Semi-supervised Learning

Introduction

- Supervised learning: $\{(x^r, \hat{y}^r)\}_{r=1}^{R}$
  - E.g.x^r: image,$\hat{y}^r$: class labels
- Semi-supervised learning: $\{(x^r, \hat{y}^r)\}_{r=1}^{R}$, $\{x^u\}_{u=R}^{R+U}$
  - A set of unlabeled data, usually U >> R
  - Transductive learning: unlabeled data is the testing data
  - Inductive learning: unlabeled data is not the testing data
- Why semi-supervised learning?
  - Collecting data is easy, but collecting “labelled” data is expensive
  - We do semi-supervised learning in our lives

Why semi-supervised learning helps?

Labelled data


cat

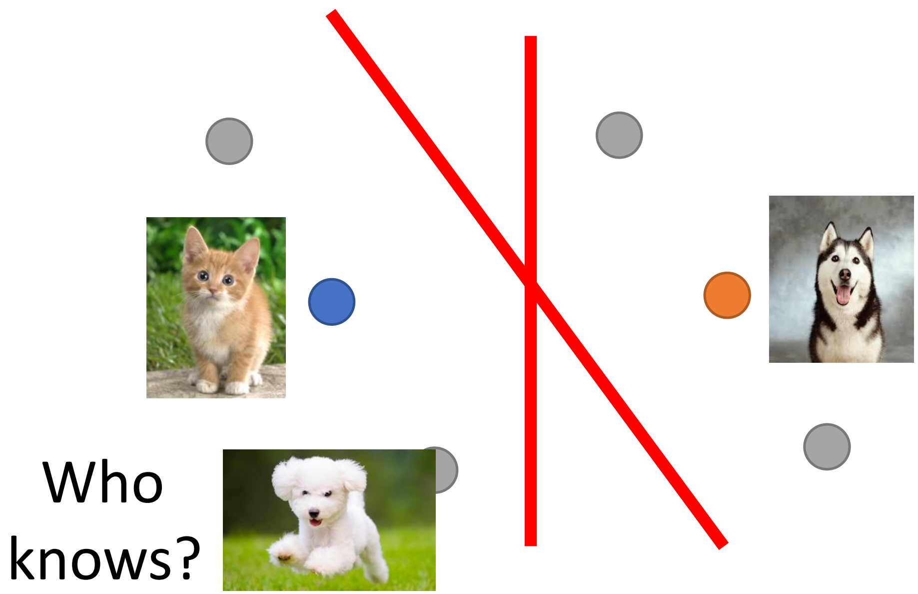

dog

Unlabeled data




(Image of cats and dogs without labeling)

Why semi-supervised learning helps?



The distribution of the unlabeled data tell us ***something***.

Usually with some assumptions

Outline

Semi-supervised Learning for Generative Model

Low-density Separation Assumption

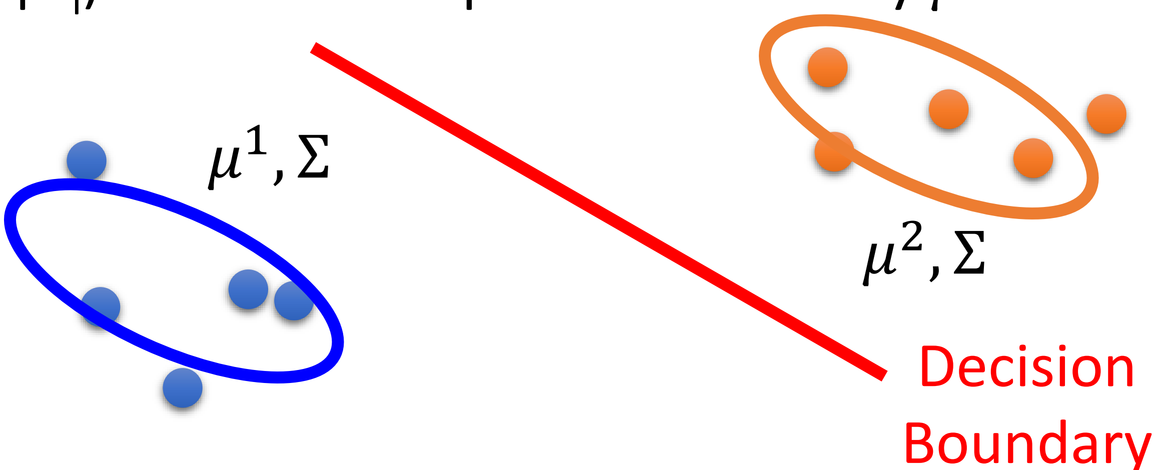
Smoothness Assumption

Better Representation

Semi-supervised Learning for Generative Model

Supervised Generative Model

- Given labelled training examples $x^r \in C_1, C_2$
  - looking for most likely prior probability $P(C_i)$ and class-dependent probability $P(x|C_i)$
  - $P(x|C_i)$ is a Gaussian parameterized by μ^i and Σ

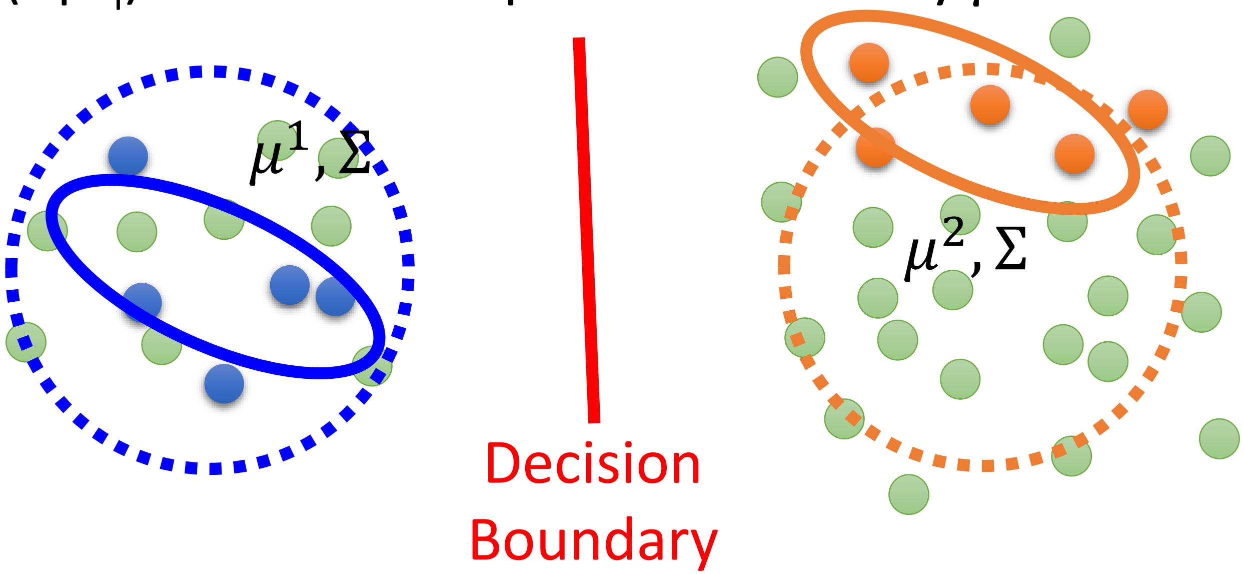


With $P(C_1)$, $P(C_2)$,μ^1,μ^2, Σ

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

Semi-supervised Generative Model

- Given labelled training examples $x^r \in C_1, C_2$
  - looking for most likely prior probability $P(C_i)$ and class-dependent probability $P(x|C_i)$
  - $P(x|C_i)$ is a Gaussian parameterized by μ^i and Σ



The unlabeled data x^u help re-estimate $P(C_1)$, $P(C_2)$, μ^1,μ^2, Σ

Semi-supervised Generative Model

The algorithm converges eventually, but the initialization influences the results.

- Initialization: $\theta = \{P(C_1), P(C_2), \mu^1, \mu^2, \Sigma\}$

E

- Step 1: compute the posterior probability of unlabeled data

$$P_\theta(C_1|x^u)$$

Depending on model θ

Back to step 1

M

- Step 2: update model

$$P(C_1) = \frac{N_1 + \sum_{x^u} P(C_1|x^u)}{N}$$

N: total number of examples
N_1: number of examples belonging to C_1

$$\mu^1 = \frac{1}{N_1} \sum_{x^r \in C_1} x^r + \frac{1}{\sum_{x^u} P(C_1|x^u)} \sum_{x^u} P(C_1|x^u) x^u \quad \ldots\ldots$$

Why?

$$\theta = \{P(C_1), P(C_2), \mu^1, \mu^2, \Sigma\}$$

- Maximum likelihood with labelled data

Closed-form solution

$$logL(\theta) = \sum_{x^r} logP_\theta(x^r, \hat{y}^r)$$

$$\boxed{\begin{aligned} &P_\theta(x^r, \hat{y}^r) \\ &= P_\theta(x^r|\hat{y}^r)P(\hat{y}^r) \end{aligned}}$$

- Maximum likelihood with labelled + unlabeled data

$$logL(\theta) = \sum_{x^r} logP_\theta(x^r, \hat{y}^r) + \sum_{x^u} logP_\theta(x^u)$$

Solved iteratively

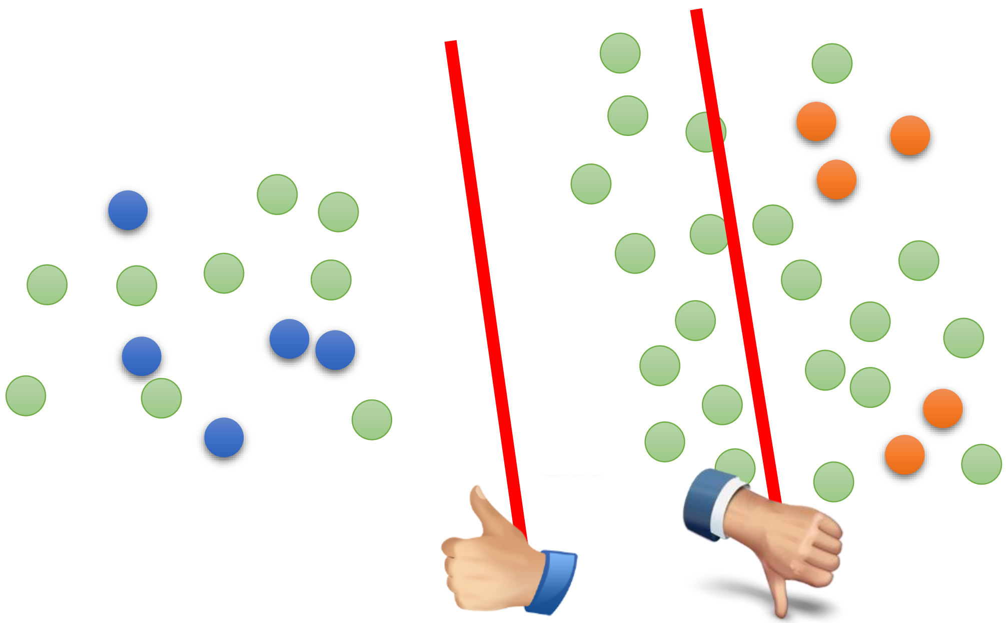
$$\boxed{P_\theta(x^u) = P_\theta(x^u|C_1)P(C_1) + P_\theta(x^u|C_2)P(C_2)}$$

(x^u can come from either C_1 and C_2)

Semi-supervised Learning
Low-density Separation

非黑即白

“Black-or-white”



Self-training

- Given: labelled data set = $\{(x^r, \hat{y}^r)\}_{r=1}^{R}$, unlabeled data set = $\{x^u\}_{u=l}^{R+U}$
- Repeat:
  - Train model f^* from labelled data set

No effect

Independent to the model

Regression?

  - Apply f^* to the unlabeled data set
    - Obtain $\{(x^u, y^u)\}_{u=l}^{R+U}$

Pseudo-label

  - Remove a set of data from unlabeled data set, and add them into the labeled data set

How to choose the data set remains open

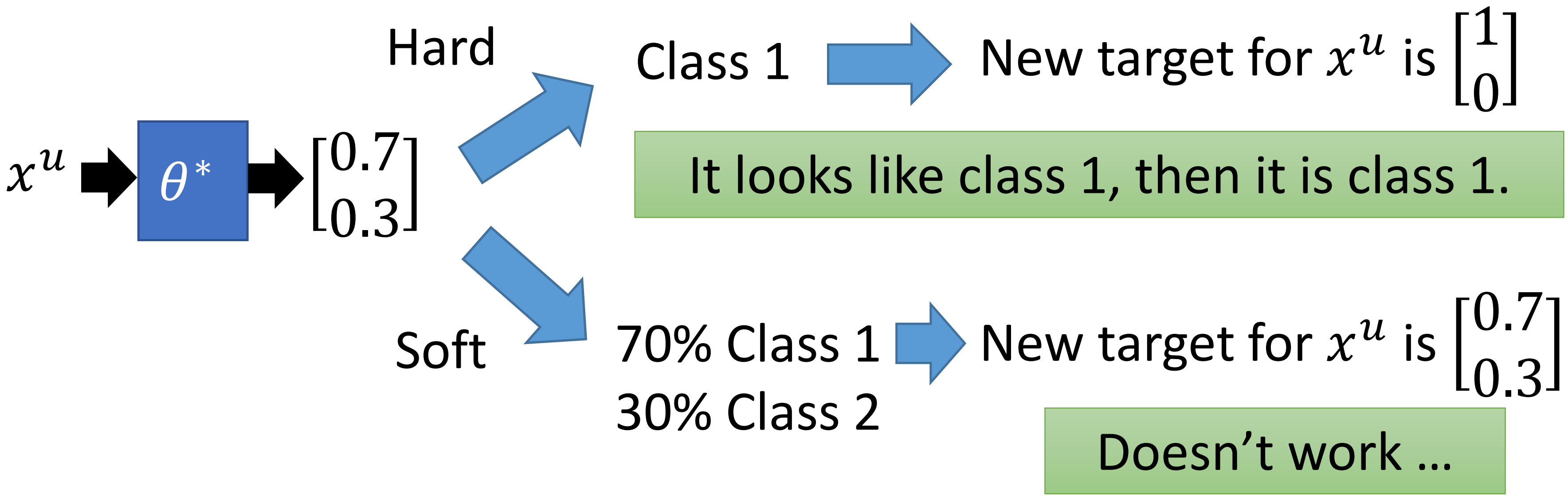
You can also provide a weight to each data.

Self-training

- Similar to semi-supervised learning for generative model
- Hard label v.s. Soft label

Considering using neural network

θ^* (network parameter) from labelled data



Entropy-based Regularization

$$x^u \rightarrow \boxed{\theta^*} \rightarrow y^u$$

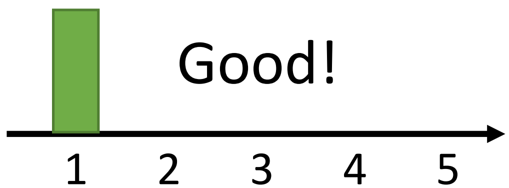
Distribution

Entropy of y^u :
Evaluate how concentrate the distribution y^u is

$$E(y^u) = -\sum_{m=1}^{5} y_m^u ln(y_m^u)$$

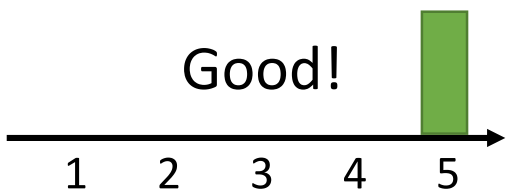
As small as possible

y^u Good!



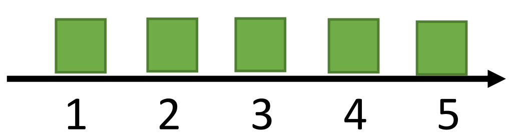
$$E(y^u) = 0$$

y^u Good!



$$E(y^u) = 0$$

y^u Bad!



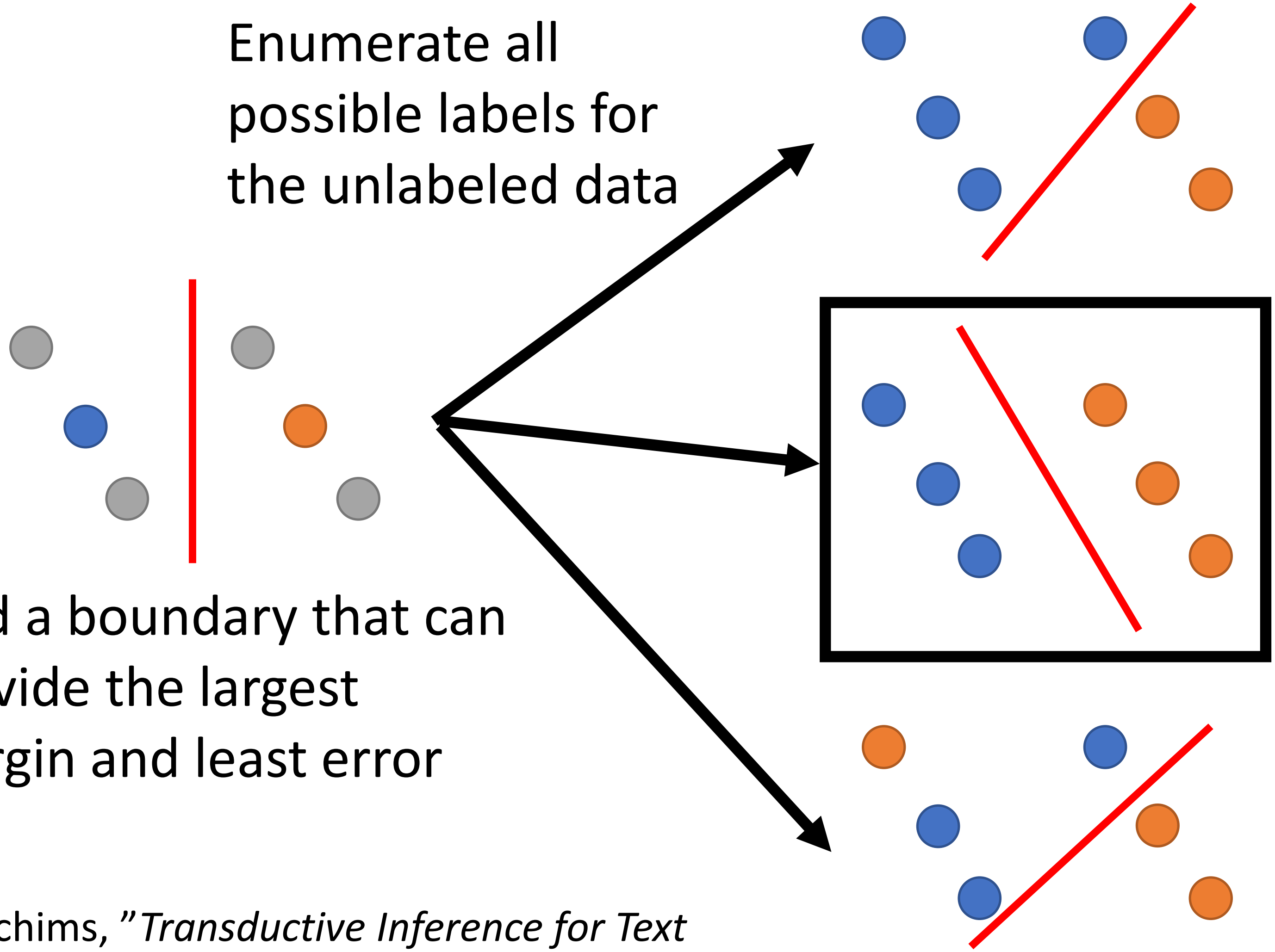
$$E(y^u) = -ln\left(\frac{1}{5}\right) = ln5$$

$$L = \sum_{x^r} C(y^r, \hat{y}^r)$$ labelled data

$$+\lambda \sum_{x^u} E(y^u)$$ unlabeled data

Outlook: Semi-supervised SVM

Enumerate all possible labels for the unlabeled data



Find a boundary that can provide the largest margin and least error

Thorsten Joachims, *"Transductive Inference for Text Classification using Support Vector Machines"*, ICML, 1999

Semi-supervised Learning
Smoothness Assumption

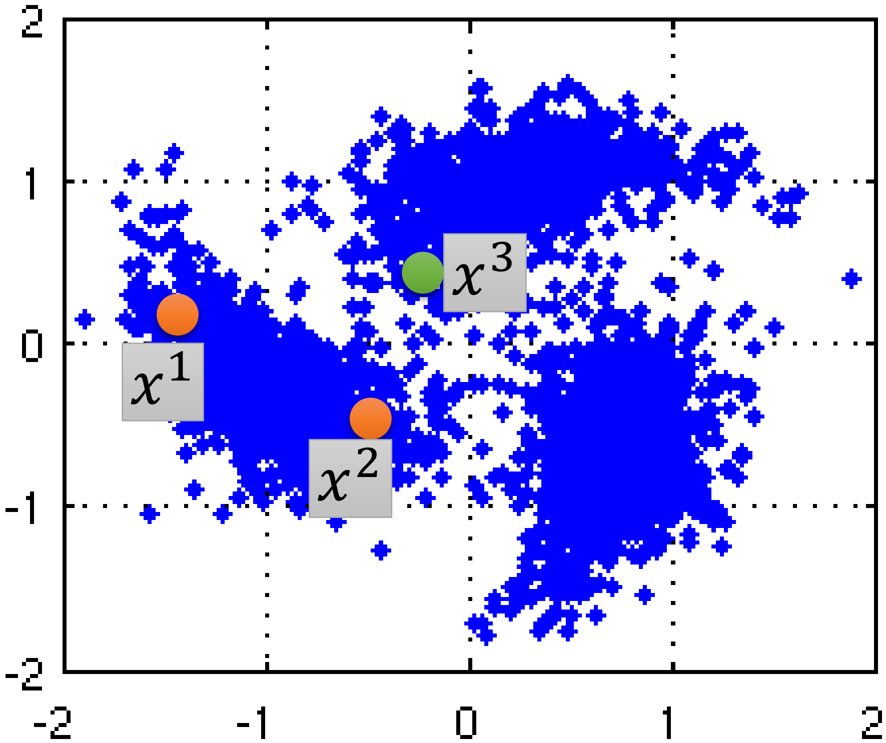
近朱者赤，近墨者黑

"You are known by the company you keep"

Smoothness Assumption

- Assumption: "similar" x has the same $\hat{y}$
- More precisely:
  - x is not uniform.
  - If x^1 and x^2 are close in ~~a high density region~~, $\hat{y}^1$ and $\hat{y}^2$ are the same.

connected by a high density path

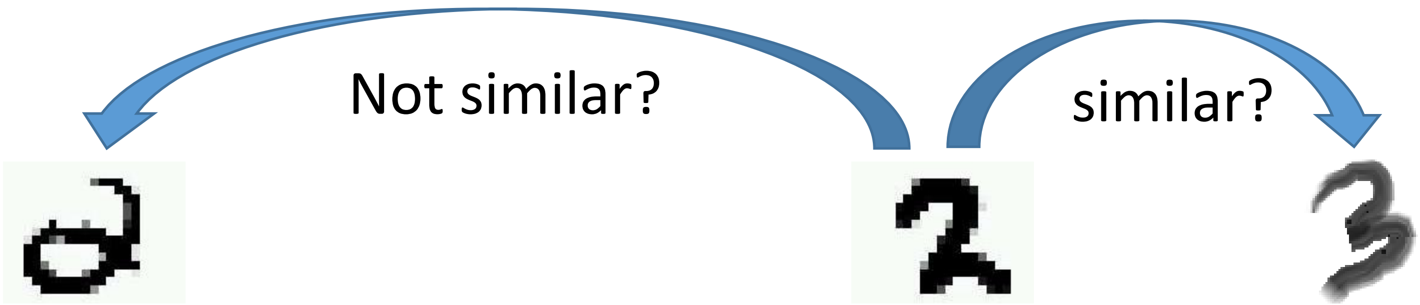


x^1 and x^2 have the same label

x^2 and x^3 have different labels

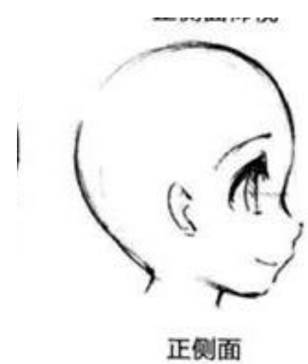
Source of image: http://hips.seas.harvard.edu/files/pinwheel.png

Smoothness Assumption



“indirectly” similar
with stepping stones

(The example is from the tutorial slides of Xiaojin Zhu.)



Source of image: http://www.moehui.com/5833.html/5/

Smoothness Assumption

- Classify astronomy vs. travel articles

	d_1	d_3	d_4	d_2
asteroid	●	●		
bright	●	●		
comet		●		
year				
zodiac				
⋮				
airport				
bike				
camp			●	
yellowstone			●	●
zion				●

	d_1	d_3	d_4	d_2
asteroid	●			
bright	●			
comet				
year				
zodiac		●		
⋮				
airport			●	
bike			●	
camp				
yellowstone				●
zion				●

(The example is from the tutorial slides of Xiaojin Zhu.)

Smoothness Assumption

- Classify astronomy vs. travel articles

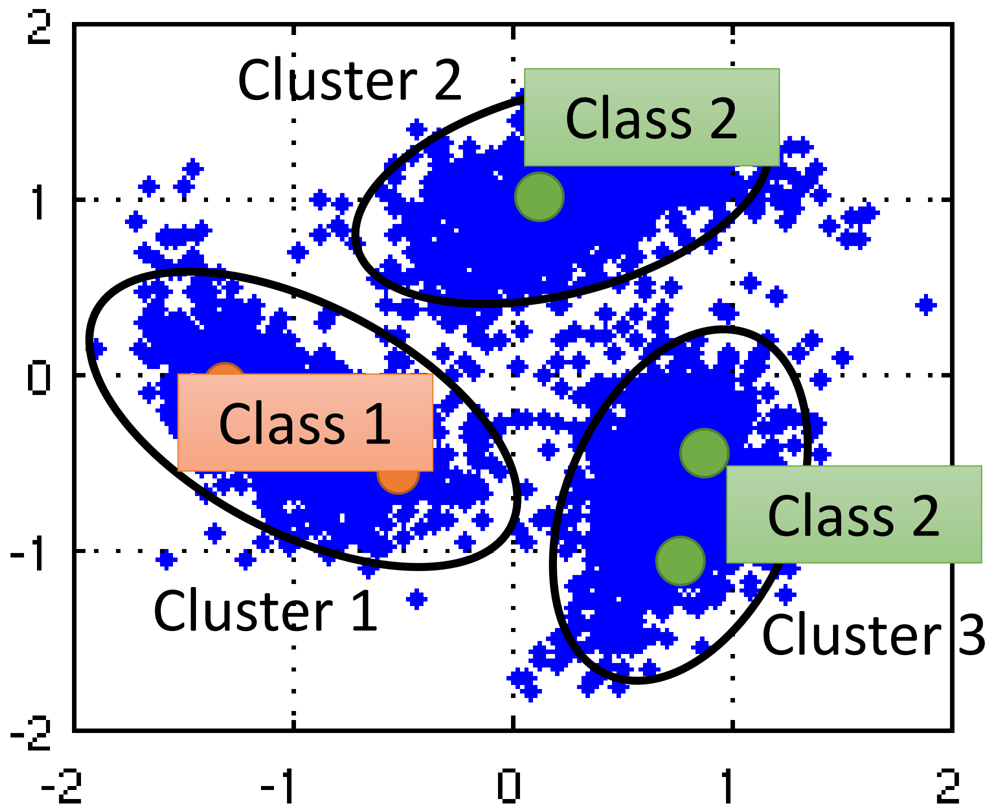
	d_1	d_5	d_6	d_7	d_3	d_4	d_8	d_9	d_2
asteroid	●								
bright	●	●							
comet		●	●						
year			●	●					
zodiac				●	●				
⋮									
airport						●			
bike						●	●		
camp							●	●	
yellowstone								●	●
zion									●

(The example is from the tutorial slides of Xiaojin Zhu.)

Cluster and then Label

Class 1

Class 2



Using all the data to learn a classifier as usual

Graph-based Approach

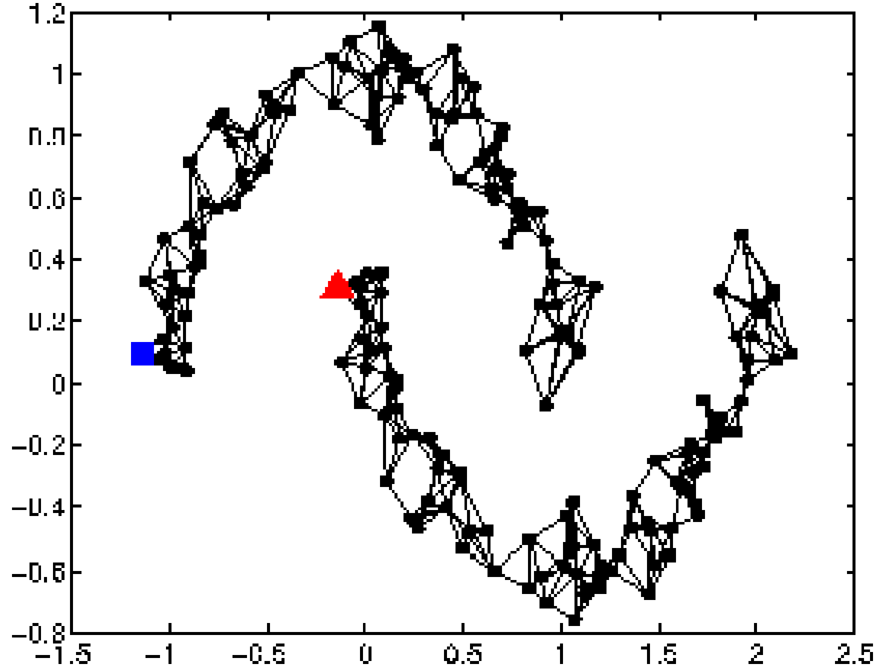
- How to know x^1 and x^2 are close in a high density region (connected by a high density path)

Represented the data points as a ***graph***

Graph representation is nature sometimes.

E.g. Hyperlink of webpages, citation of papers

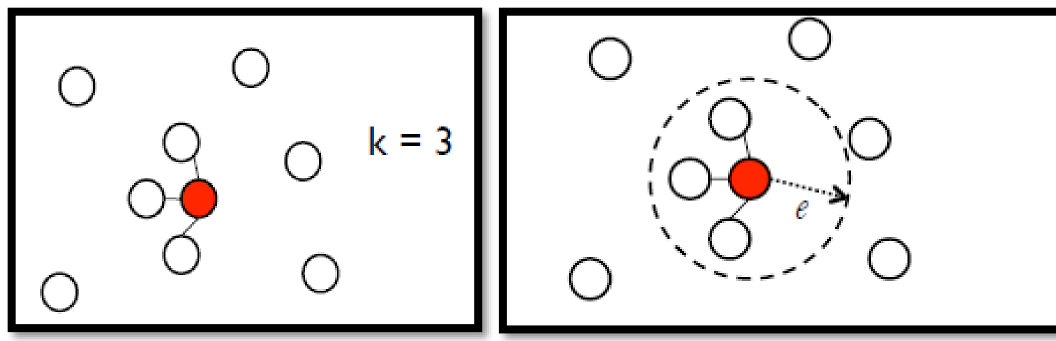
Sometimes you have to construct the graph yourself.



Graph-based Approach - Graph Construction

The image is from the tutorial slides of Amarnag Subramanya and Partha Pratim Talukdar

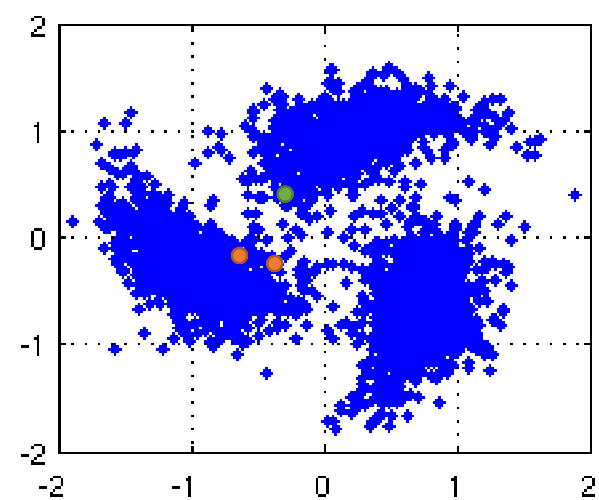
- Define the similarity $s(x^i, x^j)$ between x^i and x^j
- Add edge:
  - K Nearest Neighbor
  - *e*-Neighborhood



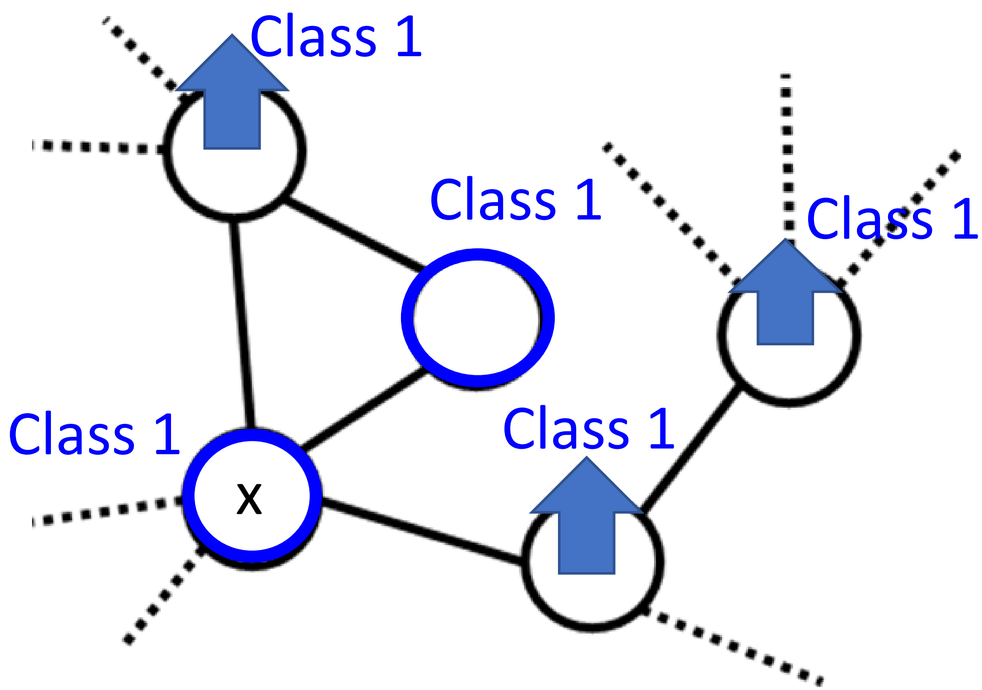
- Edge weight is proportional to s(x^i, x^j)

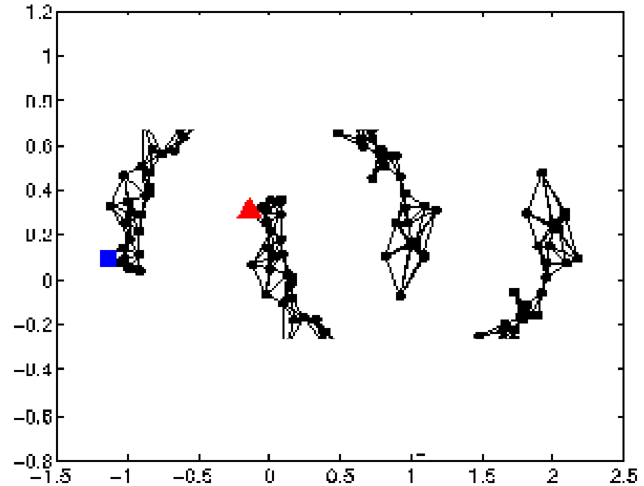
Gaussian Radial Basis Function:

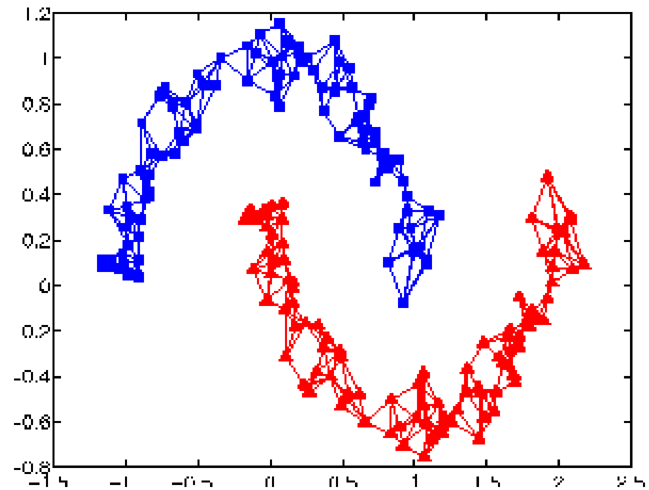
$$s(x^i, x^j) = exp\left(-\gamma \left\| x^i - x^j \right\|^2\right)$$



Graph-based Approach







The labelled data influence their neighbors.

Propagate through the graph

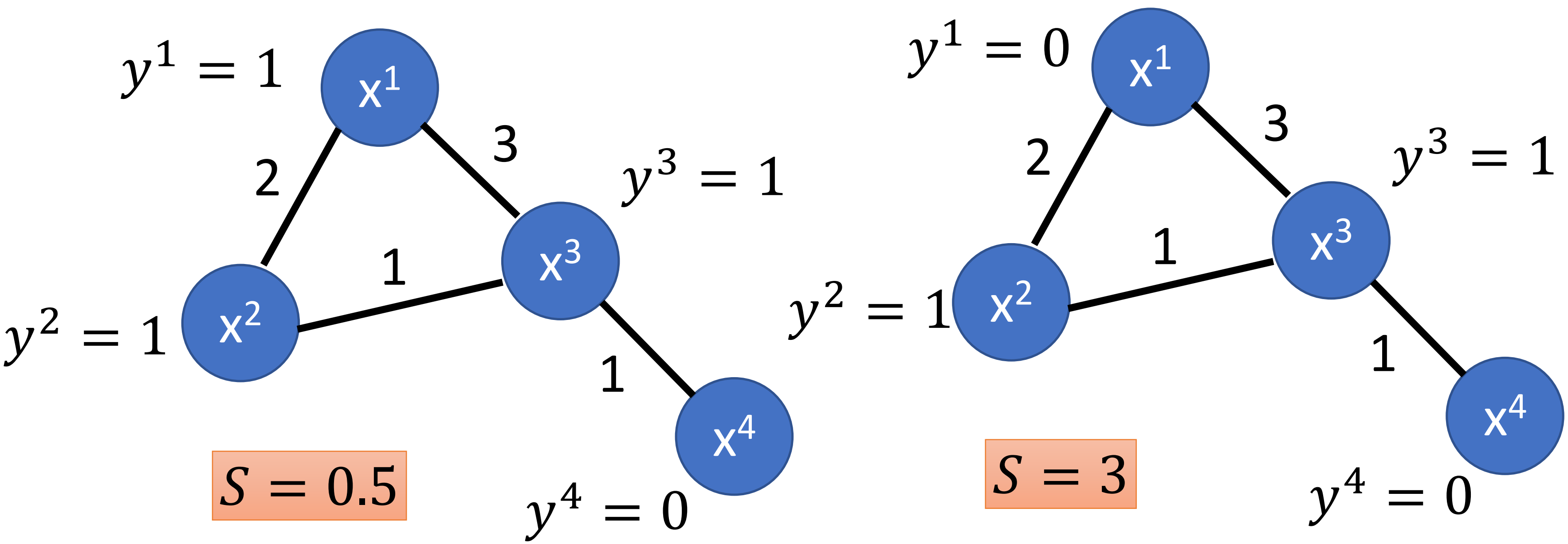
Graph-based Approach

- Define the smoothness of the labels on the graph

$$S = \frac{1}{2}\sum_{i,j} w_{i,j}\left(y^i - y^j\right)^2$$

Smaller means smoother

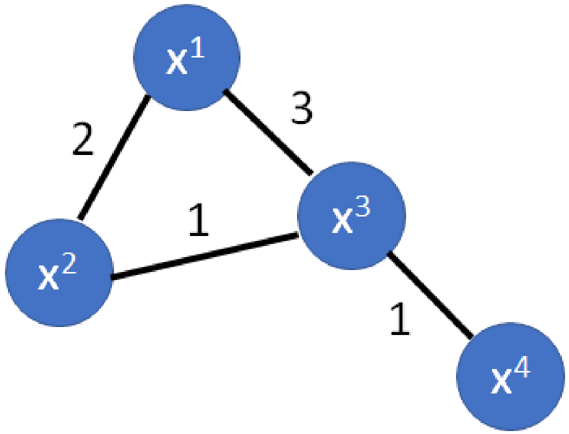
For all data (no matter labelled or not)



Graph-based Approach

- Define the smoothness of the labels on the graph

$$S = \frac{1}{2}\sum_{i,j} w_{i,j}\left(y^i - y^j\right)^2 = \boldsymbol{y}^T L \boldsymbol{y}$$



y: (R+U)-dim vector

$$\boldsymbol{y} = \left[\cdots y^i \cdots y^j \cdots\right]^{\boldsymbol{T}}$$

L: (R+U) x (R+U) matrix

Graph Laplacian

$$L = \underline{D} - \underline{W}$$

$$W = \begin{bmatrix} 0 & 2 & 3 & 0 \\ 2 & 0 & 1 & 0 \\ 3 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} \boxed{5} & 0 & 0 & 0 \\ 0 & \boxed{3} & 0 & 0 \\ 0 & 0 & \boxed{5} & 0 \\ 0 & 0 & 0 & \boxed{1} \end{bmatrix}$$

Graph-based Approach

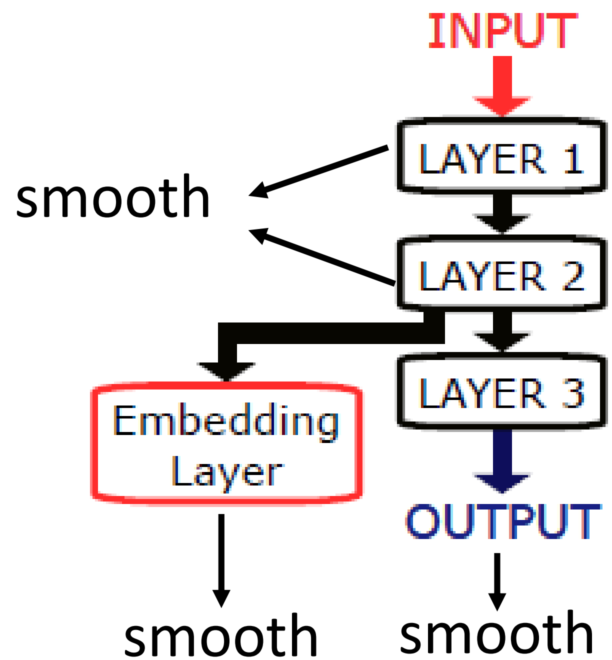
- Define the smoothness of the labels on the graph

$$S = \frac{1}{2}\sum_{i,j} w_{i,j}\left(y^i - y^j\right)^2 = \boldsymbol{y}^T L \boldsymbol{y}$$

Depending on network parameters

$$L = \sum_{x^r} C(y^r, \hat{y}^r) \boxed{+\lambda S}$$

As a regularization term



J. Weston, F. Ratle, and R. Collobert, "Deep learning via semi-supervised embedding," ICML, 2008

Semi-supervised Learning
Better Representation

去蕪存菁，化繁為簡

Looking for Better Representation

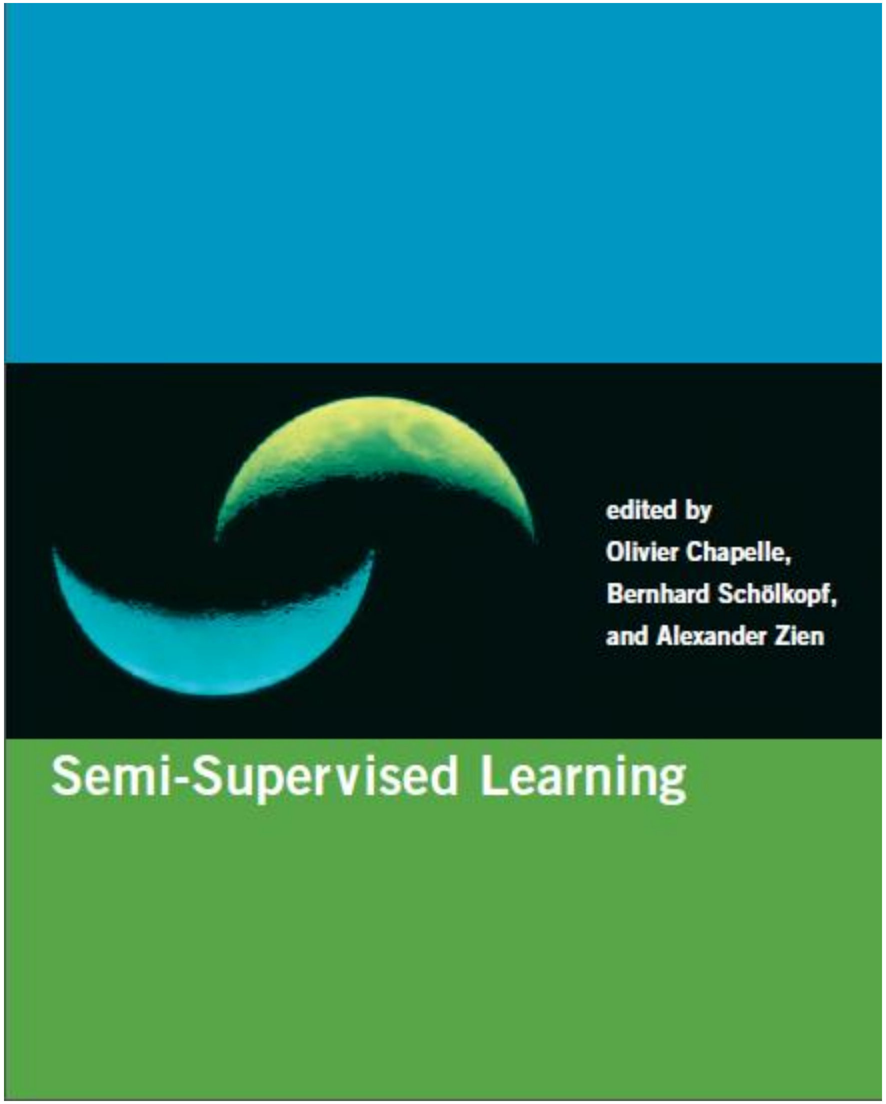
- Find the latent factors behind the observation
- The latent factors (usually simpler) are better representations

observation

Better representation
(Latent factor)

(In unsupervised learning part)

Reference



http://olivier.chapelle.cc/ssl-book/

Acknowledgement

- 感謝 劉議隆 同學指出投影片 上的錯字
- 感謝 丁勃雄 同學指出投影片 上的錯字