2019
怪兽
学堂

Semi-supervised Learning

虾米　　2019-4

~~Semi-~~Supervised Learning

Supervised Learning = learning from labeled data. Dominant paradigm in Machine Learning.

- E.g, say you want to train an email classifier to distinguish spam from important messages

~~Semi-~~Supervised Learning

Supervised Learning = learning from labeled data. Dominant paradigm in Machine Learning.

- E.g, say you want to train an email classifier to distinguish spam from important messages
- Take sample S of data, labeled according to whether they were/weren't spam.

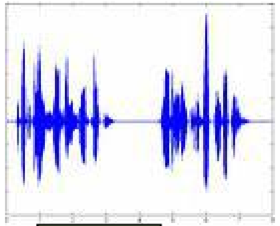
~~Semi-~~Supervised Learning

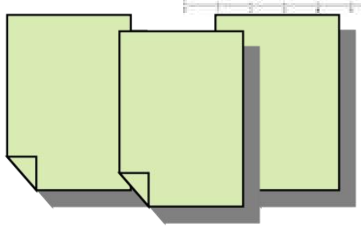
Supervised Learning = learning from labeled data. Dominant paradigm in Machine Learning.

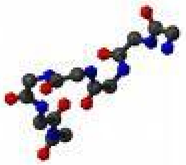
- E.g, say you want to train an email classifier to distinguish spam from important messages
- Take sample S of data, labeled according to whether they were/weren't spam.
- Train a classifier (like SVM, decision tree, etc) on S. Make sure it's not overfitting.
- Use to classify new emails.

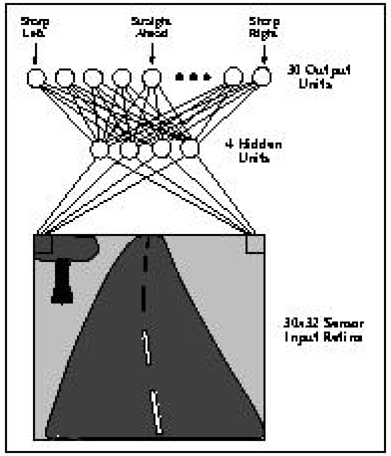
Basic paradigm has many successes

- recognize speech,
- steer a car,
- classify documents
- classify proteins
- recognizing faces, objects in images
- …









However, for many problems, labeled data can be rare or expensive.

Need to pay someone to do it, requires special testing,…

Unlabeled data is much cheaper.

However, for many problems, labeled data can be rare or expensive.

Need to pay someone to do it, requires special testing,…

Unlabeled data is much cheaper.

Speech	Customer modeling
Images	Protein sequences
Medical outcomes	Web pages

However, for many problems, labeled data can be rare or expensive.

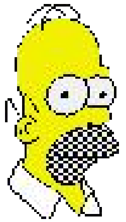
Need to pay someone to do it, requires special testing,…

Unlabeled data is much cheaper.

Can we make use of cheap unlabeled data?

Semi-Supervised Learning

Can we use unlabeled data to augment a small labeled sample to improve learning?



But unlabeled data is missing the most important info!!

But maybe still has useful regularities that we can use.

Bu Bu But…

Semi-Supervised Learning

Substantial recent work in ML. A number of interesting methods have been developed.

This talk:

- Discuss several diverse methods for taking advantage of unlabeled data.
- General framework to understand when unlabeled data can help, and make sense of what's going on.

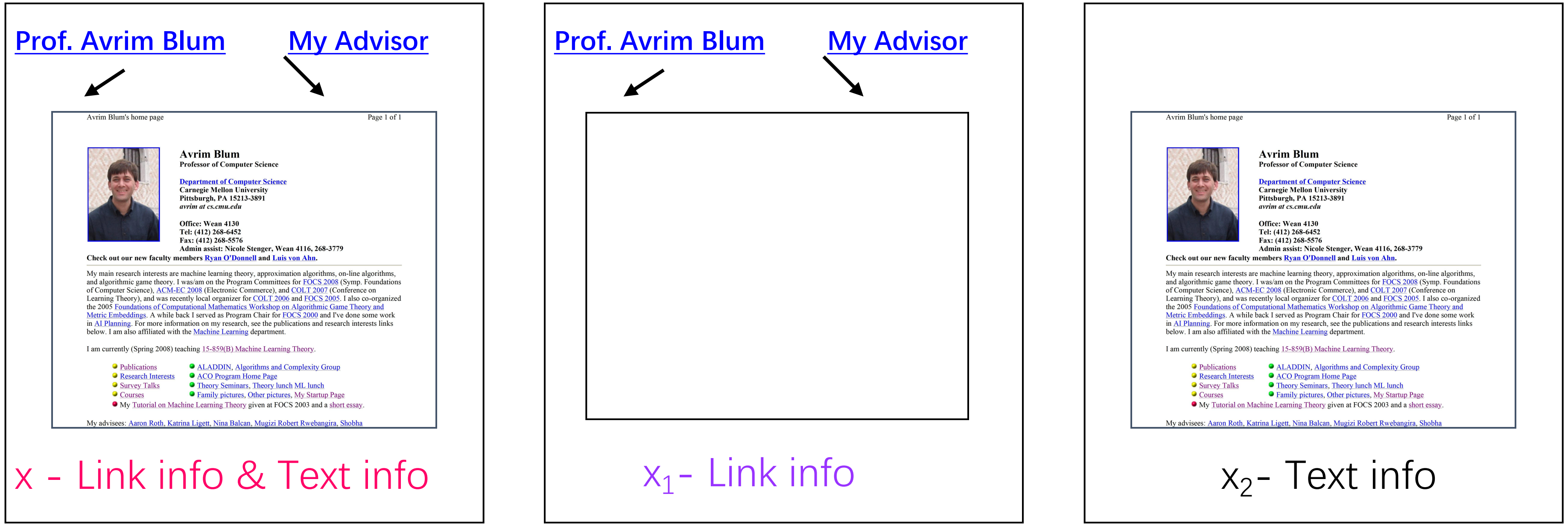
Method 1:

Co-Training

Co-training

Many problems have two different sources of info you can use to determine label.

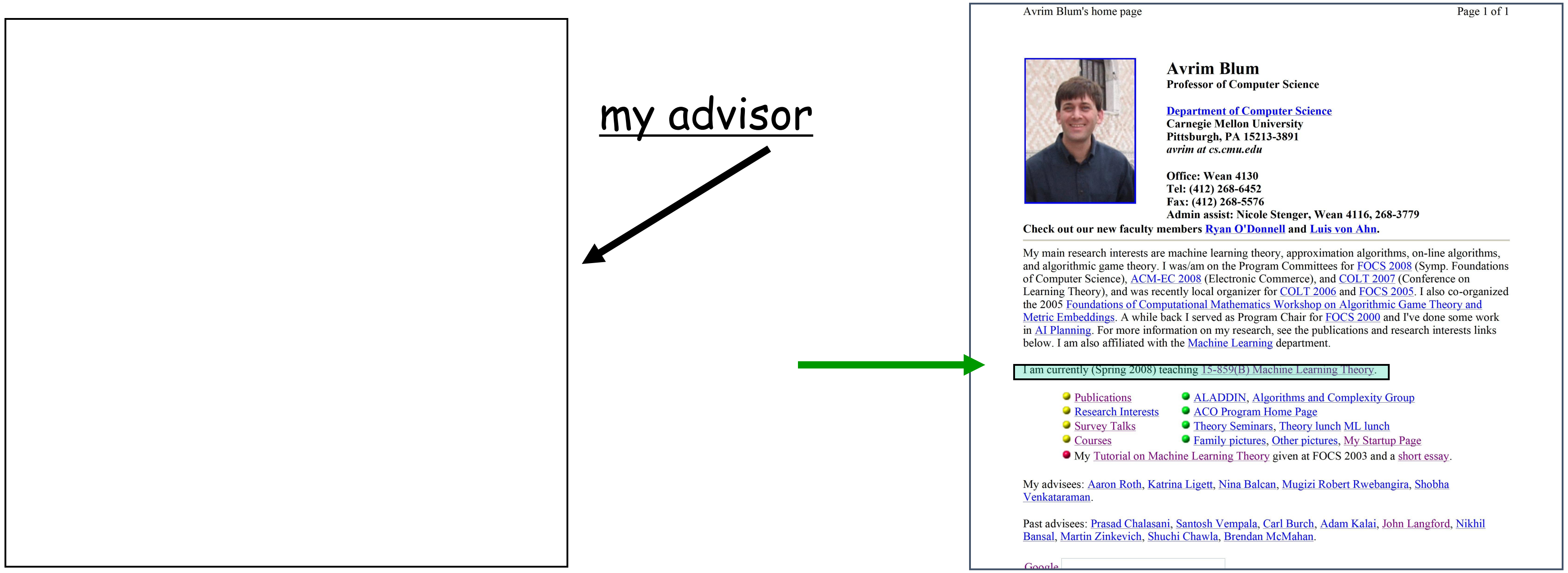
E.g., classifying webpages: can use words on page or words on links pointing to the page.



Co-training

Idea: Use small labeled sample to learn initial rules.

- E.g., “my advisor” pointing to a page is a good indicator it is a faculty home page.
- E.g., “I am teaching” on a page is a good indicator it is a faculty home page.

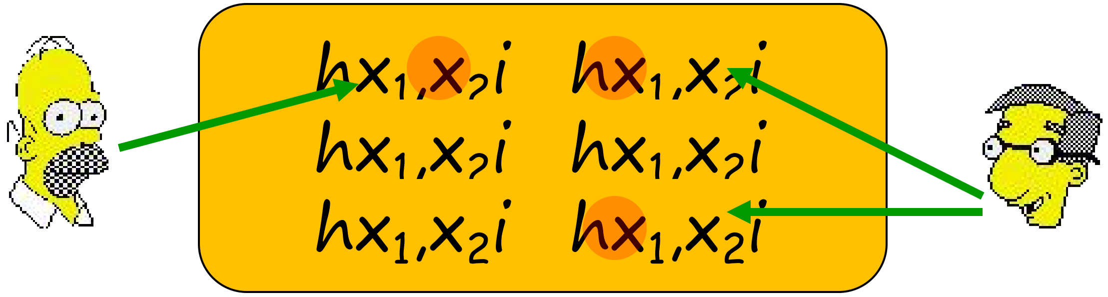


Co-training

Idea: Use small labeled sample to learn initial rules.

- E.g., "my advisor" pointing to a page is a good indicator it is a faculty home page.
- E.g., "I am teaching" on a page is a good indicator it is a faculty home page.

Then look for unlabeled examples where one rule is confident and the other is not. Have it label the example for the other.

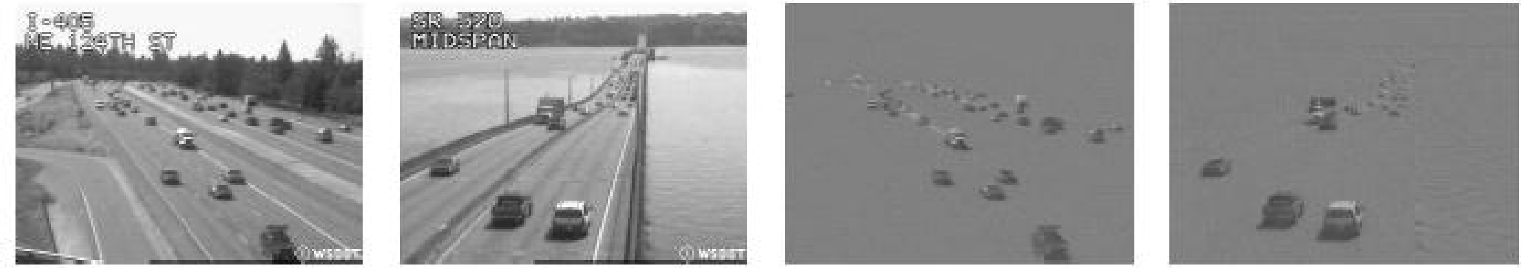


Training 2 classifiers, one on each type of info. Using each to help train the other.

Co-training

Turns out a number of problems can be set up this way.

E.g., [Levin-Viola-Freund03] identifying objects in images. Two different kinds of preprocessing.



E.g., [Collins&Singer99] named-entity extraction.

- "I arrived in London yesterday"

…

Co-training

- Setting is each example x = $h x_1, x_2 i$, where x_1, x_2 are two “views” of the data.
- Have separate algorithms running on each view. Use each to help train the other.
- Basic hope is that two views are consistent. Using agreement as proxy for labeled data.

Method 2:

semi-supervised learning for GAN

Supervised Generative Model

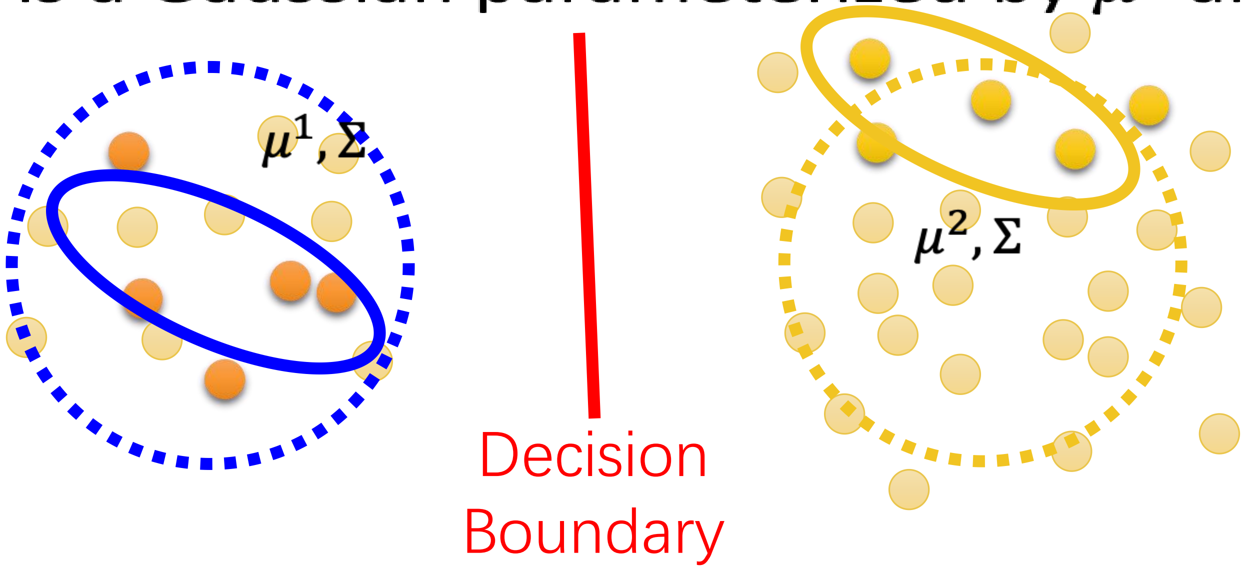
- Given labelled training examples $x^r \in C_1, C_2$
  - looking for most likely prior probability $P(C_i)$ and class-dependent probability $P(x|C_i)$
  - $P(x|C_i)$ is a Gaussian parameterized by μ^i and Σ

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

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

……

Why?

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

- Maximum likelihood with labelled data

Closed-form solution

$$\log L(\theta) = \sum_{x^r} \log P_\theta(x^r, \hat{y}^r)$$

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

- Maximum likelihood with labelled + unlabeled data

$$\log L(\theta) = \sum_{x^r} \log P_\theta(x^r, \hat{y}^r) + \sum_{x^u} \log P_\theta(x^u)$$

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)

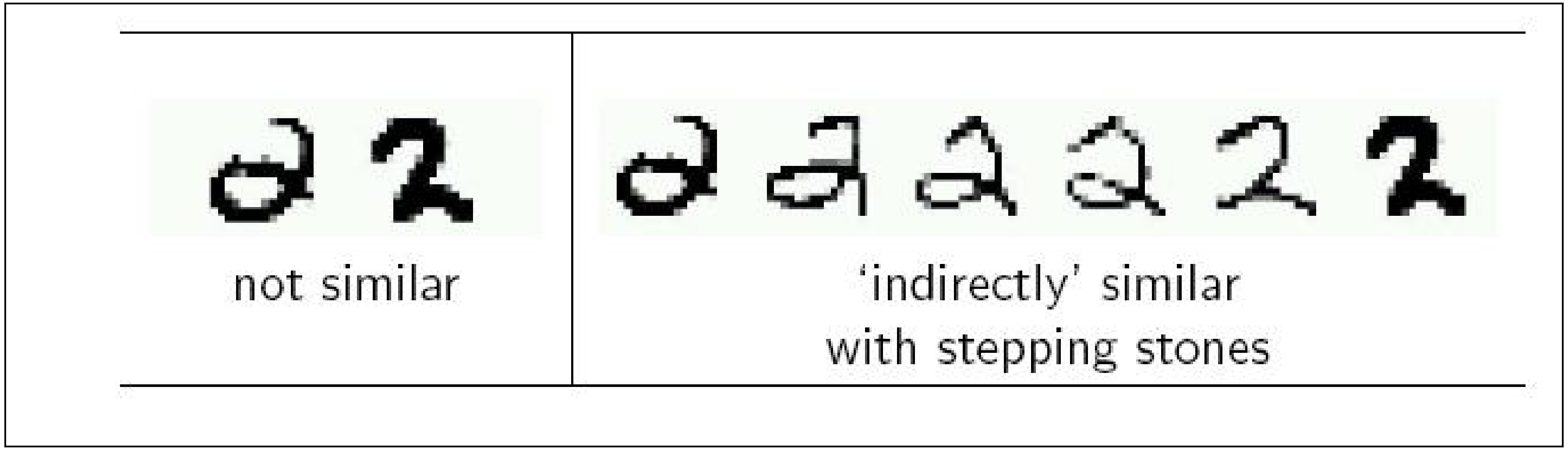
Method 3:

Graph-based methods

Graph-based methods

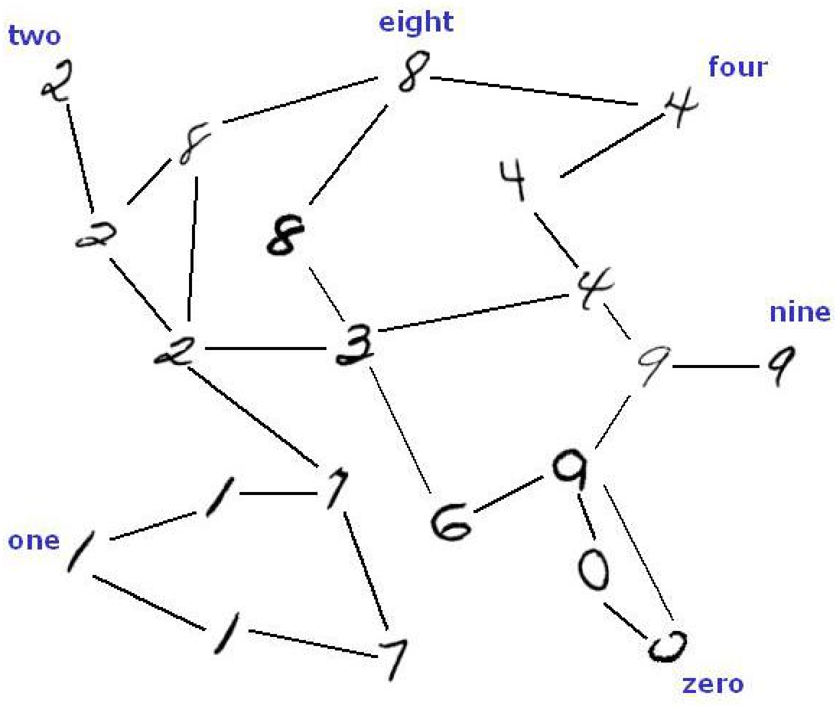
- Suppose we believe that very similar examples probably have the same label.
- If you have a lot of labeled data, this suggests a Nearest-Neighbor type of alg.
- If you have a lot of unlabeled data, perhaps can use them as "stepping stones"

E.g., handwritten digits [Zhu07]:



Graph-based methods

- Idea: construct a graph with edges between very similar examples.
- Unlabeled data can help “glue” the objects of the same class together.



Graph-based methods

- Idea: construct a graph with edges between very similar examples.
- Unlabeled data can help “glue” the objects of the same class together.

image 4005

neighbor 1: time edge

neighbor 2: color edge

neighbor 3: color edge

neighbor 4: color edge

neighbor 5: face edge

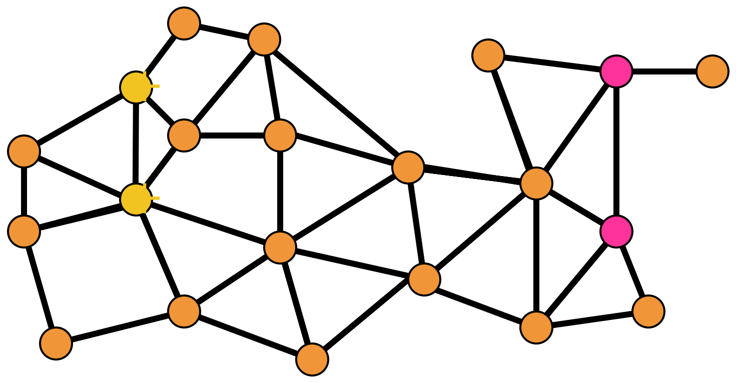
Graph-based methods

- Idea: construct a graph with edges between very similar examples.
- Unlabeled data can help "glue" the objects of the same class together.

Solve for:

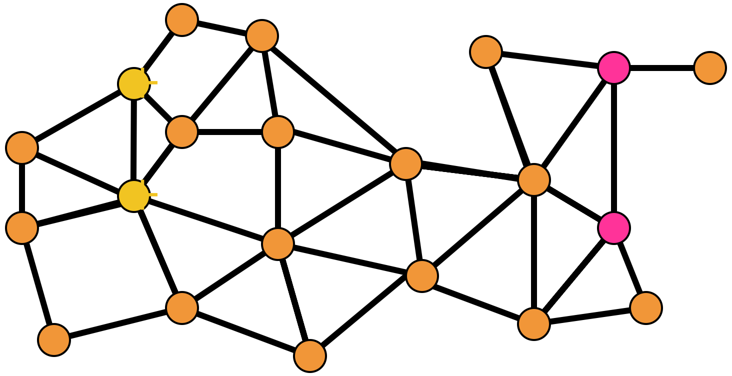
- Minimum cut [BC,BLRR]
- Minimum "soft-cut" [ZGL]

  $\sum_{e=(u,v)}(f(u)-f(v))^2$
- Spectral partitioning [J]
- …



Graph-based methods

- Suppose just two labels: 0 & 1.
- Solve for labels 0 · f(x) · 1 for unlabeled examples x to minimize:
  - $\sum_{e=(u,v)}|f(u)-f(v)|$ [soln = minimum cut]
  - $\sum_{e=(u,v)} (f(u)-f(v))^2$ [soln = electric potentials]
  - …



Method 4:

self-training

Self–training algorithm

Assumption

One's own high confidence predictions are correct.

Self-training algorithm:

1. Train f from (X_l, Y_l)
2. Predict on $x \in X_u$
3. Add $(x, f(x))$ to labeled data
4. Repeat

Advantages of self-training

- The simplest semi-supervised learning method.
- A wrapper method, applies to existing (complex) classifiers. Often used in real tasks like natural language processing.

Disadvantages of self-training

- Early mistakes could reinforce themselves.
  - Heuristic solutions, e.g. "un-label" an instance if its confidence falls below a threshold.
- Cannot say too much in terms of convergence.
  - But there are special cases when self-training is equivalent to the Expectation-Maximization (EM) algorithm.
  - There are also special cases (e.g., linear functions) when the closed-form solution is known.

2019
怪兽
学堂

THANKS

汇报人XXX