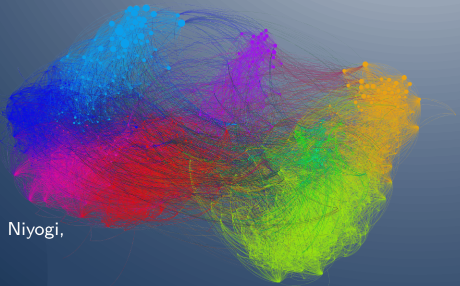
Graphs in Machine Learning

Semi-Supervised Learning Introduction

Why and When SSL Helps

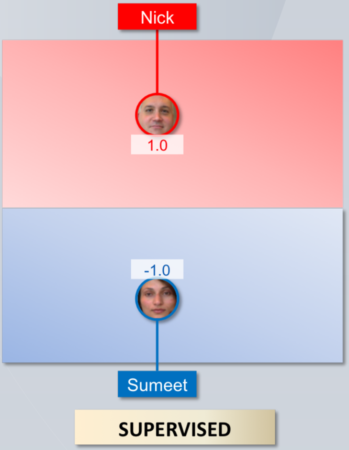
Michal Valko

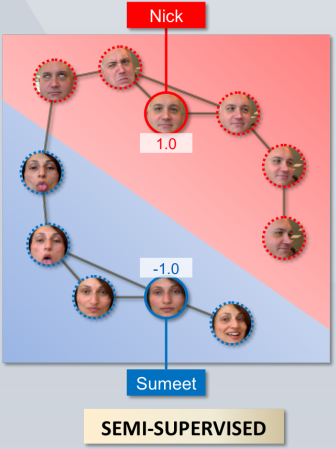
Inria & ENS Paris-Saclay, MVA



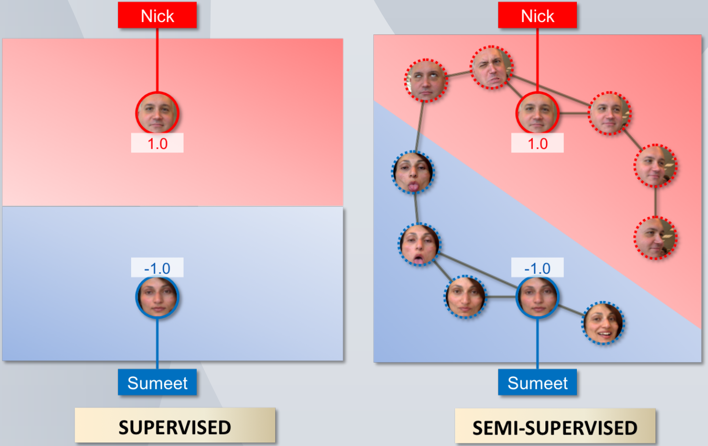
Partially based on material by: Mikhail Belkin, Partha Niyogi, Olivier Chapelle, Bernhard Schölkopf

Semi-supervised learning: How is it possible?





Semi-supervised learning: How is it possible?



This is how children learn! hypothesis

Semi-supervised learning (SSL)

SSL problem: definition

Semi-supervised learning (SSL)

SSL problem: definition

Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{N}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Semi-supervised learning (SSL)

SSL problem: definition

Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{N}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Some facts about **SSL**

- assumes that the unlabeled data is useful

Semi-supervised learning (SSL)

SSL problem: definition

Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{N}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Some facts about SSL

- assumes that the unlabeled data is useful
- works with data geometry assumptions
  - cluster assumption — low-density separation
  - smoothness assumptions, generative models, ...
  - manifold assumption

Semi-supervised learning (SSL)

SSL problem: definition

Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{N}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Some facts about SSL

- assumes that the unlabeled data is useful
- works with data geometry assumptions
  - cluster assumption — low-density separation
  - smoothness assumptions, generative models, ...
  - manifold assumption
- now it helps now, now it does not (sic)
  - provable cases when it helps

Semi-supervised learning (SSL)

SSL problem: definition

Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{N}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Some facts about **SSL**

- assumes that the unlabeled data is useful
- works with data geometry assumptions
  - cluster assumption — low-density separation
  - smoothness assumptions, generative models, ...
  - manifold assumption
- now it helps now, now it does not (sic)
  - provable cases when it helps
- inductive or transductive/out-of-sample extension

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

SSL: Overview: Self-Training

SSL: Self-Training

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$

Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$
Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$

Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method
- heavily depends on the the internal classifier

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$
Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method
- heavily depends on the the internal classifier
- some theory exist for specific classifiers

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$
Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method
- heavily depends on the the internal classifier
- some theory exist for specific classifiers
- nobody uses it anymore ($\neq$ self supervised)

SSL: Overview: Self-Training

SSL: Self-Training

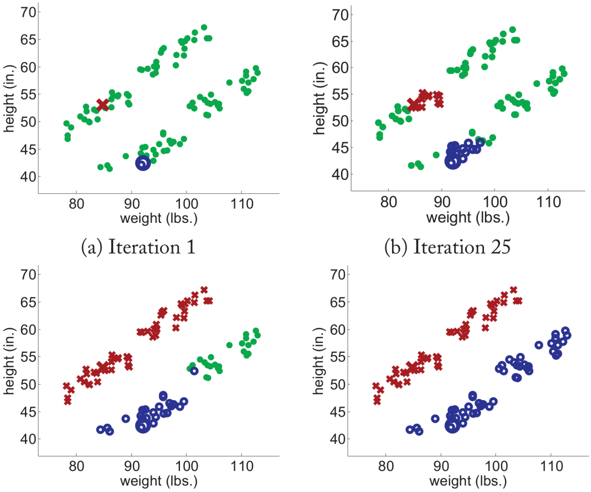
Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$
Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method
- heavily depends on the the internal classifier
- some theory exist for specific classifiers
- nobody uses it anymore ($\neq$ self supervised)
- errors propagate (unless the clusters are well separated)

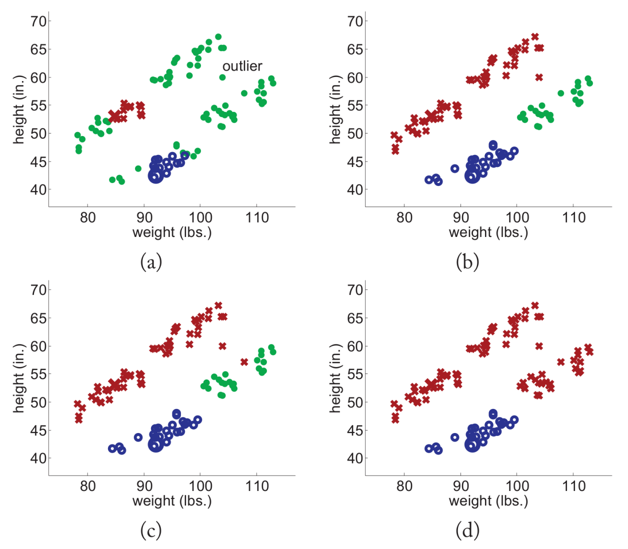
SSL: Self-Training (Good Case)



(a) Iteration 1 (b) Iteration 25

(c) Iteration 74 (d) Final labeling of all instances

SSL: Self-Training (Bad Case)



SSL$(\mathcal{G})$

semi-supervised learning with graphs and harmonic functions

...our running example for learning with graphs

SSL with Graphs: Prehistory

Blum/Chawla: Learning from Labeled and Unlabeled Data using Graph Mincuts
http://www.aladdin.cs.cmu.edu/papers/pdfs/y2001/mincut.pdf

SSL with Graphs: Prehistory

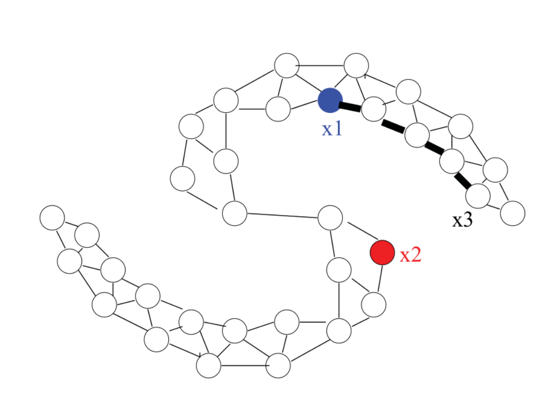
Blum/Chawla: Learning from Labeled and Unlabeled Data using Graph Mincuts
`http://www.aladdin.cs.cmu.edu/papers/pdfs/y2001/mincut.pdf`

*following some insights from vision research in 1980s

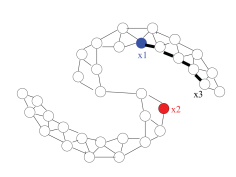
SSL with Graphs: Prehistory

Blum/Chawla: Learning from Labeled and Unlabeled Data using Graph Mincuts
`http://www.aladdin.cs.cmu.edu/papers/pdfs/y2001/mincut.pdf`

*following some insights from vision research in 1980s

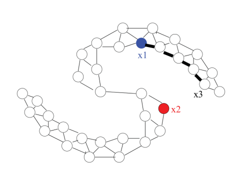


SSL with Graphs: MinCut



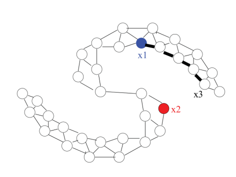
MinCut SSL: an idea similar to MinCut clustering

SSL with Graphs: MinCut



MinCut SSL: an idea similar to MinCut clustering
Where is the link?

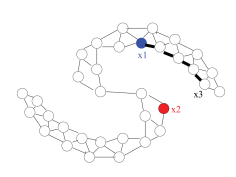
SSL with Graphs: MinCut



MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

SSL with Graphs: MinCut

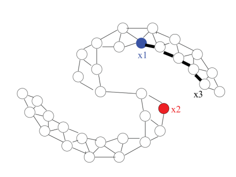


MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

What is the formal statement?

SSL with Graphs: MinCut



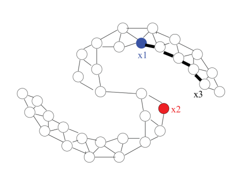
MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

What is the formal statement? We look for $f(\mathbf{x}) \in \{\pm 1\}$

$$\mathrm{cut} =$$

SSL with Graphs: MinCut



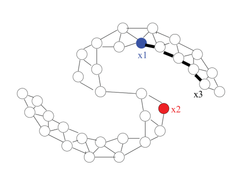
MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

What is the formal statement? We look for $f(\mathbf{x}) \in \{\pm 1\}$

$$\mathrm{cut} = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

SSL with Graphs: MinCut



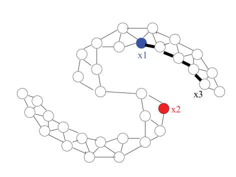
MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

What is the formal statement? We look for $f(\mathbf{x}) \in \{\pm 1\}$

$$\mathrm{cut} = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j) \right)^2 = \Omega(f)$$

SSL with Graphs: MinCut



MinCut SSL: an idea similar to MinCut clustering

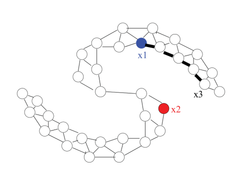
Where is the link? connected classes, not necessarily compact

What is the formal statement? We look for $f(\mathbf{x}) \in \{\pm 1\}$

$$\mathrm{cut} = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2 = \Omega(f)$$

Why $\left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$ and not $|f(\mathbf{x}_i) - f(\mathbf{x}_j)|$?

SSL with Graphs: MinCut



MinCut SSL: an idea similar to MinCut clustering

Where is the link? connected classes, not necessarily compact

What is the formal statement? We look for $f(\mathbf{x}) \in \{\pm 1\}$

$$\mathrm{cut} = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2 = \Omega(f)$$

Why $\left(f\left(\mathbf{x}_i\right) - f\left(\mathbf{x}_j\right)\right)^2$ and not $|f(\mathbf{x}_i) - f(\mathbf{x}_j)|$? It does not matter.

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j) \right)^2$$

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

Clustering was unsupervised, here we have supervised data.

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

Clustering was unsupervised, here we have supervised data.

Recall the general objective-function framework:

$$\min_{\mathbf{w},b} \sum_{i}^{n_l} V(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \lambda \Omega(\mathbf{f})$$

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

Clustering was unsupervised, here we have supervised data.

Recall the general objective-function framework:

$$\min_{\mathbf{w},b} \sum_{i}^{n_l} V(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \lambda \Omega(\mathbf{f})$$

It would be nice if we match the prediction on labeled data:

$$V(\mathbf{x}, y, f(\mathbf{x}))$$

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

Clustering was unsupervised, here we have supervised data.

Recall the general objective-function framework:

$$\min_{\mathbf{w},b} \sum_{i}^{n_l} V(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \lambda \Omega(\mathbf{f})$$

It would be nice if we match the prediction on labeled data:

$$V(\mathbf{x}, y, f(\mathbf{x})) = \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2$$

SSL with Graphs: MinCut

We look for $f(\mathbf{x}) \in \{\pm 1\}$ to minimize the cut $\Omega(\mathbf{f})$

$$\Omega(\mathbf{f}) = \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

Clustering was unsupervised, here we have supervised data.

Recall the general objective-function framework:

$$\min_{\mathbf{w},b} \sum_{i}^{n_l} V(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \lambda\Omega(\mathbf{f})$$

It would be nice if we match the prediction on labeled data:

$$V(\mathbf{x}, y, f(\mathbf{x})) = \infty \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2$$

SSL with Graphs: MinCut

SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f} \in \{\pm 1\}^{n_l + n_u}} \infty \sum_{i=1}^{n_l} (f(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2$$

SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f}\in\{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} (f(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2$$

This is an integer program :(

SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f} \in \{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} (f(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2$$

This is an integer program :(

Can we solve it?

SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f}\in\{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} (f(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2$$

This is an integer program :(

Can we solve it? It still just MinCut.

SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f}\in\{\pm1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

This is an integer program :(

Can we solve it? It still just MinCut. Are we happy?

SSL with Graphs: MinCut

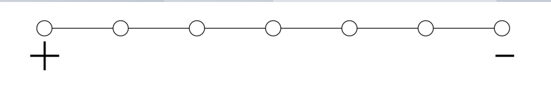
Final objective function:

$$\min_{\mathbf{f}\in\{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij}\left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

This is an integer program :(

Can we solve it? It still just MinCut.

Are we happy?



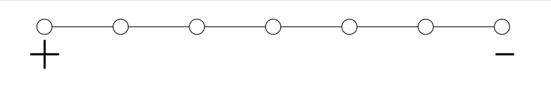
SSL with Graphs: MinCut

Final objective function:

$$\min_{\mathbf{f}\in\{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij}\left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

This is an integer program :(

Can we solve it? It still just MinCut. Are we happy?



There are six solutions.

SSL with Graphs: MinCut

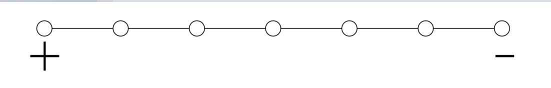
Final objective function:

$$\min_{\mathbf{f}\in\{\pm 1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} (f(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

This is an integer program :(

Can we solve it? It still just MinCut.

Are we happy?



There are six solutions. All equivalent.

SSL with Graphs: MinCut

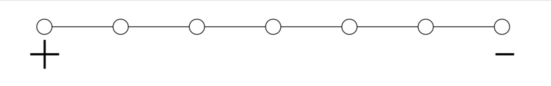
Final objective function:

$$\min_{\mathbf{f}\in\{\pm1\}^{n_l+n_u}} \infty \sum_{i=1}^{n_l} \left(f(\mathbf{x}_i) - y_i\right)^2 + \lambda \sum_{i,j=1}^{n_l+n_u} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2$$

This is an integer program :(

Can we solve it? It still just MinCut.

Are we happy?



There are six solutions. All equivalent.

We need a better way to reflect the confidence.

Michal Valko

michal.valko@inria.fr

Inria & ENS Paris-Saclay, MVA

https://misovalko.github.io/mva-ml-graphs.html