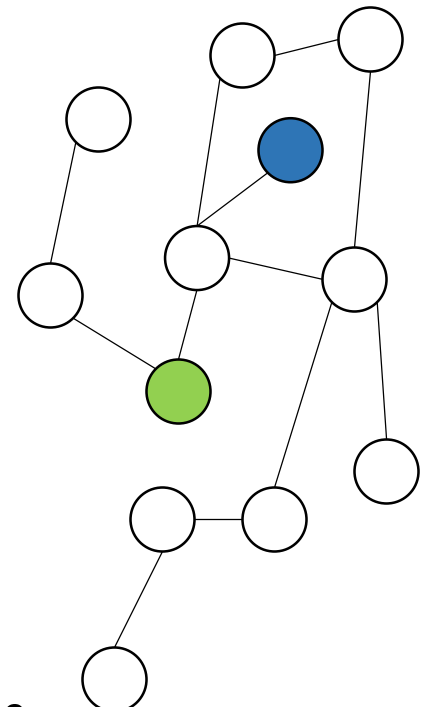
Mining Data Graphs

Semi-supervised learning, label propagation,

Web Search

Data graphs



- Data graphs are common in Web data
  - Web link graph
  - Chains of discussions

- It is also possible to create data graphs from Web data
  - Using similarity methods between data elements

- Graphs from Web data
  - The graph vertices are the elements we whish to analyse
  - The graph edges capture the level of affinity between two of such elements

However, in Web domain…

- **I have a good idea, but I can't afford to label lots of data!**
- **I have lots of labeled data, but I have even more unlabeled data**
  - **It's not just for small amounts of labeled data anymore!**

What is semi-supervised learning (SSL)?

- Labeled data (entity classification)

  - …, says Mr. **Cooper**, vice president of …
  - … **Firing Line** Inc., a **Philadelphia** gun shop.

<u>Labels</u>

person

location

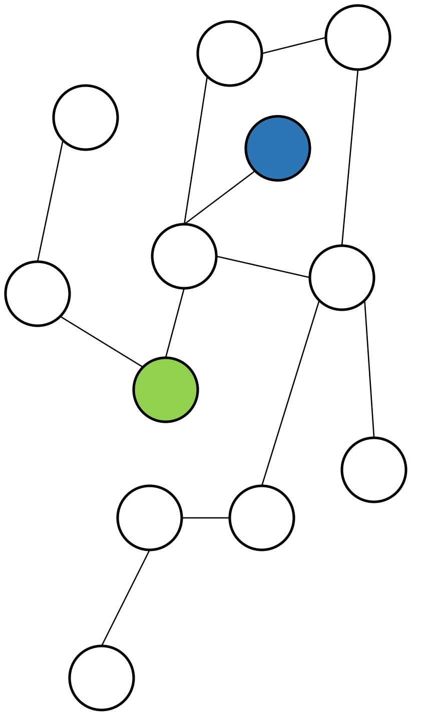
organization

- Lots more unlabeled data

  - …, Yahoo's own Jerry Yang is right …
  - … The details of Obama's San Francisco mis-adventure …

Graph-based semi-supervised Learning

- From items to graphs
- Basic graph-based algorithms
  - Mincut
  - Label propagation
  - Graph consistency



Text classification: easy example

- Two classes: astronomy vs. travel
- Document = 0-1 bag-of-word vector
- Cosine similarity

x1="bright asteroid", y1=astronomy
x2="yellowstone denali", y2=travel
x3="asteroid comet"?
x4="camp yellowstone"?

Easy, by word overlap

Hard example

x1="bright asteroid", y1=astronomy

x2="yellowstone denali", y2=travel

x3="zodiac"?

x4="airport bike"?

- No word overlap
- Zero cosine similarity
- Pretend you don't know English

Hard example

	x1	x3	x4	x2
asteroid	1			
bright	1			
comet				
zodiac		1		
airport			1	
bike			1	
yellowstone				1
denali				1

Unlabeled data comes to the rescue

	x1	x5	x6	x7	x3	x4	x8	x9	x2
asteroid	1								
bright	1	1	1						
comet		1	1	1					
zodiac				1	1				
airport						1			
bike						1	1	1	
yellowstone							1	1	1
denali								1	1

Intuition

1. Some **unlabeled documents** are similar to the **labeled documents** ➔ same label

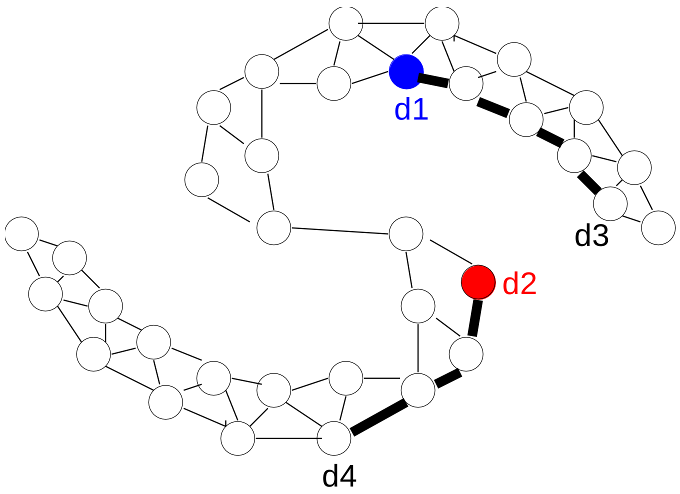
2. Some **other unlabeled documents** are similar to the above **unlabeled documents** ➔ same label

3. ad infinitum

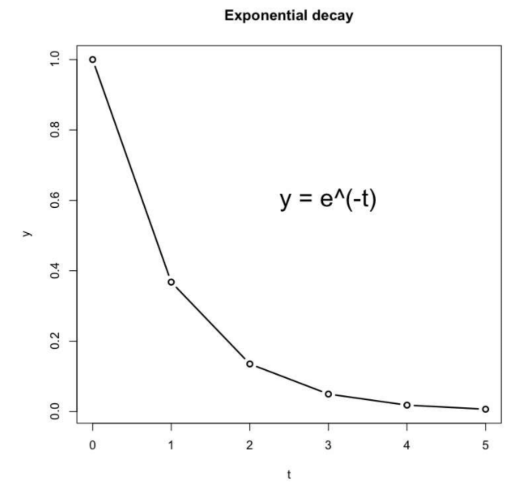
We will formalize this with graphs.

The graph

- Nodes $\{x_1, \ldots, x_l\} \cup \{x_{l+1}, \ldots, x_{m+l}\}$
- Weighted, undirected edges w_{ij}
  - Large weight ➔ similar x_i, x_j
- Known labels $y_1, \ldots, y_l$
- Want to know
  - transduction: $y_{l+1}, \ldots, y_{m+l}$
  - induction: y^* for new test item x^*



How to create a graph



1. Compute distance between i, j

$$w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

2. For each i, connect to its kNN. k very small but still connects the graph

3. Optionally put weights on (only) those edges

4. Tune σ

Mincut (s-t cut)

- Binary labels $y_i \in \{0,1\}$.
- Fix $Y_l = \{y_1, \dots, y_l\}$
- Solve for $Y_u = \{y_{l+1}, \dots, y_{l+m}\}$

$$\min_{Y_u} \sum_{i,j=1}^{n} w_{i,j} \left(y_i - y_j\right)^2$$

- Combinatorial problem (integer program) but efficient polynomial time solver (Boykov, Veksler, Sabih PAMI 2001).

Mincut example: Opinion detection

- **Task:** classify each sentence in a document into **objective**/**subjective**. (Pang,Lee. ACL 2004)

- NB/SVM for isolated classification
  - Subjective data (y=1): Movie review snippets "bold, imaginative, and impossible to resist"
  - Objective data (y=0): IMDB

Mincut example: Opinion detection

- Key observation: sentences next to each other tend to have the same label

$$w_{ij} = c \text{ if } x_i, x_j \text{ are close, } 0 \text{ otherwise.}$$

- Two special labeled nodes (source, sink)

$$(x_s, y_s = 1), (x_o, y_o = 0)$$

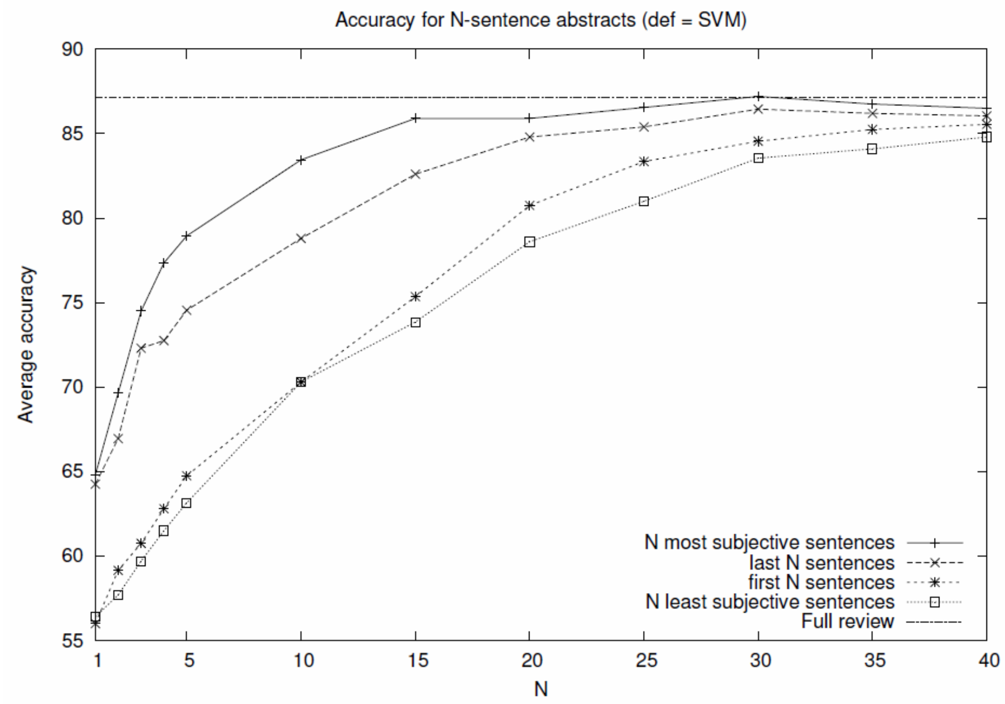
- Every sentence connects to both with different weight

$$w_{si} = Pr(y_i = 1 | x_i, NB)$$
$$w_{io} = Pr(y_i = 0 | x_i, NB)$$

Opinion detection

- Min cut classifies sentences as subjective vs objective.

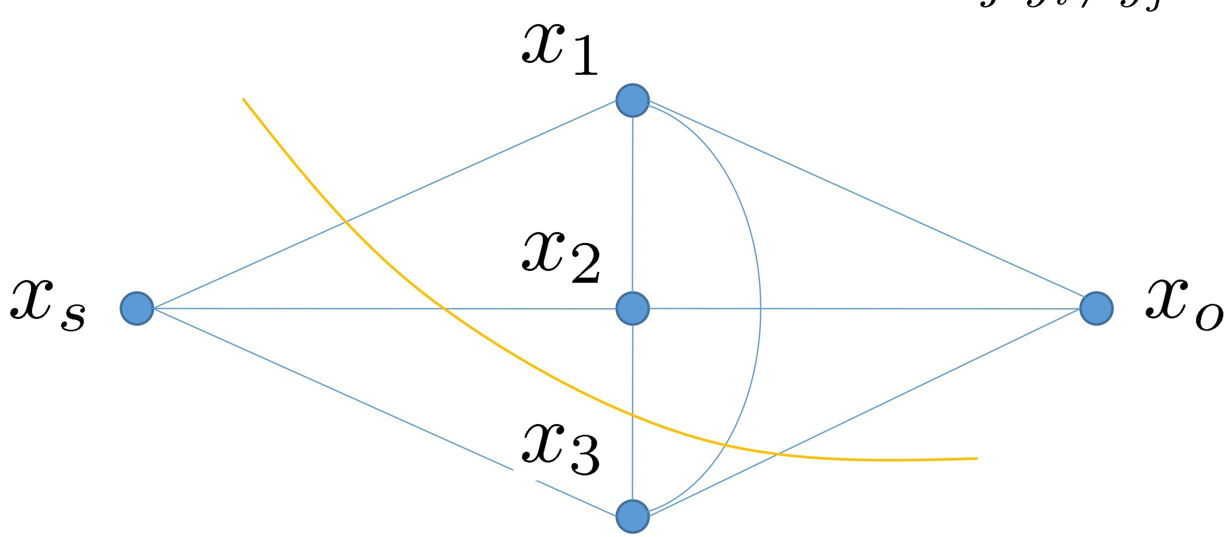
- Impact on the detection of opinion positive/negative:



Mincut example (s-t cut)

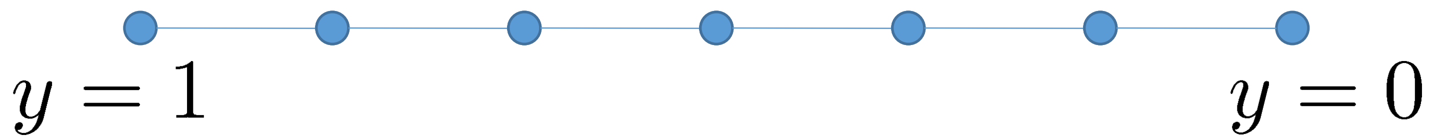
$$\min \sum_{ij} w_{ij}(y_i - y_j)^2 \text{ minimizes the cut}$$

$$\sum_{ij:y_i \neq y_j} w_{ij}$$



Some issues with mincut

- <u>Multiple equally min cuts, but different in practice:</u>



- Lacks classification confidence

- These are addressed by harmonic functions and label propagation

Relaxing mincut

- Labels are now real values in the interval [0,1]

$$f(x_l) = y_l$$

$$\min_{f_u} \sum_{i,j=1}^{n} w_{i,j}\left(f_i - f_j\right)^2$$

- Same as mincut except that $f_u \in R$

- $f_u \in [0,1]$ and is less confident near 0.5

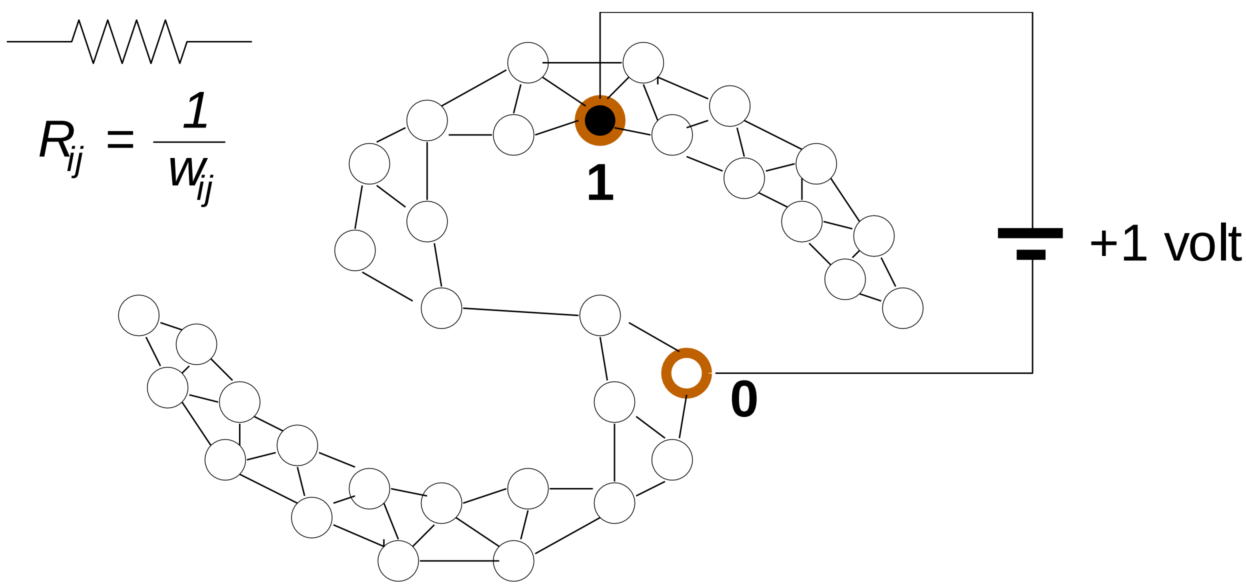
An electric network interpretation

Edges has conductance w_{ij}

1-volt battery connects to labeled points y_ℓ

Voltage at node $i = f_i$

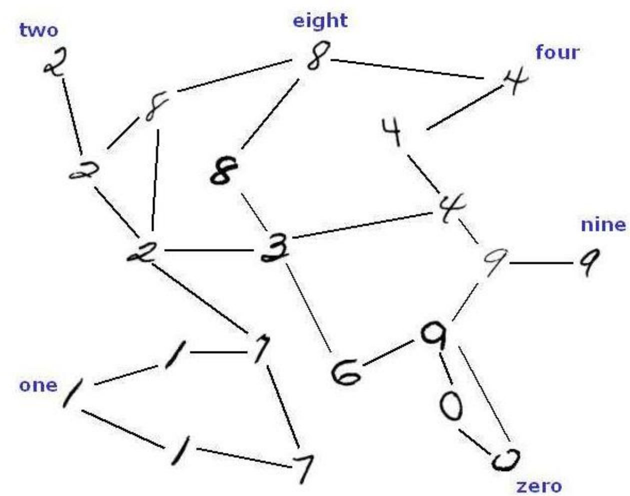
Similar voltage if many strong paths exist.



Label propagation

- Algorithm:

1. Set $f_u = 0$
2. Set $f_l = y_l$.
3. Propagate: $f_u = \frac{\sum_{k=1}^{n} w_{ku} \cdot f_k}{\sum_{k=1}^{n} w_{ku}}$.
4. Row normalize f
5. Repeat from step 2



Label propagation example: WSD

- Word sense disambiguation from context, e.g., "interest", "line" (Niu,Ji,Tan ACL 2005)
- x_i: context of the ambiguous word, features: POS, words, collocations
- d_{ij}: cosine similarity or JS-divergence
- w_{ij}: kNN graph
- Labeled data: a few x_i's are tagged with their word sense.

Label propagation example: WSD

- SENSEVAL-3, as percent labeled:

Percentage	SVM	LP_{cosine}	LP_{JS}
1%	24.9±2.7%	27.5±1.1%	28.1±1.1%
10%	53.4±1.1%	54.4±1.2%	54.9±1.1%
25%	62.3±0.7%	62.3±0.7%	63.3±0.9%
50%	66.6±0.5%	65.7±0.5%	66.9±0.6%
75%	68.7±0.4%	67.3±0.4%	68.7±0.3%
100%	69.7%	68.4%	70.3%

(Niu,Ji,Tan ACL 2005)

Graph consistency

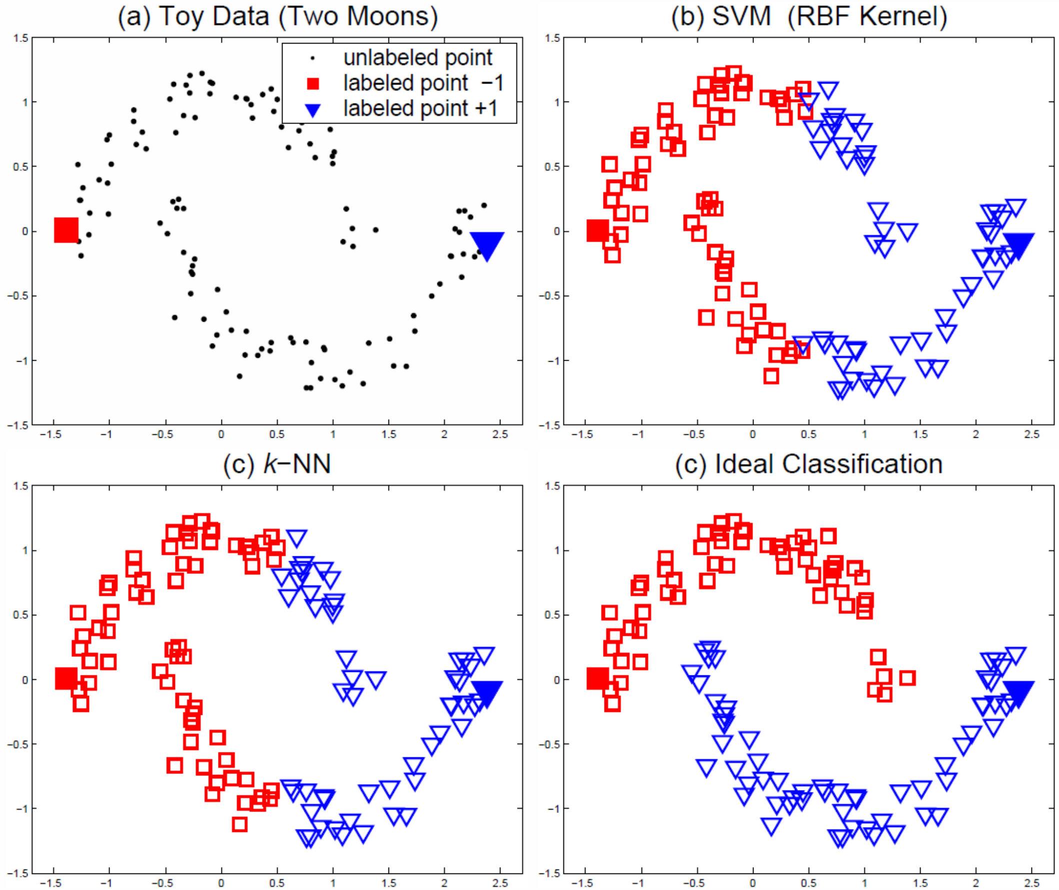
- The key to semi-supervised learning problems is the prior assumption of consistency:

    - **Local Consistency**: nearby points are likely to have the same label;

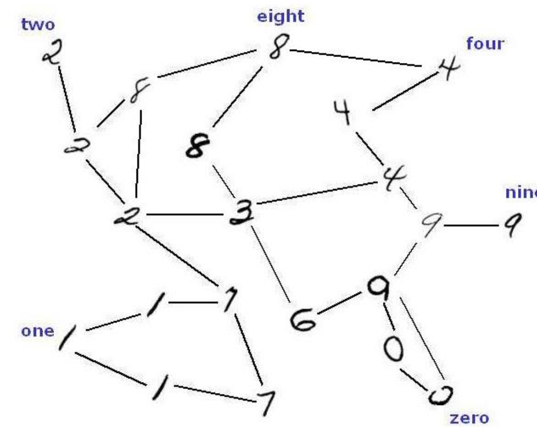
    - **Global Consistency**: Points on the same structure (cluster or manifold) are likely to have the same label;

Local and Global Consistency

- The key to the consistency algorithm is to **let every point iteratively spread its label information to its neighbors** until a global stable state is achieved.



Definitions



- Data points: $\{x_1, \ldots, x_l\} \cup \{x_{l+1}, \ldots, x_\mathrm{n}\}$
- Label set: $L = \{1, \ldots, c\}$
- Y is the initial classification on $\{x_1, \ldots, x_l\}$ with:

$$Y_{ij} = \begin{cases} 1, & if\ x_i\ is\ labeled\ as\ y_i = j \\ 0, & otherwise \end{cases}$$

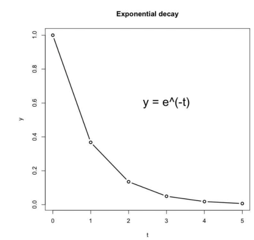
- F, a classification on x:

$$F_{n \times c} = \begin{bmatrix} F_{11} & \ldots & F_{1c} \\ \ldots & \ldots & \ldots \\ F_{n1} & \ldots & F_{nc} \end{bmatrix}$$

Labeling $\{x_{l+1}, \ldots, x_\mathrm{n}\}$ as $\mathrm{y_i} = \mathrm{argmax}_{j \leq c} F_{ij}^{*}$

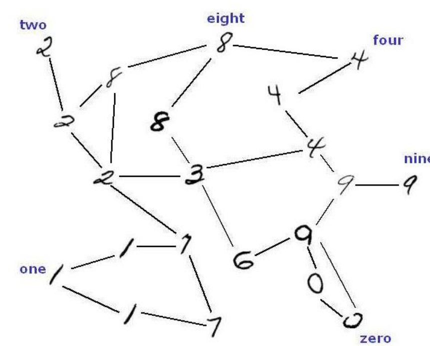
Consistency algorithm: the graph

1. Construct the affinity matrix W defined by a Gaussian kernel:



$$w_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) & , if\ i \neq j \\ 0 & if\ i = j \end{cases}$$

2. Normalize W symmetrically by



$$S = D^{-1/2} W D^{-1/2}$$

where D is a diagonal matrix with $D_{ii} = \sum_k w_{ik}$

Consistency algorithm: the propagation

3. Iterate until convergence:

$$F(t+1) = \alpha \cdot S \cdot F(t) + (1-\alpha) \cdot Y$$

- **First term**: each point receive information from its neighbors.
- **Second term**: retains the initial information.
- Normalize F on each iteration.

4. Let F^* denote the limit of the sequence {F(t)}.
The classification results are:

Labeling $\mathrm{x_i}$ as $\mathrm{y_i} = \mathrm{argmax}_{j \leq c} F^*_{ij}$

Closed-form solution

- From the iteration equation, we can show that:

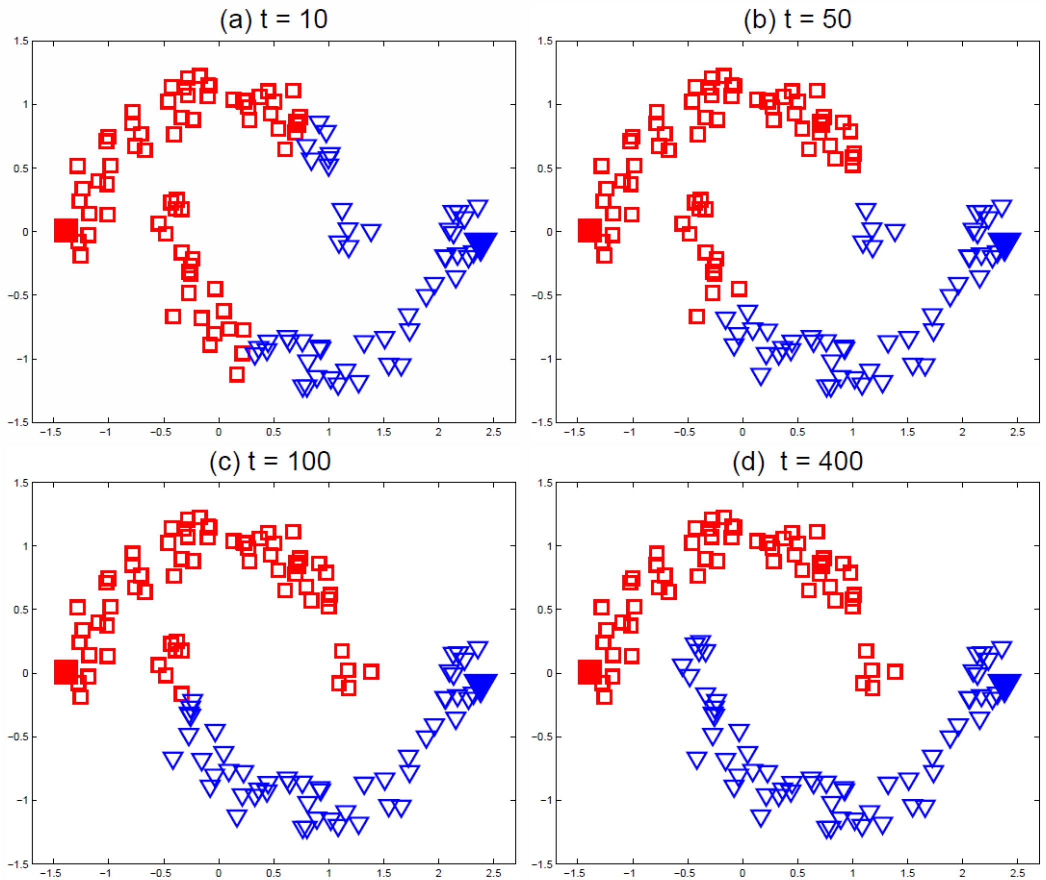
$$F^* = \lim_{t \to \infty} F(t) = (I - \alpha S)^{-1} \cdot \mathrm{Y}$$

- So we could compute F* directly without iterations.

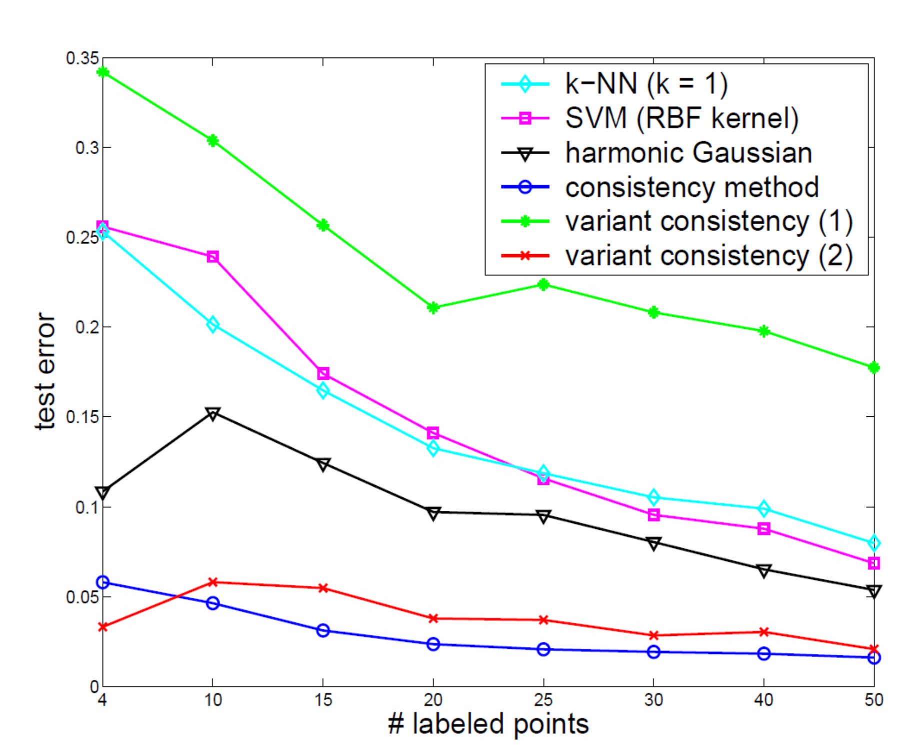
- The closed-form may be too complex to calculate for very large graphs (the matrix inversion step)

The convergence process

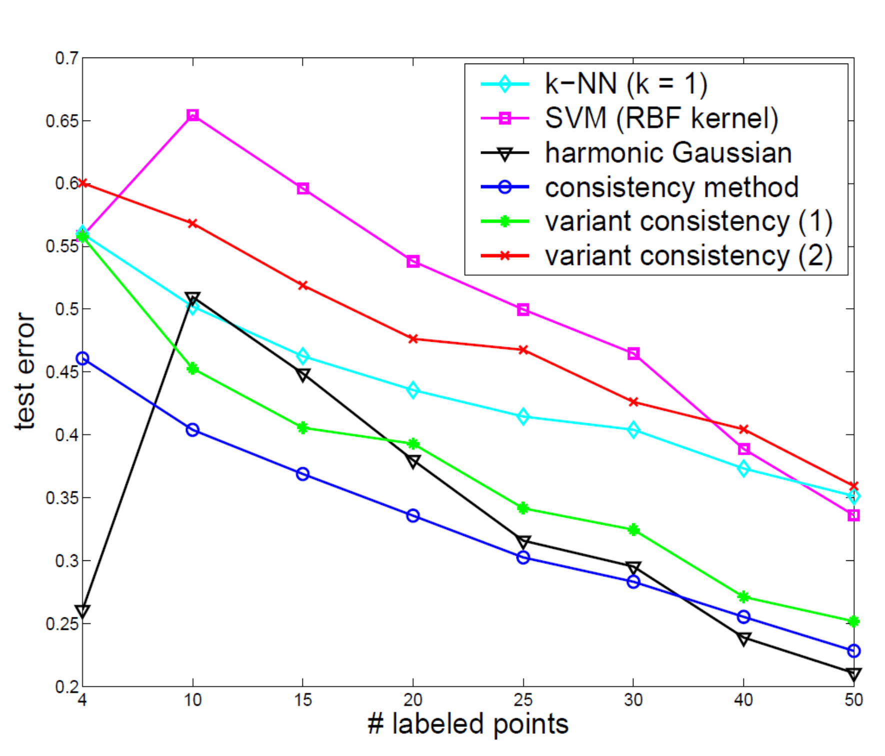
- The initial label information are diffused along the two moons.



Experimental Results



Digit recognition: digit 1-4 from the USPS data set



Text classification: topics including autos, motorcycles, baseball and hockey from the 20-newsgroups

Caution

- Advantages of graph-based methods:
  - Clear intuition, elegant math
  - Performs well if the graph fits the task
- Disadvantages:
  - Performs poorly if the graph is bad: sensitive to graph structure and edge weights
  - Usually we do not know which will happen!

Conclusions

- The key to semi-supervised learning problem is the consistency assumption.

- The consistency algorithm proposed was demonstrated effective on the data set considered.