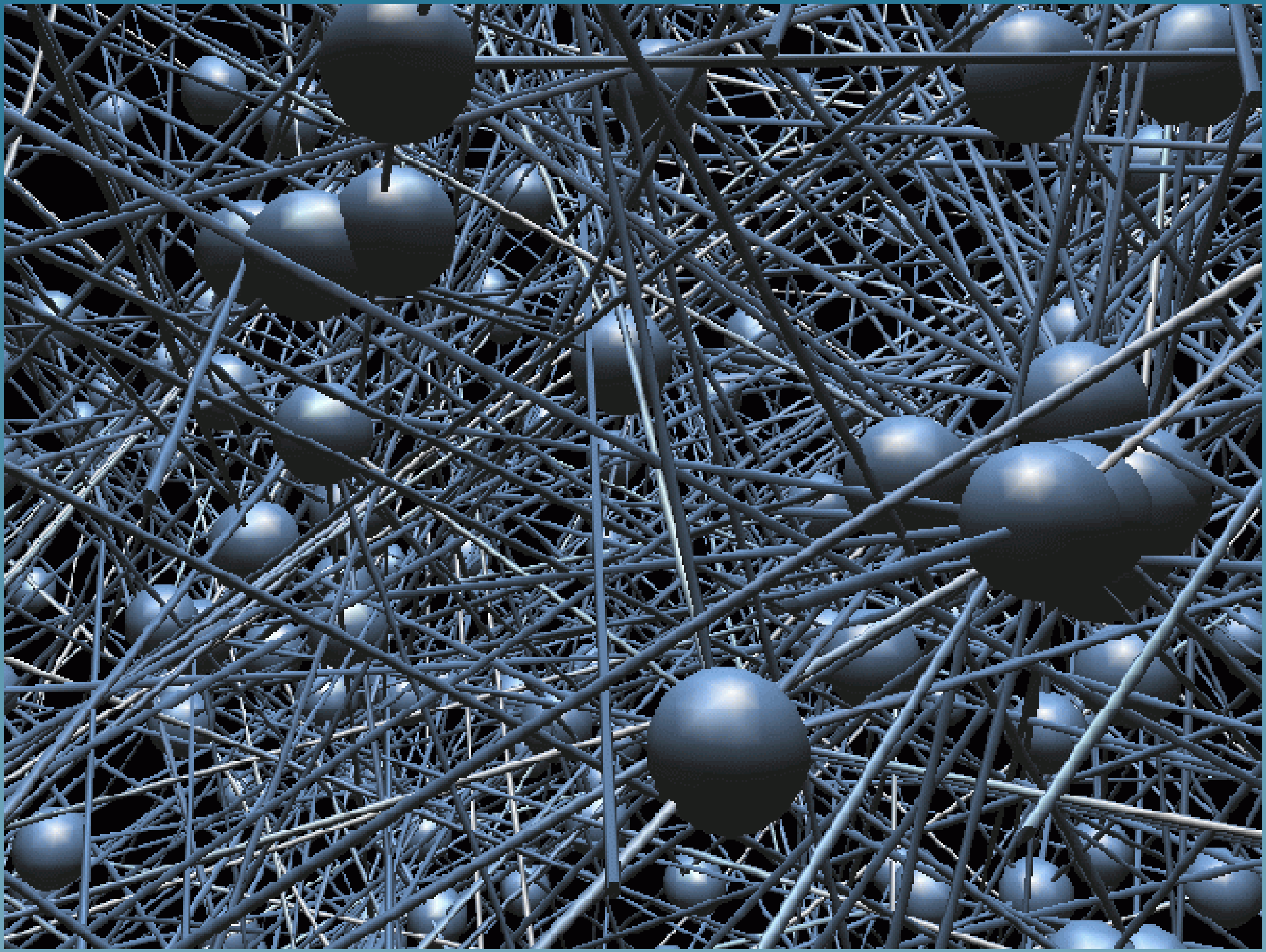
Data to be analyzed are often not vectors, but rather nodes of a network

- by nature,
- by discretization/sampling of a continuous space
- because it's convenient.

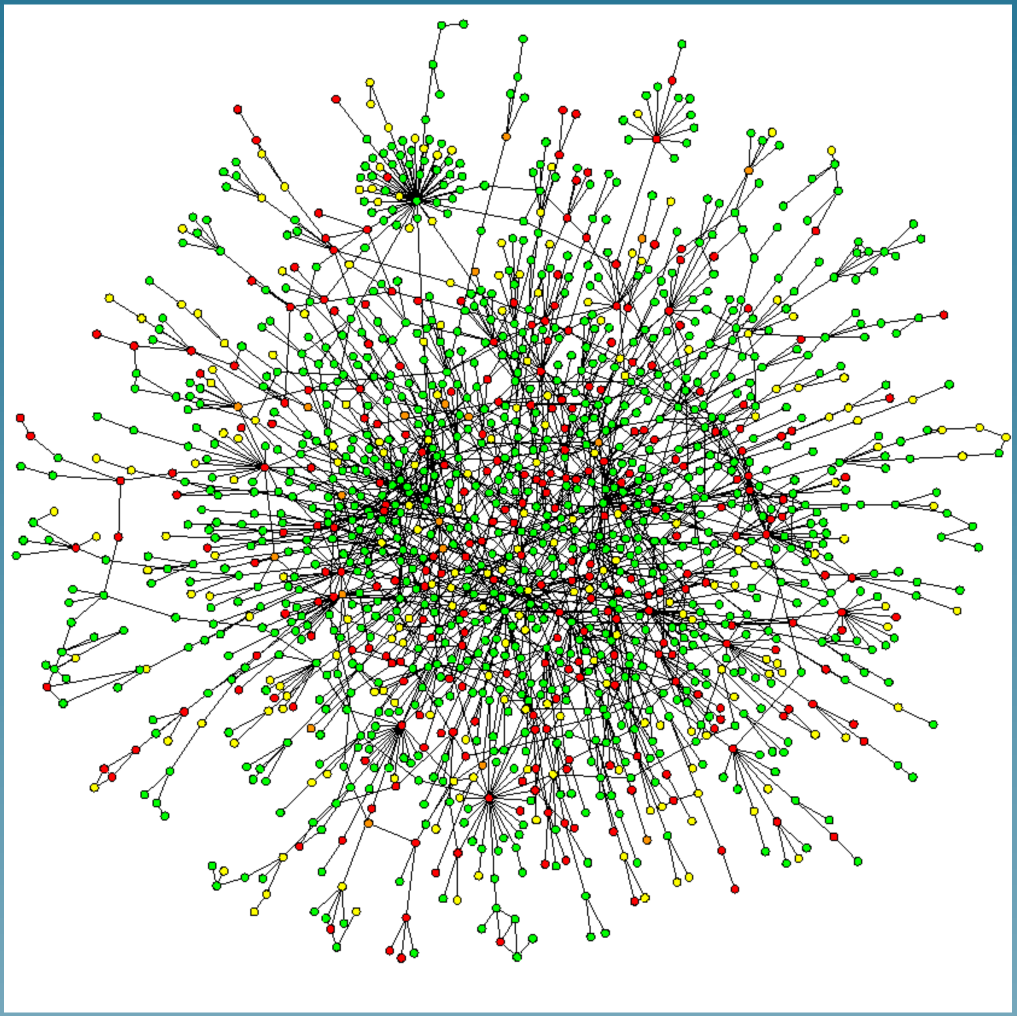
Internet (by nature)



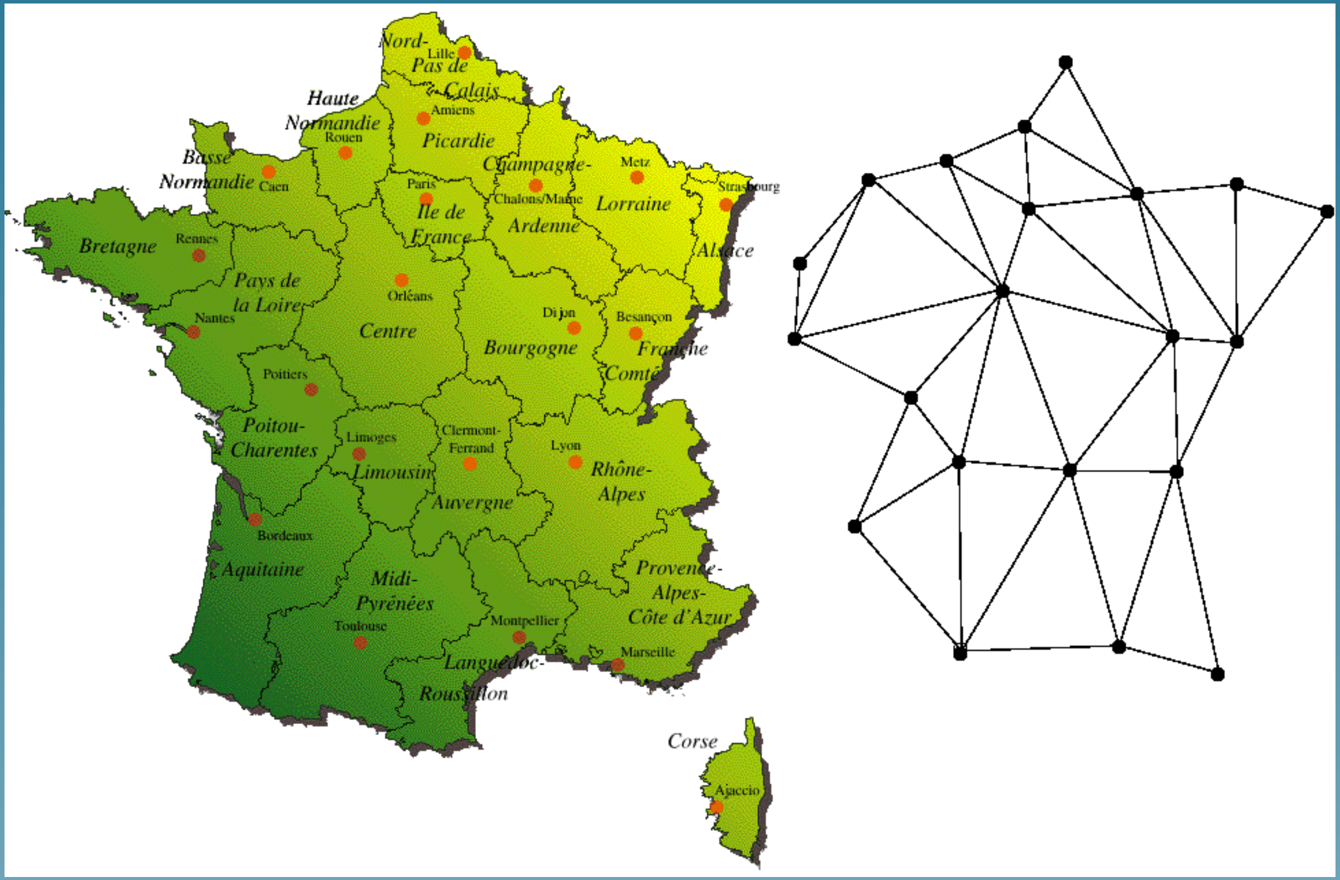
Social Network (by nature)



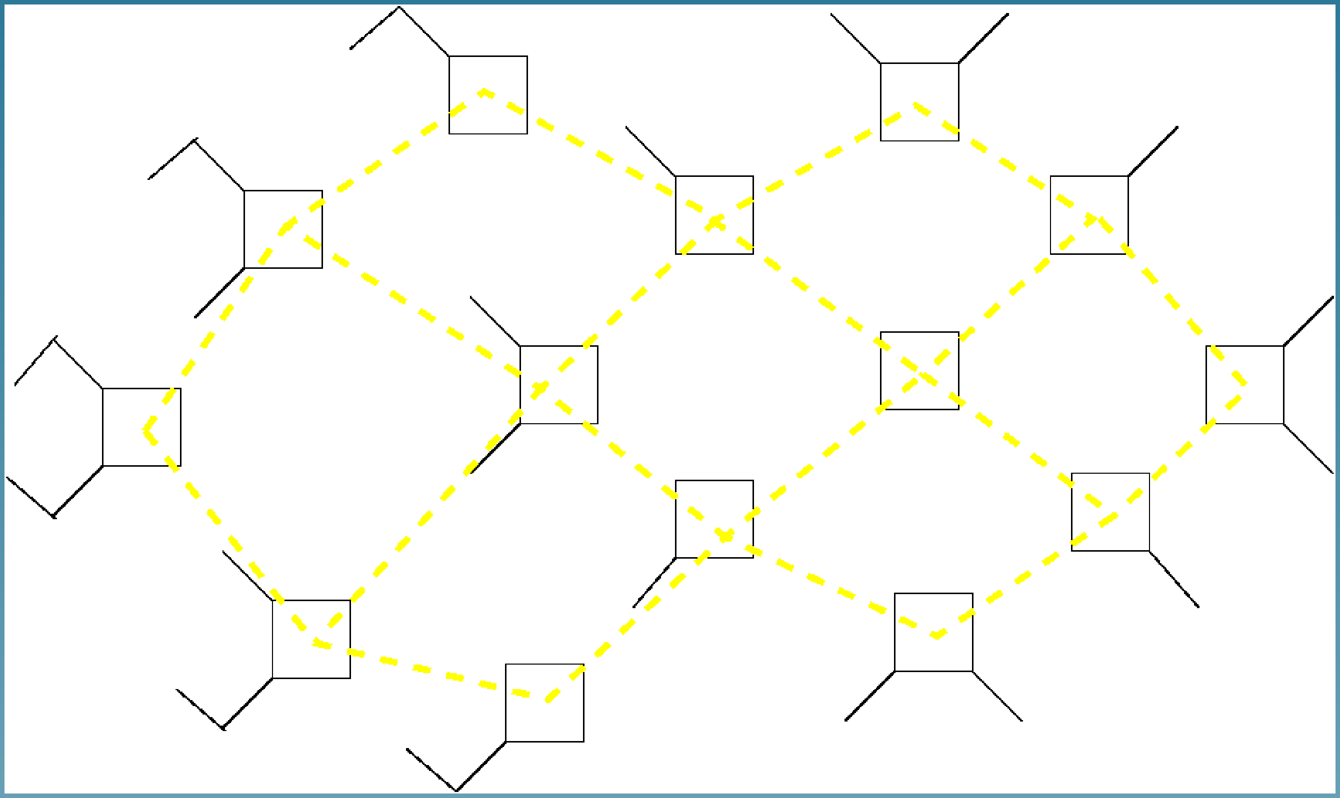
Protein interaction network (by nature)



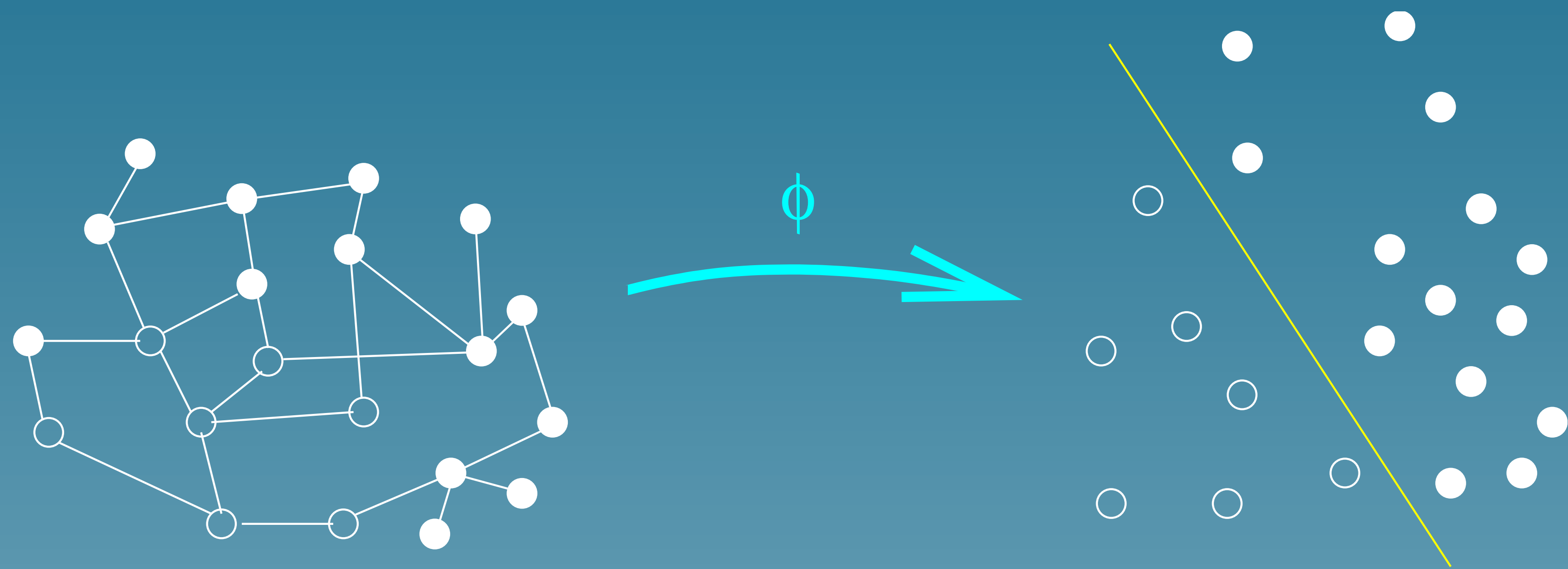
Spatial data (by discretization)



Molecules (by convenience)



SVM on a graph



We need a kernel $K(x, y)$ between nodes.

Using a distance?

- Remember the Gaussian kernel

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

- Let $d(x, x')$ a distance on the graph, e.g., the length of the shortext path between nodes.
- Soit $K(x, x') = \exp(-d(x, x')^2/2\sigma^2)$
- Problem: not a valid kernel...

Using the heat equation?

Let $K_x(t,y)$ the temperature at time t and position y. K_x solves the heat equation:

$$\frac{\partial K_x}{\partial t} = \Delta K_x.$$

The solution is the Gaussian kernel:

$$K_x(t,y) = \frac{1}{\sqrt{4\pi}} \exp\left(-\frac{||x-y||^2}{4t}\right)$$

(interpretation: describes how heat, gas, introduced at x, diffuse over time)

The Laplacian

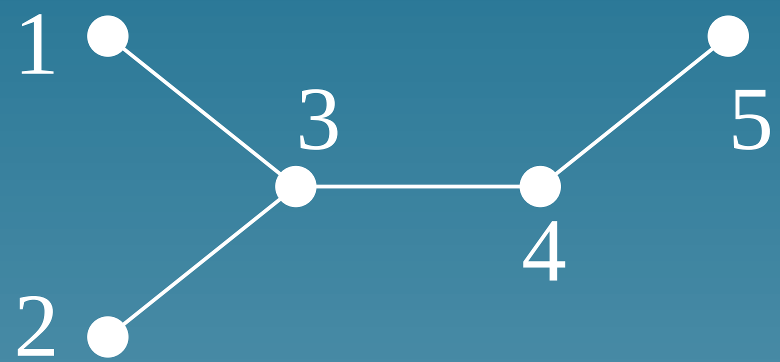
- For vectors,

$$\Delta = \sum_{i=1}^{p} \frac{\partial}{\partial x_i}.$$

- On a graph: for any function f on the graph, Δf is the function defined by:

$$\Delta f(x) = \sum_{x' \sim x} (f(x') - f(x))$$

Example



$$\Delta = \begin{pmatrix} -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 1 & 1 & -3 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$

Heat equation on a graph

- The heat equation is the same:

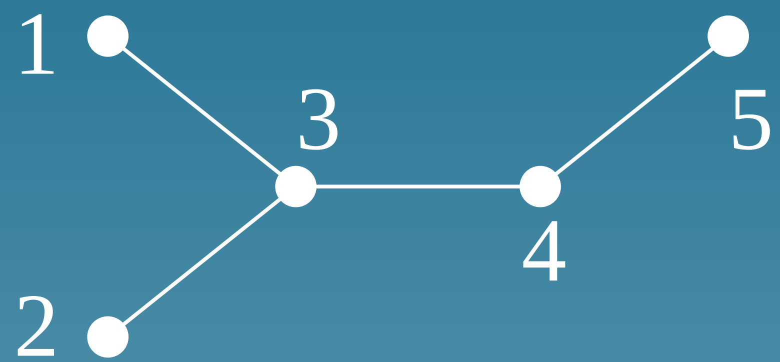
$$\frac{\partial K_x}{\partial t} = \Delta K_x.$$

- The solution is the heat kernel:

$$K(t) = \exp(t\Delta)$$

(Remember $e^A = Id + A + \frac{A^2}{2!} + \frac{A^3}{3!} + \ldots$)

Heat kernel example



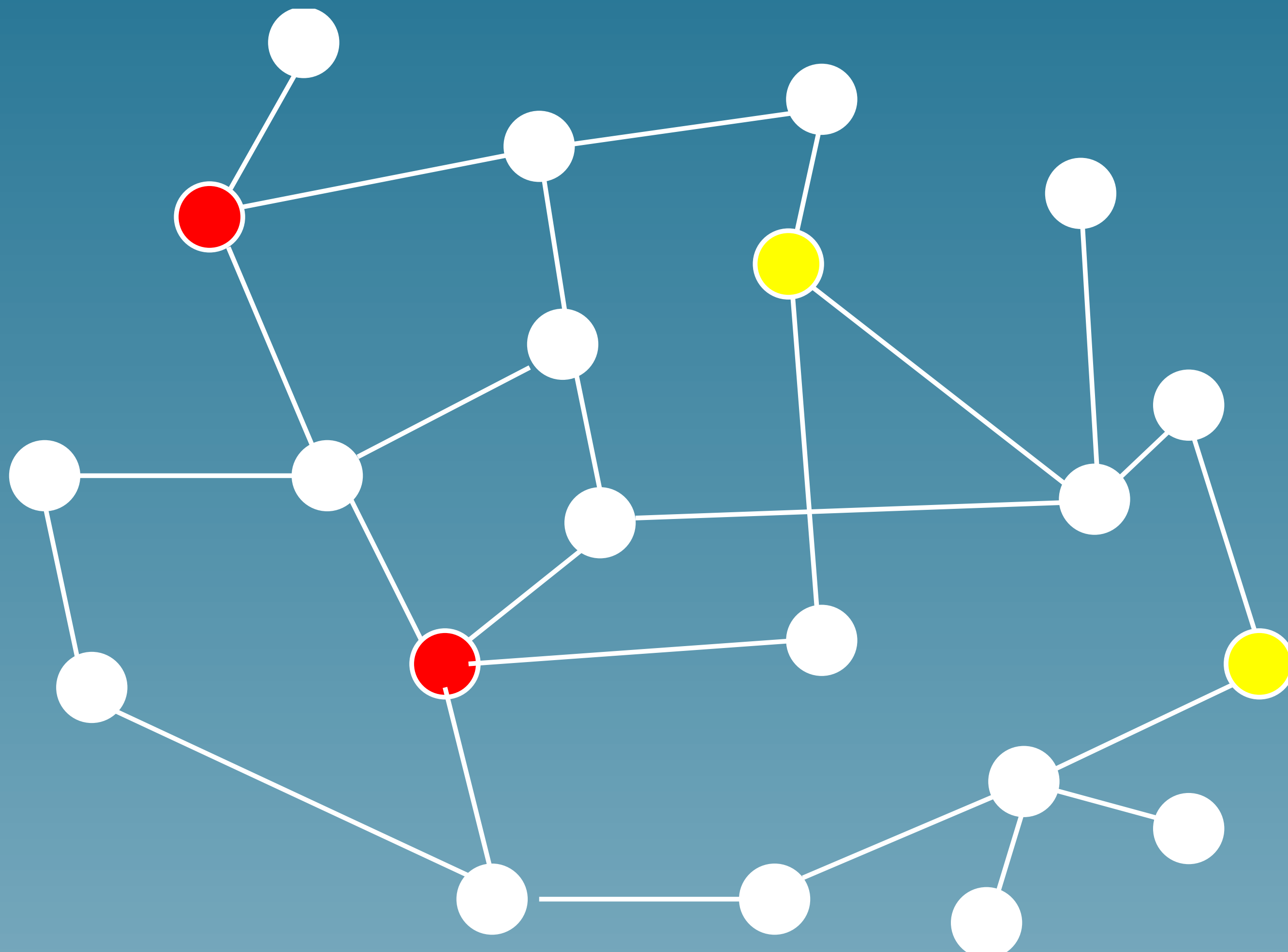
$$K = \exp(\Delta) = \begin{pmatrix} 0.49 & 0.12 & 0.23 & 0.10 & 0.03 \\ 0.12 & 0.49 & 0.23 & 0.10 & 0.03 \\ 0.23 & 0.23 & 0.24 & 0.17 & 0.10 \\ 0.10 & 0.10 & 0.17 & 0.31 & 0.30 \\ 0.03 & 0.03 & 0.10 & 0.30 & 0.52 \end{pmatrix}$$

Interpretation

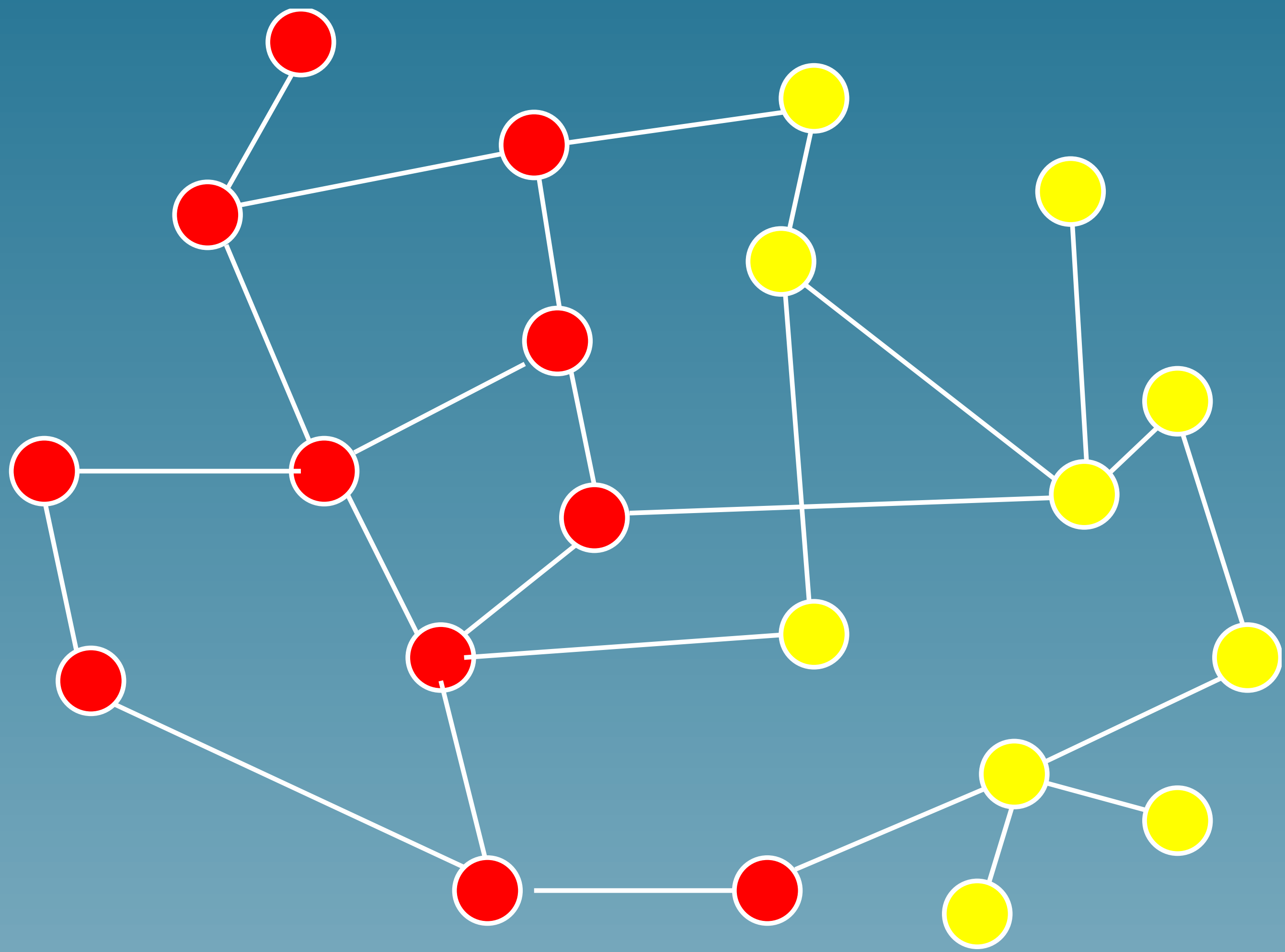
$$K_t(x,y) = \left[e^{t\Delta}\right]_{x,y}.$$

- a discrete version of the Gaussian
- is related to diffusions on the graph
- increases when there are many short paths between x and y

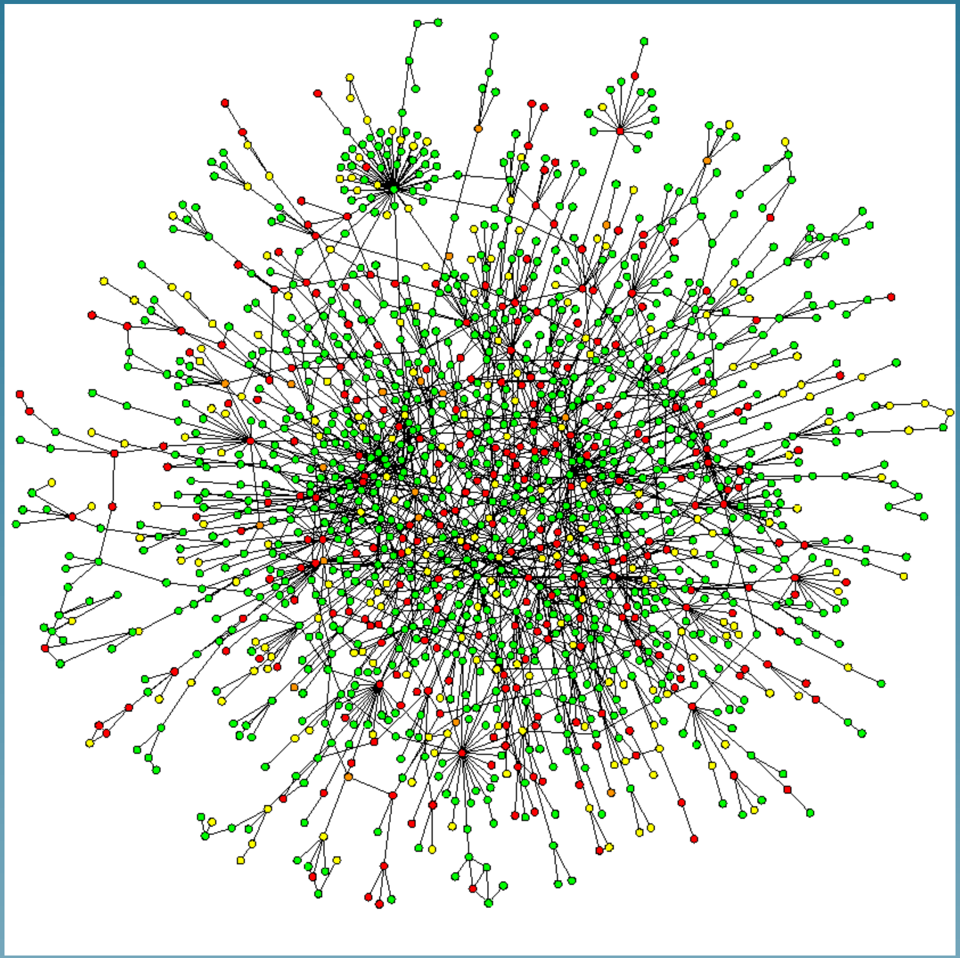
Inference on graphs



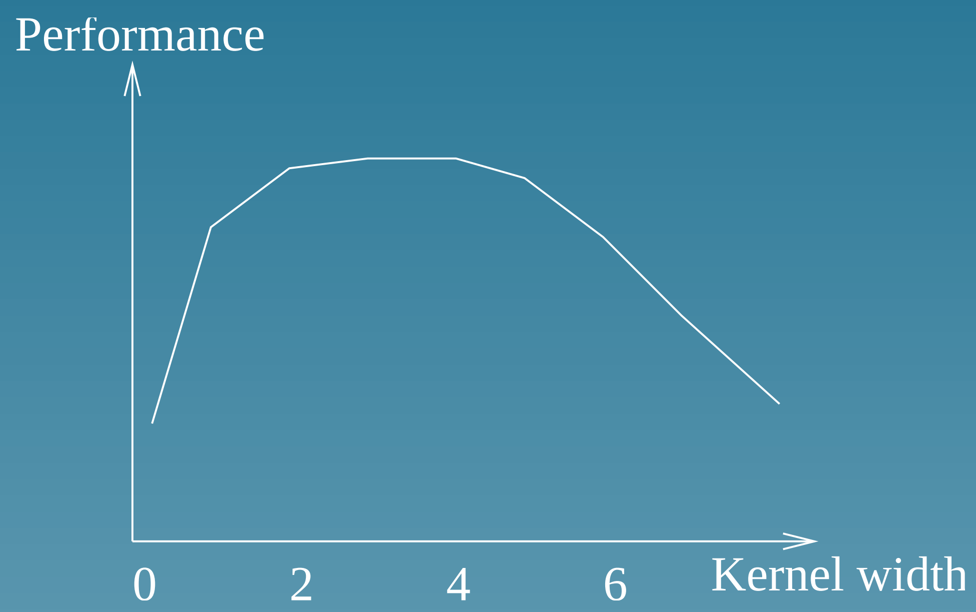
Inference on graphs



Example: protein function prediction



Example: protein function prediction



Conclusion

Conclusion

- SVM and kernel methods are powerful machine learning tools
- The kernel trick enables the use of SVM for nonvectorial data
- SVM on graph is possible and leads to good experimental results
- Applications in marketing?