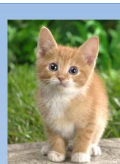
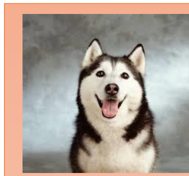
# Semi-supervised Learning

# Introduction

Labelled data



cat



dog

Unlabeled data

(Image of cats and dogs without labeling)

- 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

# Outline

Semi-supervised Learning for Generative Model

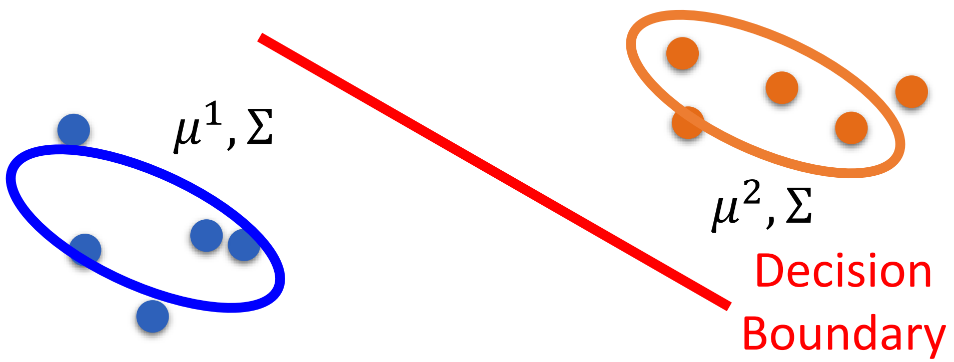
Low-density Separation Assumption

Smoothness Assumption

Better Representation

# 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 $\mu^i$ and $\Sigma$




With $P(C_1), P(C_2), \mu^1, \mu^2, \Sigma$ $\quad P(C_1|x) = \dfrac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$

The unlabeled data $x^u$ help re-estimate $P(C_1)$, $P(C_2)$, $\mu^1, \mu^2, \Sigma$

# Semi-supervised Generative Model

- 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 $\theta$

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

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

- Maximum likelihood with labelled + unlabeled data

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

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 $C_2$)

# 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 = $\{x^u\}_{u=1}^{U}$
- Repeat:
  - Train model $f^*$ from labelled data set

    You can use any model here.

    Regression?

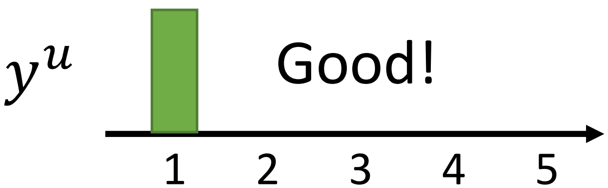
  - Apply $f^*$ to the unlabeled data set
    - Obtain $\{(x^u, y^u)\}_{u=1}^{U}$ Pseudo-label
  - Remove a set of data from unlabeled data set, and add them into the labeled data set

    How to choose the data set remains open

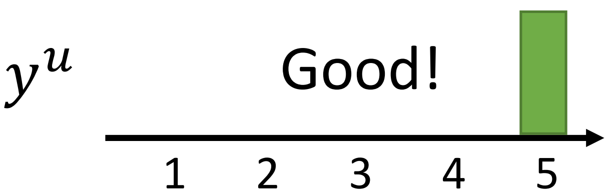
    You can also provide a weight to each data.

# Entropy-based Regularization

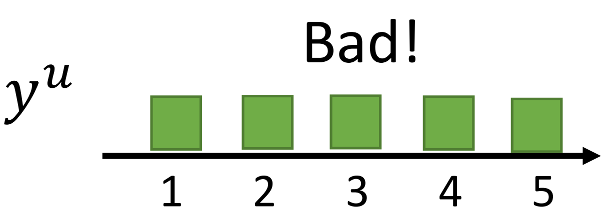

Distribution




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




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




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

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

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

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

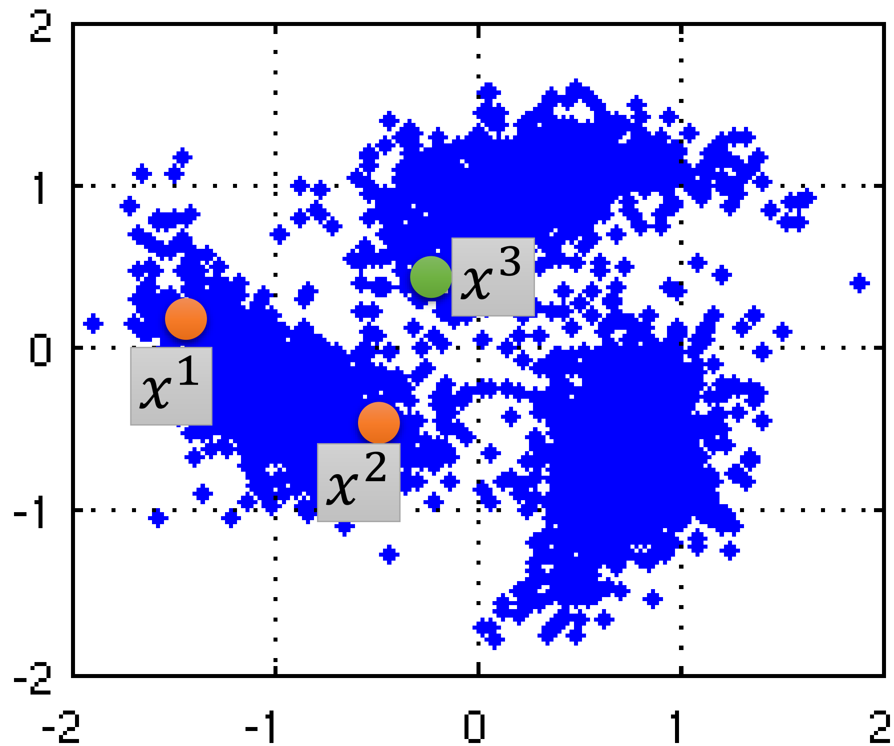
# Smoothness Assumption

近朱者赤，近墨者黑

*"You are known by the company you keep"*

- 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

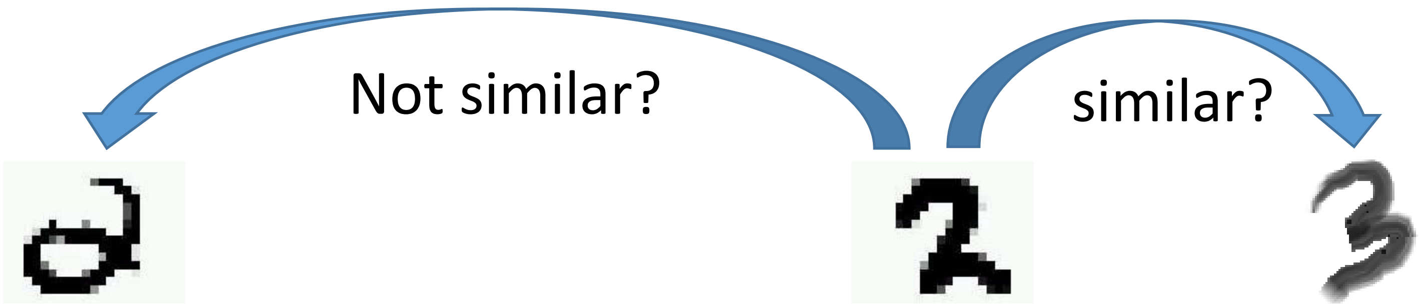



$x^1$ and $x^2$ have the same label

$x^2$ 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/

# Graph-based Approach

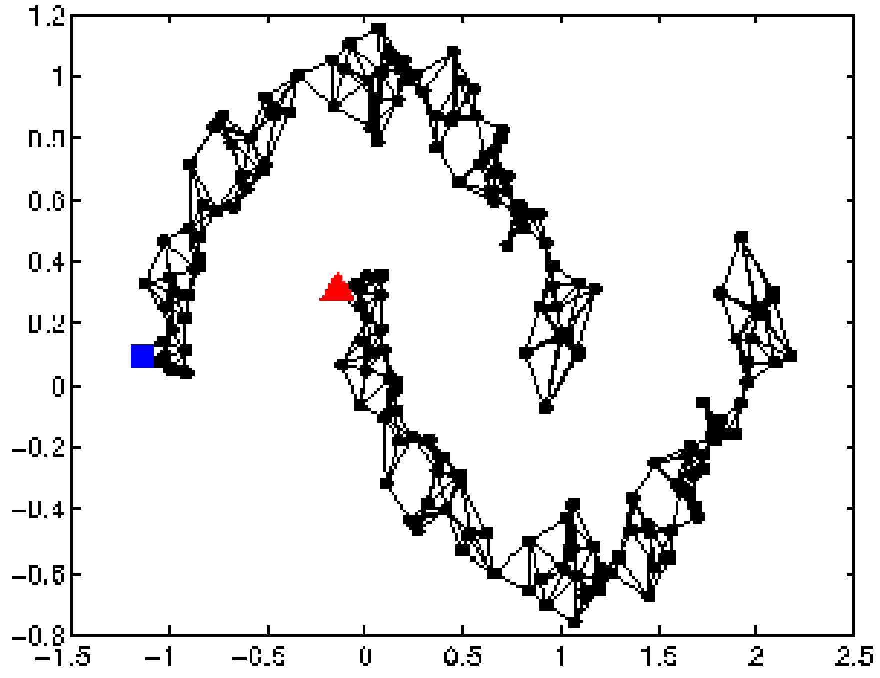
- How to know $x^1$ and $x^2$ are connected by a high density path

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

Graph representation is nature sometimes.

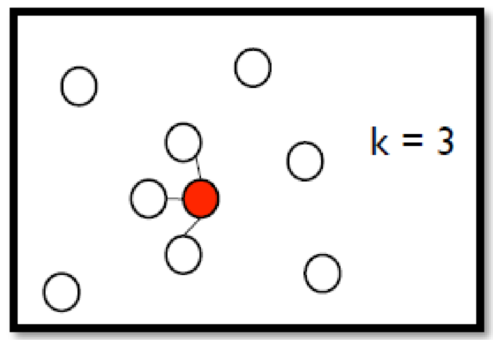
E.g. Hyperlink of webpages, citation of papers

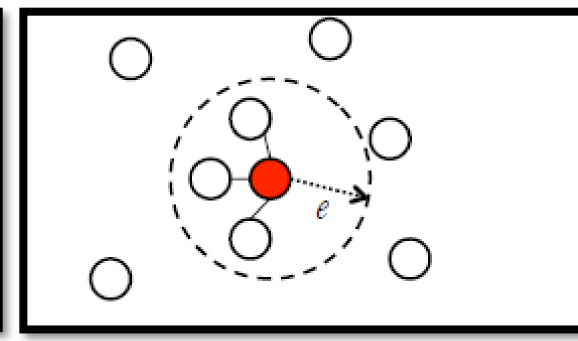
Sometimes you have to construct the graph yourself.



# Graph-based Approach
# - Graph Construction

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

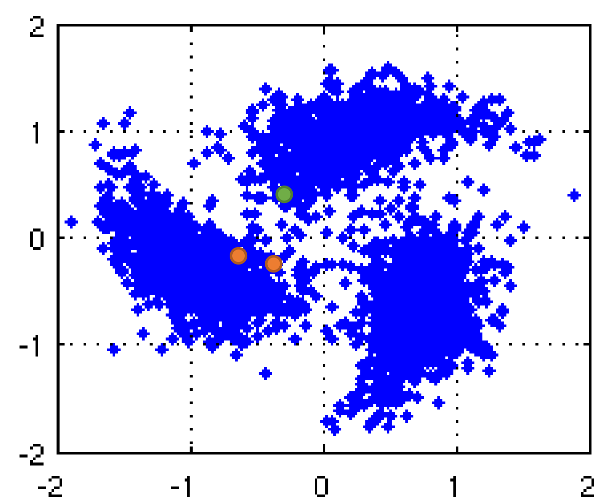


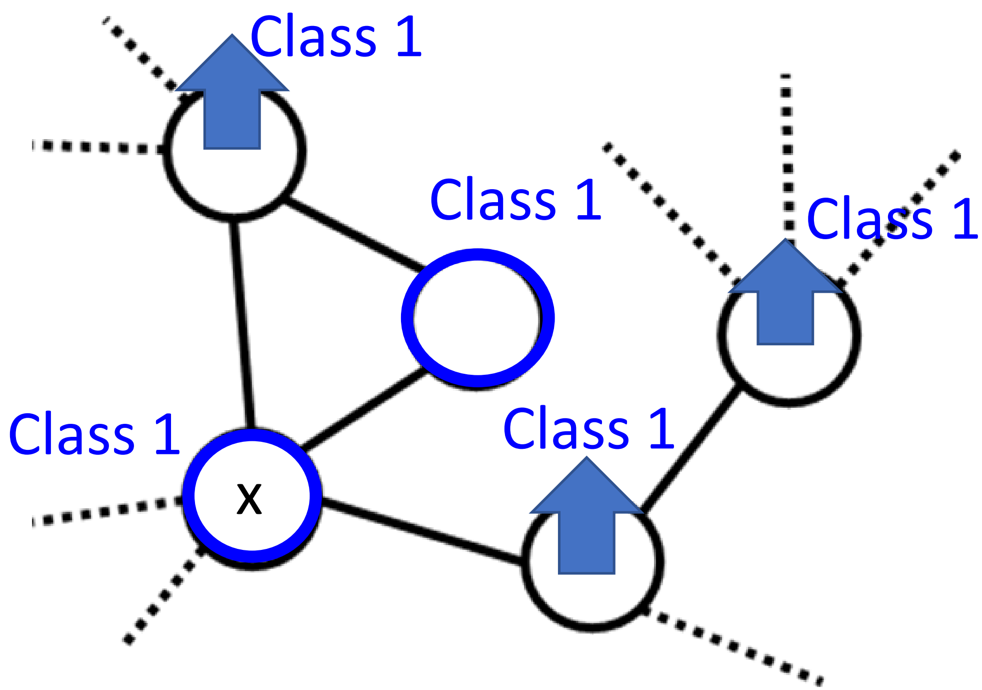

- Edge weight is proportional to s$\left(x^i, x^j\right)$

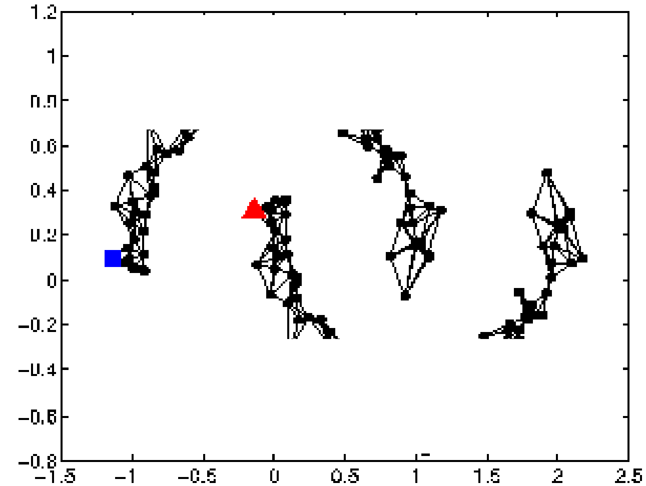
Gaussian Radial Basis Function:

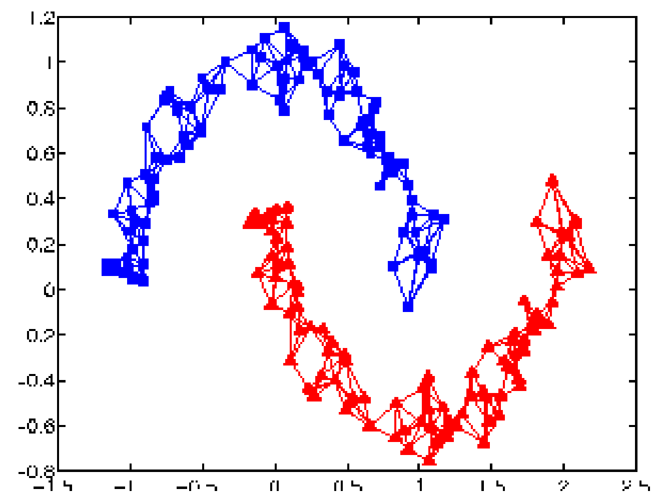
$$s\left(x^i, x^j\right) = exp\left(-\gamma \left\| x^i - x^j \right\|^2\right)$$



# Graph-based Approach




The labelled data influence their neighbors.

Propagate through the graph

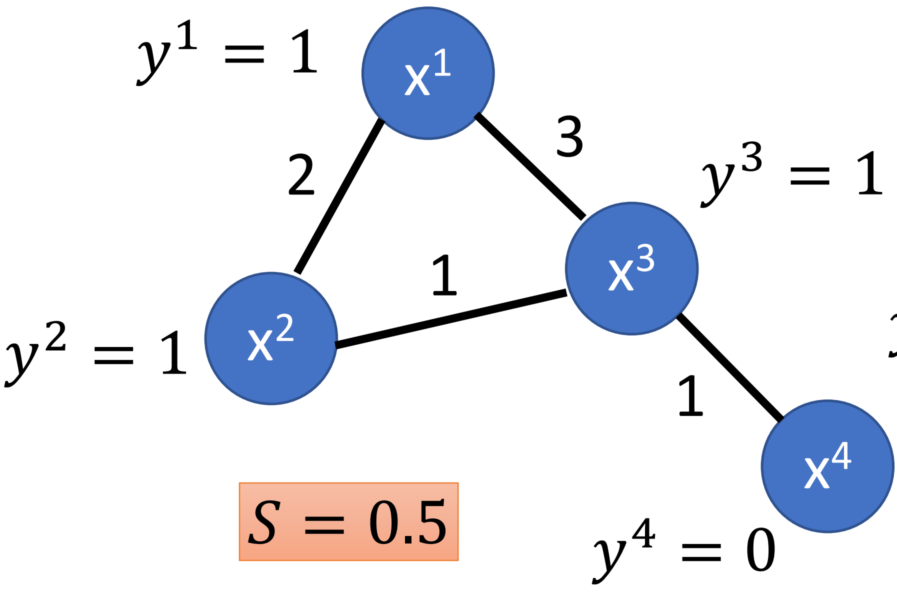
# Graph-based Approach

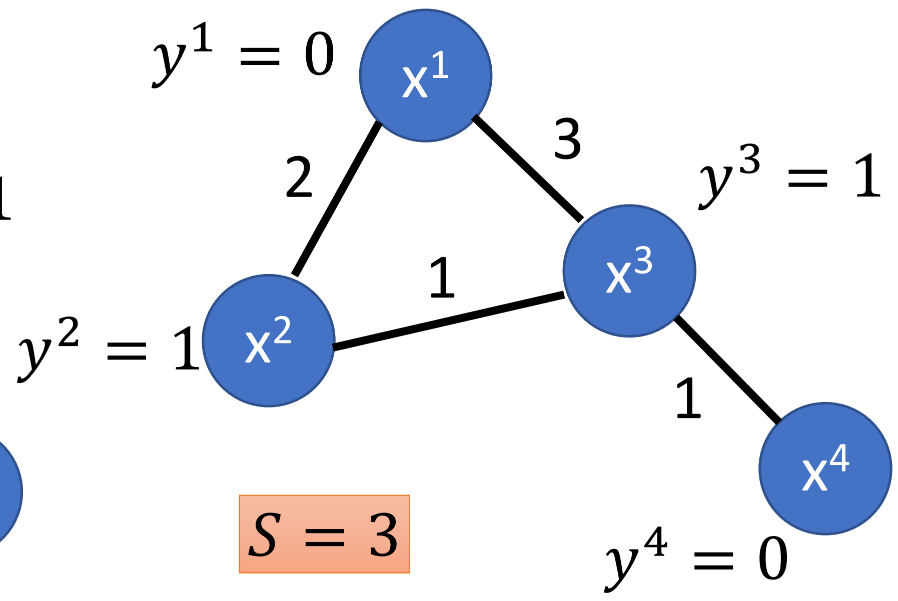
- Define the smoothness of the labels on the graph

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

Smaller means smoother

For all data (no matter labelled or not)

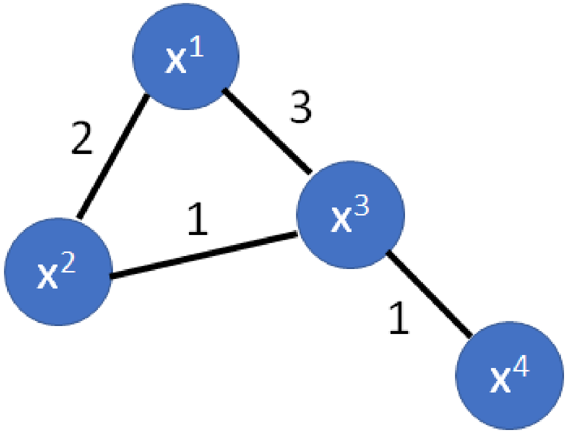


# Graph-based Approach

- 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

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

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

# 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 model parameters

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

As a regularization term

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