



# Graphs in Machine Learning

Michal Valko

*Inria Lille - Nord Europe, France*

TA: Pierre Perrault

Partially based on material by: Mikhail Belkin, Branislav Kveton



## Previous Lecture

- ▶ Manifold learning with Laplacian eigenmaps
- ▶ Semi-Supervised Learning
  - ▶ Why and when it helps?
  - ▶ Self-training
  - ▶ Semi-supervised SVMs
- ▶ Graph-based semi-supervised learning
- ▶ SSL with MinCuts
- ▶ Gaussian random fields and harmonic solution
- ▶ Regularization of harmonic solution
- ▶ Soft-harmonic solution
- ▶ Inductive and transductive semi-supervised learning
- ▶ Manifold regularization

# This Lecture

- ▶ Max-Margin Graph Cuts
- ▶ Theory of Laplacian-based manifold methods
- ▶ Transductive learning stability based bounds
- ▶ Online Semi-Supervised Learning
- ▶ Online incremental  $k$ -centers

# Previous Lab Session

- ▶ 23. 10. 2017 by Pierre Perrault
- ▶ Content
  - ▶ Graph Construction
  - ▶ Test sensitivity to parameters:  $\sigma$ ,  $k$ ,  $\varepsilon$
  - ▶ Spectral Clustering
  - ▶ Spectral Clustering vs.  $k$ -means
  - ▶ Image Segmentation
- ▶ Short written report
- ▶ Questions to piazza (without giving away solutions)
- ▶ *Deadline:* 6. 11. 2017 **Today!**

# Next Lab Session

- ▶ 13. 11. 2017 by Pierre Perrault
- ▶ Content
  - ▶ Semi-supervised learning
  - ▶ Graph quantization
  - ▶ Online face recognizer
- ▶ AR: **record a video with faces**
- ▶ Install VM (in case you have not done it yet for TD1)
- ▶ Short written report
- ▶ Questions to piazza
- ▶ **Deadline: 27. 11. 2017**

# Final Class projects

- ▶ detailed description on the class website
- ▶ preferred option: you come up with the topic
- ▶ theory/implementation/review or a combination
- ▶ one or two people per project (exceptionally three)
- ▶ grade 60%: report + short presentation of the **team**
- ▶ deadlines
  - ▶ 20. 11. 2017 - strongly recommended DL for taking projects
  - ▶ 27. 11. 2017 - hard DL for taking projects
  - ▶ 08. 01. 2018 - submission of the project report
  - ▶ 09. 01. 2018 or later - project presentation
- ▶ list of suggested topics on piazza

# Checkpoint 1

Semi-supervised learning with graphs:

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

Regularized harmonic Solution:

$$\mathbf{f}_u = (\mathbf{L}_{uu} + \gamma \mathbf{I})^{-1} (\mathbf{W}_{ul} \mathbf{f}_l)$$

## Checkpoint 2

Unconstrained regularization in general:

$$\mathbf{f}^* = \min_{\mathbf{f} \in \mathbb{R}^N} (\mathbf{f} - \mathbf{y})^T \mathbf{C} (\mathbf{f} - \mathbf{y}) + \mathbf{f}^T \mathbf{Q} \mathbf{f}$$

Out of sample extension: Laplacian SVMs

$$f^* = \arg \min_{f \in \mathcal{H}_{\mathcal{K}}} \sum_i^{n_i} \max(0, 1 - y f(\mathbf{x})) + \lambda_1 \|f\|_{\mathcal{K}}^2 + \lambda_2 \mathbf{f}^T \mathbf{L} \mathbf{f}$$



# SSL with Graphs: Laplacian SVMs

$$f^* = \arg \min_{f \in \mathcal{H}_{\mathcal{K}}} \sum_i^{n_i} \max(0, 1 - yf(\mathbf{x})) + \lambda_1 \|f\|_{\mathcal{K}}^2 + \lambda_2 \mathbf{f}^T \mathbf{L} \mathbf{f}$$

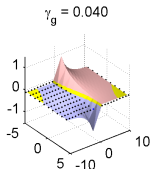
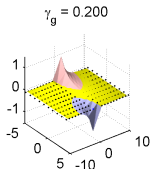
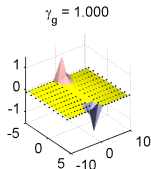
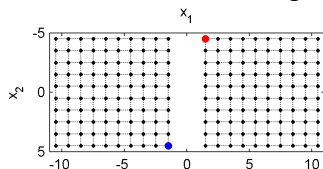
$\mathcal{H}_{\mathcal{K}}$  is nice and expressive.

Can there be a problem with certain  $\mathcal{H}_{\mathcal{K}}$ ?

We look for  $f$  only in  $\mathcal{H}_{\mathcal{K}}$ .

If it is simple (e.g., **linear**) minimization of  $\mathbf{f}^T \mathbf{L} \mathbf{f}$  can perform badly.

Consider again this 2D data and linear  $\mathcal{K}$ .



# SSL with Graphs: Laplacian SVMs

Linear  $\mathcal{K} \equiv$  functions with slope  $\alpha_1$  and intercept  $\alpha_2$ .

$$\min_{\alpha_1, \alpha_2} \sum_i^{n_l} V(f, \mathbf{x}_i, y_i) + \lambda_1 [\alpha_1^2 + \alpha_2^2] + \lambda_2 \mathbf{f}^T \mathbf{L} \mathbf{f}$$

For this simple case we can write down  $\mathbf{f}^T \mathbf{L} \mathbf{f}$  explicitly.

$$\begin{aligned} \mathbf{f}^T \mathbf{L} \mathbf{f} &= \frac{1}{2} \sum_{i,j} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \\ &= \frac{1}{2} \sum_{i,j} w_{ij} (\alpha_1(\mathbf{x}_{i1} - \mathbf{x}_{j1}) + \alpha_2(\mathbf{x}_{i2} - \mathbf{x}_{j2}))^2 \\ &= \frac{\alpha_1^2}{2} \underbrace{\sum_{i,j} w_{ij} (\mathbf{x}_{i1} - \mathbf{x}_{j1})^2}_{\Delta=218.351} + \frac{\alpha_2^2}{2} \underbrace{\sum_{i,j} w_{ij} (\mathbf{x}_{i2} - \mathbf{x}_{j2})^2}_{\Delta=218.351} \end{aligned}$$

# SSL with Graphs: Laplacian SVMs

2D data and linear  $\mathcal{K}$  objective

$$\min_{\alpha_1, \alpha_2} \sum_i^{n_l} V(f, \mathbf{x}_i, y_i) + \left( \lambda_1 + \frac{\lambda_2 \Delta}{2} \right) [\alpha_1^2 + \alpha_2^2]$$

Setting  $\lambda^* = \left( \lambda_1 + \frac{\lambda_2 \Delta}{2} \right)$ :

$$\min_{\alpha_1, \alpha_2} \sum_i^{n_l} V(f, \mathbf{x}_i, y_i) + \lambda^* [\alpha_1^2 + \alpha_2^2]$$

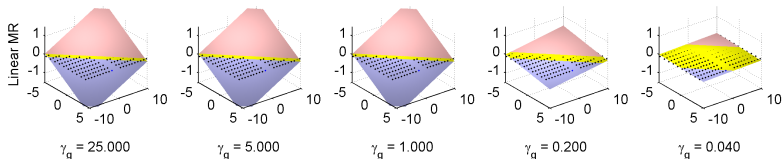
What does this objective function correspond to?

The only influence of unlabeled data is through  $\lambda^*$ .

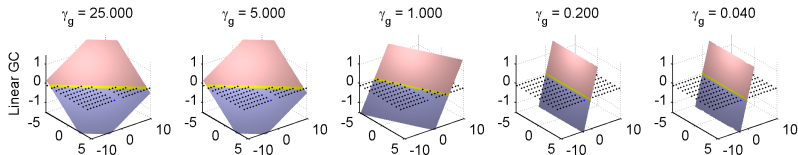
The same value of the objective as for supervised learning for some  $\lambda$  **without the unlabeled data!** This is not good.

# SSL with Graphs: Laplacian SVMs

MR for 2D data and linear  $\mathcal{K}$  only changes the slope



What would we like to see?



One solution: We use the unlabeled data **before** optimizing over  $\mathcal{H}_{\mathcal{K}}$ !

# SSL with Graphs: Max-Margin Graph Cuts

Let's take the confident data and use them as true!

$$\begin{aligned} f^* &= \min_{f \in \mathcal{H}_{\mathcal{X}}} \sum_{i: |\ell_i^*| \geq \varepsilon} V(f, \mathbf{x}_i, \text{sgn}(\ell_i^*)) + \gamma \|f\|_{\mathcal{K}}^2 \\ \text{s.t. } \ell^* &= \arg \min_{\ell \in \mathbb{R}^N} \ell^T (\mathbf{L} + \gamma_g \mathbf{I}) \ell \\ \text{s.t. } \ell_i &= y_i \text{ for all } i = 1, \dots, n_I \end{aligned}$$

Wait, but this is what we did not like in self-training!

Will we get into the same trouble?

Representer theorem is still cool:

$$f^*(\mathbf{x}) = \sum_{i: |f_i^*| \geq \varepsilon} \alpha_i^* \mathcal{K}(\mathbf{x}_i, \mathbf{x})$$

# SSL with Graphs: Generalization Bounds

Why is this not a witchcraft?

We take GC as an example. MR or HFS are similar.

What kind of guarantees we want?

We may want to bound the **risk**

$$R_P(f) = \mathbb{E}_{P(\mathbf{x})} [\mathcal{L}(f(\mathbf{x}), y(\mathbf{x}))]$$

for some **loss**, e.g., 0/1 loss

$$\mathcal{L}(y', y) = \mathbb{1}\{\text{sgn}(y') \neq y\}$$

What makes sense to bound  $R_P(f)$  with?

**empirical risk** + **error terms**

# SSL with Graphs: Generalization Bounds

True risk vs. empirical risk

$$R_P(f) = \frac{1}{N} \sum_i (f_i - y_i)^2$$

$$\hat{R}_P(f) = \frac{1}{n_I} \sum_{i \in I} (f_i - y_i)^2$$

We look for the bound in the form

$$R_P(f) \leq \hat{R}_P(f) + \text{errors}$$

$$\text{errors} = \text{transductive} + \text{inductive}$$

# SSL with Graphs: Generalization Bounds

Bounding **inductive** error (using classical SLT tools)

With probability  $1 - \eta$ , using Equations 3.15 and 3.24 [Vap95]

$$R_P(f) \leq \frac{1}{n} \sum_i \mathcal{L}(f(\mathbf{x}_i), y_i) + \Delta_I(h, n, \eta).$$

$n \equiv$  number of samples ,  $h \equiv$  VC dimension of the class

$$\Delta_I(h, n, \eta) = \sqrt{\frac{h(\ln(2n/h) + 1) - \ln(\eta/4)}{n}}$$

How to bound  $\mathcal{L}(f(\mathbf{x}_i), y_i)$ ? For any  $y_i \in \{-1, 1\}$  and  $\ell_i^*$

$$\mathcal{L}(f(\mathbf{x}_i), y_i) \leq \mathcal{L}(f(\mathbf{x}_i), \text{sgn}(\ell_i^*)) + (\ell_i^* - y_i)^2.$$



# SSL with Graphs: Generalization Bounds

Bounding **transductive error** (using stability analysis)

<http://www.cs.nyu.edu/~mohri/pub/str.pdf>

How to bound  $(\ell_i^* - y_i)^2$ ?

Bounding  $(\ell_i^* - y_i)^2$  for hard case is difficult  $\rightarrow$  we bound soft HFS:

$$\ell^* = \min_{\ell \in \mathbb{R}^N} (\ell - \mathbf{y})^T \mathbf{C} (\ell - \mathbf{y}) + \ell^T \mathbf{Q} \ell$$

Closed form solution

$$\ell^* = (\mathbf{C}^{-1} \mathbf{Q} + \mathbf{I})^{-1} \mathbf{y}$$

# SSL with Graphs: Generalization Bounds

Bounding **transductive** error

$$\ell^* = \min_{\ell \in \mathbb{R}^N} (\ell - \mathbf{y})^T \mathbf{C} (\ell - \mathbf{y}) + \ell^T \mathbf{Q} \ell$$

Think about **stability** of this solution.

Consider two datasets differing in exactly one *labeled* point.

$$\mathcal{C}_1 = \mathbf{C}_1^{-1} \mathbf{Q} + \mathbf{I} \text{ and } \mathcal{C}_2 = \mathbf{C}_2^{-1} \mathbf{Q} + \mathbf{I}$$

What is the maximal difference in the solutions?

$$\begin{aligned} \ell_2^* - \ell_1^* &= \mathcal{C}_2^{-1} \mathbf{y}_2 - \mathcal{C}_1^{-1} \mathbf{y}_1 \\ &= \mathcal{C}_2^{-1} (\mathbf{y}_2 - \mathbf{y}_1) - (\mathcal{C}_1^{-1} - \mathcal{C}_2^{-1}) \mathbf{y}_1 \\ &= \mathcal{C}_2^{-1} (\mathbf{y}_2 - \mathbf{y}_1) - (\mathcal{C}_1^{-1} [(\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}) \mathbf{Q}] \mathcal{C}_2^{-1}) \mathbf{y}_1 \end{aligned}$$

Note that  $\mathbf{v} \in \mathbb{R}^{N \times 1}$ ,  $\lambda_m(A) \|\mathbf{v}\|_2 \leq \|\mathbf{A}\mathbf{v}\|_2 \leq \lambda_M(A) \|\mathbf{v}\|_2$

$$\|\ell_2^* - \ell_1^*\|_2 \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\lambda_m(\mathcal{C}_2)} + \frac{\lambda_M(\mathbf{Q}) \|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\lambda_m(\mathcal{C}_2) \lambda_m(\mathcal{C}_1)}$$

# SSL with Graphs: Generalization Bounds

Bounding **transductive** error

$$\ell^* = \min_{\ell \in \mathbb{R}^N} (\ell - \mathbf{y})^T \mathbf{C}(\ell - \mathbf{y}) + \ell^T \mathbf{Q} \ell$$

$$\|\ell_2^* - \ell_1^*\|_2 \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\lambda_m(\mathbf{C}_2)} + \frac{\lambda_M(\mathbf{Q}) \|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\lambda_m(\mathbf{C}_2) \lambda_m(\mathbf{C}_1)}$$

Using  $\lambda_m(\mathbf{C}) \geq \frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1$

$$\|\ell_2^* - \ell_1^*\|_2 \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1} + \frac{\lambda_M(\mathbf{Q}) \|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_2)} + 1\right) \left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1\right)}$$

# SSL with Graphs: Generalization Bounds

Bounding **transductive** error

$$\|\ell_2^* - \ell_1^*\|_\infty \leq \beta \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1} + \frac{\lambda_M(\mathbf{Q}) \|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_2)} + 1\right) \left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1\right)}$$

Now, let us plug in the values for our problem.

Take  $c_l = 1$  and  $c_l > c_u$ . We have  $|y_i| \leq 1$  and  $|\ell_i^*| \leq 1$ .

$$\beta \leq 2 \left[ \frac{\sqrt{2}}{\lambda_m(\mathbf{Q}) + 1} + \sqrt{2n_l} \frac{1 - c_u}{c_u} \frac{\lambda_M(\mathbf{Q})}{(\lambda_m(\mathbf{Q}) + 1)^2} \right]$$

$\mathbf{Q}$  is reg.  $\mathbf{L}$ :  $\lambda_m(\mathbf{Q}) = \lambda_m(\mathbf{L}) + \gamma_g$  and  $\lambda_M(\mathbf{Q}) = \lambda_M(\mathbf{L}) + \gamma_g$

$$\beta \leq 2 \left[ \frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l} \frac{1 - c_u}{c_u} \frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1} \right]$$

This algorithm is  $\beta$ -stable!

# SSL with Graphs: Generalization Bounds

## Bounding **transductive** error

[http://web.cse.ohio-state.edu/~mbelkin/papers/RSS\\_COLT\\_04.pdf](http://web.cse.ohio-state.edu/~mbelkin/papers/RSS_COLT_04.pdf)

By the generalization bound of Belkin [BMN04]

$$R_P(\ell^*) \leq \hat{R}_P(\ell^*) + \underbrace{\beta + \sqrt{\frac{2 \ln(2/\delta)}{n_l}} (n_l \beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$
$$\beta \leq 2 \left[ \frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l} \frac{1 - c_u}{c_u} \frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1} \right]$$

holds with probability  $1 - \delta$ , where

$$R_P(\ell^*) = \frac{1}{N} \sum_i (\ell_i^* - y_i)^2$$
$$\hat{R}_P(\ell^*) = \frac{1}{n_l} \sum_{i \in I} (\ell_i^* - y_i)^2.$$

# SSL with Graphs: Generalization Bounds

## Bounding **transductive** error

$$R_P(\ell^*) \leq \hat{R}_P(\ell^*) + \underbrace{\beta + \sqrt{\frac{2 \ln(2/\delta)}{n_I}}}_{\text{transductive error } \Delta_T(\beta, n_I, \delta)} (n_I \beta + 4)$$
$$\beta \leq 2 \left[ \frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_I} \frac{1 - c_u}{c_u} \frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1} \right]$$

Does the bound say anything useful?

- 1) The error is controlled.
- 2) Practical when error  $\Delta_T(\beta, n_I, \delta)$  decreases at rate  $O(n_I^{-\frac{1}{2}})$ .  
Achieved when  $\beta = O(1/n_I)$ . That is,  $\gamma_g = \Omega(n_I^{\frac{3}{2}})$ .

We have an idea how to set  $\gamma_g$ !

# SSL with Graphs: Generalization Bounds

Combining **inductive** + **transductive** error

With probability  $1 - (\eta + \delta)$ .

$$R_P(f) \leq \frac{1}{n} \sum_i \mathcal{L}(f(\mathbf{x}_i), \text{sgn}(\ell_i^*)) + \widehat{R}_P(\ell^*) + \Delta_T(\beta, n_I, \delta) + \Delta_I(h, N, \eta)$$

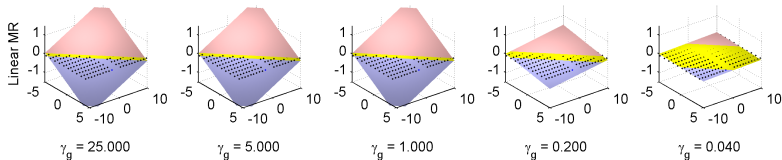
We need to account for  $\varepsilon$ . With probability  $1 - (\eta + \delta)$ .

$$R_P(f) \leq \frac{1}{n} \sum_{i: |\ell_i^*| \geq \varepsilon} \mathcal{L}(f(\mathbf{x}_i), \text{sgn}(\ell_i^*)) + \frac{2\varepsilon n_\varepsilon}{N} + \widehat{R}_P(\ell^*) + \Delta_T(\beta, n_I, \delta) + \Delta_I(h, N, \eta)$$

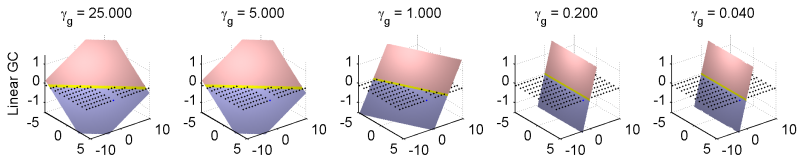
We should have  $\varepsilon \leq n_I^{-1/2}$ !

# SSL with Graphs: LapSVMs and MM Graph Cuts

MR for 2D data and **linear**  $\mathcal{K}$  only changes the slope



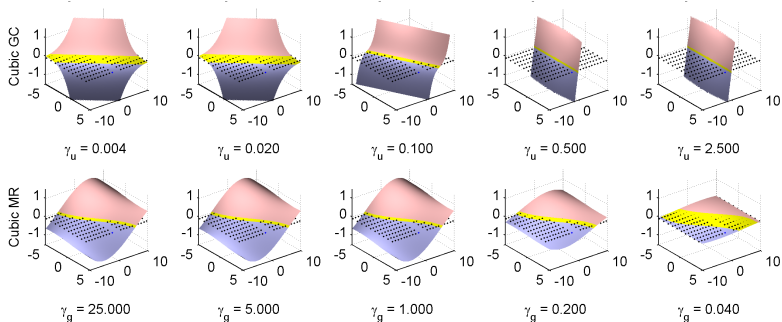
MMGC for 2D data and **linear**  $\mathcal{K}$  works as we want





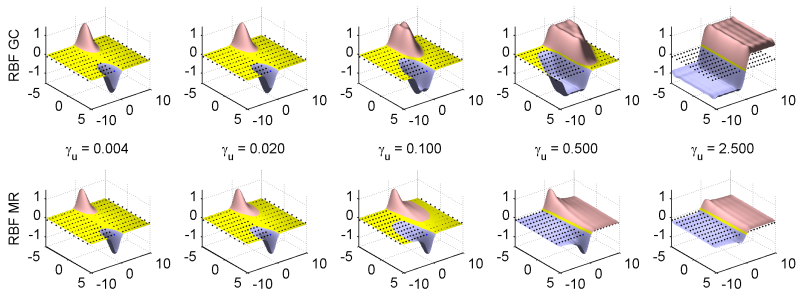
# SSL with Graphs: LapSVMs and MM Graph Cuts

MR for 2D data and **cubic**  $\mathcal{K}$  is also not so good

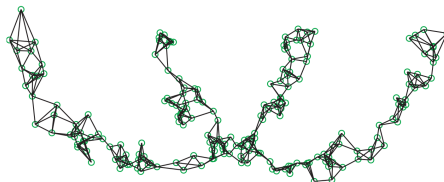


# SSL with Graphs: LapSVMs and MM Graph Cuts

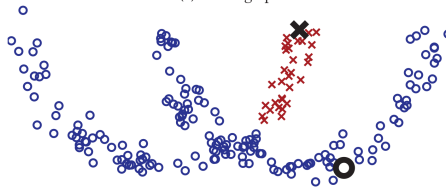
MMGC and MR for 2D data and RBF  $\mathcal{K}$



# SSL with Graphs



(a) 4-NN graph



(b) Harmonic function predictions

Graph-based SSL is obviously sensitive to graph construction!

# Online SSL with Graphs

## Offline learning setup

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**).



## Online learning setup

At the beginning:  $\{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$  from  $\mathbb{R}^d$

At time  $t$ :

receive  $\mathbf{x}_t$

predict  $y_t$

# Online SSL with Graphs

---

## Online HFS: Straightforward solution

---

- 1: **while** new unlabeled example  $\mathbf{x}_t$  comes **do**
- 2: Add  $\mathbf{x}_t$  to graph  $G(\mathbf{W})$
- 3: Update  $\mathbf{L}_t$
- 4: Infer labels

$$\mathbf{f}_u = (\mathbf{L}_{uu} + \gamma_g \mathbf{I})^{-1} (\mathbf{W}_{ul} \mathbf{f}_l)$$

- 5: Predict  $\hat{y}_t = \text{sgn}(\mathbf{f}_u(t))$
  - 6: **end while**
- 

What is wrong with this solution?

The cost and memory of the operations.

What can we do?

## Online SSL with Graphs

Let's keep only  $k$  vertices!

Limit memory to  $k$  **centroids** with  $\widetilde{\mathbf{W}}^q$  weights.

Each centroid represents *several* others.

Diagonal  $\mathbf{V} \equiv$  **multiplicity**. We have  $\mathbf{V}_{ii}$  copies of centroid  $i$ .

Can we compute it compactly? Compact harmonic solution.

$$\ell^q = (\mathbf{L}_{uu}^q + \gamma_g V)^{-1} \mathbf{W}_{ul}^q \ell_l \quad \text{where} \quad \mathbf{W}^q = V \widetilde{\mathbf{W}}^q V$$

Proof? Using electric circuits.

Why do we keep the multiplicities?

# Online SSL with Graphs

---

## Online HFS with Graph Quantization

---

- 1: **Input**
  - 2:  $k$  number of representative nodes
  - 3: **Initialization**
  - 4:  $\mathbf{V}$  matrix of multiplicities with 1 on diagonal
  - 5: **while** new unlabeled example  $\mathbf{x}_t$  comes **do**
  - 6: Add  $\mathbf{x}_t$  to graph  $G$
  - 7: **if** # nodes  $> k$  **then**
  - 8:     **quantize**  $G$
  - 9: **end if**
  - 10: Update  $\mathbf{L}_t$  of  $G(\mathbf{V}\mathbf{W}\mathbf{V})$
  - 11: Infer labels
  - 12: Predict  $\hat{y}_t = \text{sgn}(\mathbf{f}_u(t))$
  - 13: **end while**
-

# Online SSL with Graphs: Graph Quantization

An idea: incremental  $k$ -centers

Doubling algorithm of Charikar et al. [Cha+97]

Keeps up to  $k$  centers  $C_t = \{\mathbf{c}_1, \mathbf{c}_2, \dots\}$  with

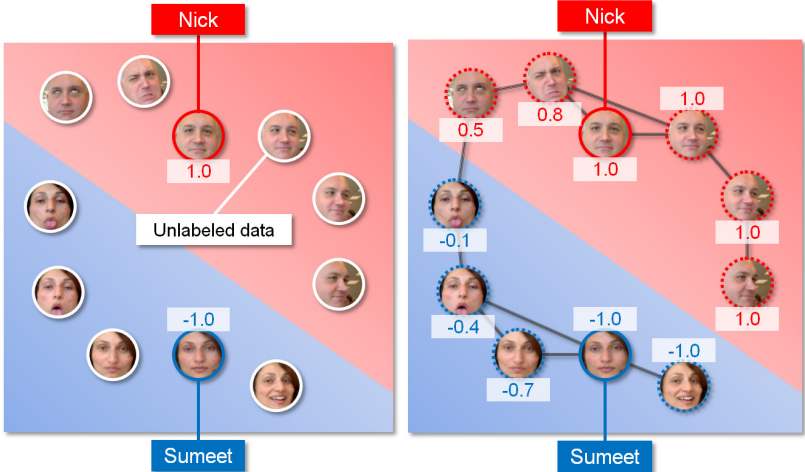
- ▶ Distance  $\mathbf{c}_i, \mathbf{c}_j \in C_t$  is at least  $\geq R$
- ▶ For each new  $\mathbf{x}_t$ , distance to some  $\mathbf{c}_i \in C_t$  is less than  $R$ .
- ▶  $|C_t| \leq k$
- ▶ if not possible,  $R$  is doubled



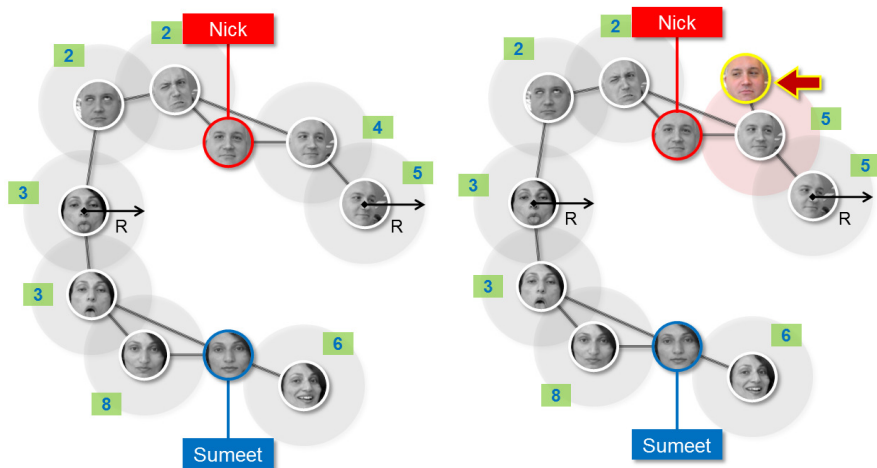
# Online SSL with Graphs: Graph Quantization



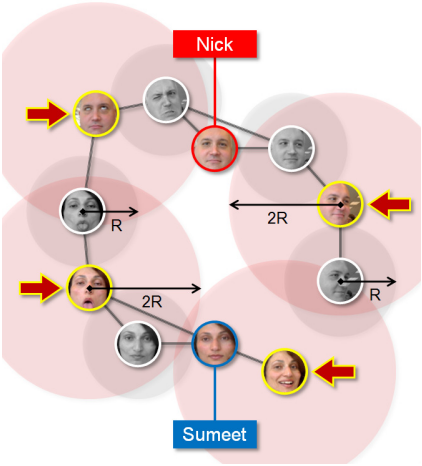
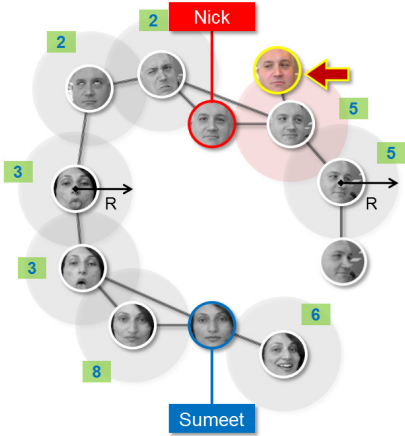
# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization

Doubling algorithm [Cha+97]

To reduce growth of  $R$ , we use  $R \leftarrow m \times R$ , with  $m \geq 1$

$C_t$  is changing. How far can  $\mathbf{x}$  be from some  $\mathbf{c}$ ?

$$R + \frac{R}{m} + \frac{R}{m^2} + \dots = R \left( 1 + \frac{1}{m} + \frac{1}{m^2} + \dots \right) = \frac{Rm}{m-1}$$

Guarantees:  $(1 + \varepsilon)$ -approximation algorithm.

Why not incremental  $k$ -means?

# Online SSL with Graphs: Graph Quantization

---

## Online $k$ -centers

---

- 1: an unlabeled  $\mathbf{x}_t$ , a set of centroids  $C_{t-1}$ , multiplicities  $\mathbf{v}_{t-1}$
  - 2: **if** ( $|C_{t-1}| = k + 1$ ) **then**
  - 3:      $R \leftarrow mR$
  - 4:     greedily repartition  $C_{t-1}$  into  $C_t$  such that:
  - 5:         no two vertices in  $C_t$  are closer than  $R$
  - 6:         for any  $\mathbf{c}_i \in C_{t-1}$  exists  $\mathbf{c}_j \in C_t$  such that  $d(\mathbf{c}_i, \mathbf{c}_j) < R$
  - 7:     update  $\mathbf{v}_t$  to reflect the new partitioning
  - 8: **else**
  - 9:      $C_t \leftarrow C_{t-1}$
  - 10:     $\mathbf{v}_t \leftarrow \mathbf{v}_{t-1}$
  - 11: **end if**
  - 12: **if**  $\mathbf{x}_t$  is closer than  $R$  to any  $\mathbf{c}_i \in C_t$  **then**
  - 13:      $\mathbf{v}_t(i) \leftarrow \mathbf{v}_t(i) + 1$
  - 14: **else**
  - 15:      $\mathbf{v}_t(|C_t| + 1) \leftarrow 1$
  - 16: **end if**
-

# Online SSL with Graphs

## Video examples

<http://www.bkveton.com/videos/Coffee.mp4>

<http://www.bkveton.com/videos/Ad.mp4>

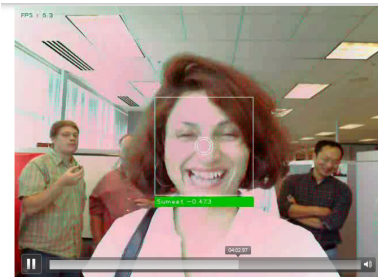
<http://researchers.lille.inria.fr/~valko/hp/serve.php?what=publications/kveton2009nipsdemo.adaptation.mov>

<http://researchers.lille.inria.fr/~valko/hp/serve.php?what=publications/kveton2009nipsdemo.officespace.mov>

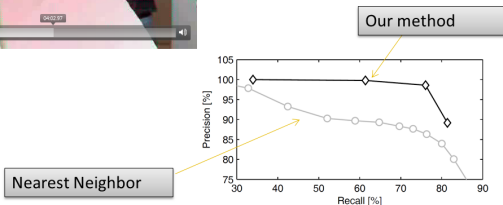
<http://bcove.me/a2derjeh>

or: <http://researchers.lille.inria.fr/~valko/hp/publications/press-intel-2015.mp4>

# SSL with Graphs: Some experimental results



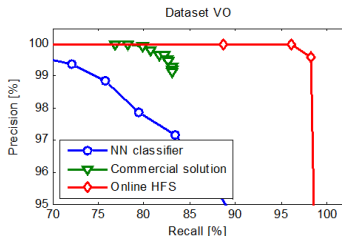
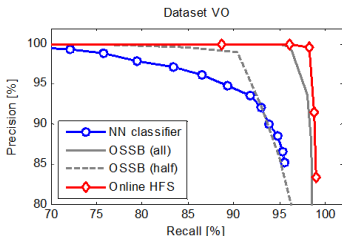
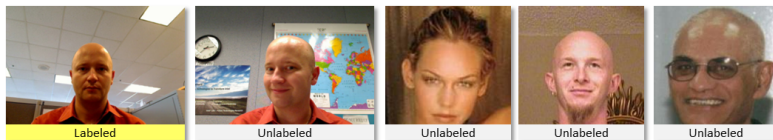
- 8 people classification
- Making funny faces
- 4 faces/person are labeled





# SSL with Graphs: Some experimental results

- One person moves among various indoor locations
- 4 labeled examples of a person in the cubicle

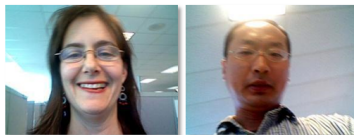


Online HFS outperforms OSSB (even when the weak learners are chosen using future data)

Online HFS yields better results than a commercial solution at 20% of the computational cost

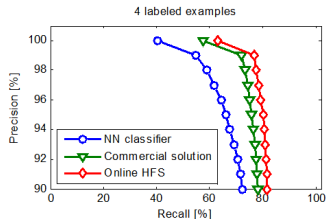
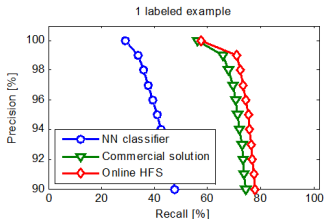
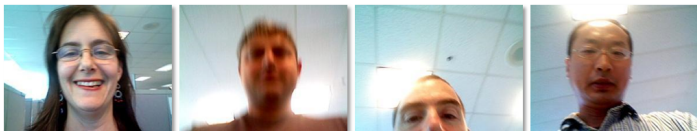
# SSL with Graphs: Some experimental results

- **Logging in** with faces instead of password
- Able to **learn** and improve



# SSL with Graphs: Some experimental results

- 16 people log twice into a tablet PC at 10 locations



Online HFS yields better results than a commercial solution at 20% of the computational cost

*Michal Valko*

michal.valko@inria.fr

ENS Paris-Saclay, MVA 2017/2018

Sequel team, Inria Lille — Nord Europe

<https://team.inria.fr/sequel/>