# Graphs in Machine Learning

Daniele Calandriello

*DeepMind Paris, France*

Collaborators: Achraf Azize,
Michal Valko

Partially based on material by: Mikhail Belkin,
Jerry Zhu, Olivier Chapelle, Branislav Kveton

13 Feb, 2021

**MVA 2022/2023**

# Previous lecture

- manifold learning with Laplacians eigenmaps
- inductive and transductive semi-supervised learning
- graph-based semi-supervised learning
- graph based manifold regularization

# This lecture Deeper in the theory

- transductive learning stability based bounds
- analysis of inductive and transductive semi-supervised learning
- online semi-supervised Learning
- online incremental $k$-centers
- Analysis of online SSL
- SSL learnability
- When does graph-based SSL provably help?

# SSL($\mathcal{G}$)

## semi-supervised learning with graphs and harmonic functions

...our running example for learning with graphs

# Checkpoint 2

Unconstrained regularization in general ($\mathbf{C} = diag(\{c_l, c_u\}), \mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$):

Transductive case
with $N$ datapoints
($n_l$ labelled and $n_u$ unlabelled)

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

# Checkpoint 2

+1
-1
0

Unconstrained regularization in general ($\mathbf{C} = diag(\{c_l, c_u\}), \mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$):

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

diag

contain laplacian and $\gamma_g$ that are regularizer

Out of sample extension: Laplacian SVMs

↳ predict new datapoints that are arriving

$$f^\star = \arg\min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i}^{n_l} \max(0, 1 - y f(\mathbf{x})) + \gamma_1 \|f\|_\mathcal{K}^2 + \gamma_2 \mathbf{f}^\mathsf{T} \mathbf{L} \mathbf{f}$$

# Transductive Generalization Bounds

check stability

True risk vs. empirical risk

$$\text{true risk} \rightarrow R_N(f) = \frac{1}{N}\sum_i (f_i - y_i)^2$$

$$\text{empirical risk} \rightarrow \widehat{R}_N(f) = \frac{1}{n_l}\sum_{i\in l} (f_i - y_i)^2$$

# Transductive Generalization Bounds

True risk vs. empirical risk

$$
\begin{aligned}
R_N(f) &= \frac{1}{N}\sum_i (f_i - y_i)^2 \\
\widehat{R}_N(f) &= \frac{1}{n_l}\sum_{i \in l} (f_i - y_i)^2
\end{aligned}
$$

We look for **transductive** bounds in the form

$$
\underbrace{R_N(f) \le \widehat{R}_N(f) + \text{errors}}_{\text{what we want: empirical risk} \approx \text{true risk}}
$$

# Transductive Generalization Bounds

Bounding transductive error using stability analysis (in our case Belkin [BMN04])

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

http://web.cse.ohio-state.edu/~mbelkin/papers/RSS_COLT_04.pdf

# Transductive Generalization Bounds

Bounding transductive error using stability analysis (in our case Belkin [BMN04])

http://www.cs.nyu.edu/~mohri/pub/str.pdf http://web.cse.ohio-state.edu/~mbelkin/papers/RSS_COLT_04.pdf

one label point is diff

(Informal) let $\beta$ be the maximal difference $\|\ell_2^\star - \ell_1^\star\|_\infty \leq \beta$ between solutions

$$\ell_1^\star = (\mathbf{C}_1^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}_1 \text{ and } \ell_2^\star = (\mathbf{C}_2^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}_2$$

for two datasets differing in exactly one *labeled* point. Then w.p. $1 - \delta$

the stats are coming from the $n_l$ labeled points

# Transductive Generalization Bounds

Bounding transductive error using stability analysis (in our case Belkin [BMN04])

http://www.cs.nyu.edu/~mohri/pub/str.pdf http://web.cse.ohio-state.edu/~mbelkin/papers/RSS_COLT_04.pdf

(Informal) let $\beta$ be the maximal difference $\|\ell_2^\star - \ell_1^\star\|_\infty \leq \beta$ between solutions

$$\ell_1^\star = (\mathbf{C}_1^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}_1 \text{ and } \ell_2^\star = (\mathbf{C}_2^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}_2$$

for two datasets differing in exactly one *labeled* point. Then w.p. $1 - \delta$

stability considerate, if b big is not stable

$$R_N(\ell^\star) \leq \widehat{R}_N(\ell^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)} .$$

# Transductive Generalization Bounds

Stability analysis from last class

$$\|\boldsymbol{\ell}_2^\star - \boldsymbol{\ell}_1^\star\|_\infty$$

# Transductive Generalization Bounds

Stability analysis from last class

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

the stability depends on the labels, Laplacian and symmetry

# Transductive Generalization Bounds

Stability analysis from last class

$$\|\boldsymbol{\ell}_2^\star - \boldsymbol{\ell}_1^\star\|_\infty \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1} + \frac{\lambda_M(\mathbf{Q})\|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_2)} + 1\right)\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1\right)}$$

stability

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

reg of laplacian

# Transductive Generalization Bounds

Stability analysis from last class

$$\|\boldsymbol{\ell}_2^\star - \boldsymbol{\ell}_1^\star\|_\infty \leq \frac{\|\mathbf{y}_2 - \mathbf{y}_1\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1} + \frac{\lambda_M(\mathbf{Q})\|\mathbf{C}_1^{-1} - \mathbf{C}_2^{-1}\|_2 \cdot \|\mathbf{y}_1\|_2}{\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_2)} + 1\right)\left(\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C}_1)} + 1\right)}$$

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

Soft HFS is $\beta$-stable!

# Transductive Generalization Bounds

$$R_N(\boldsymbol{\ell}^\star) \;\leq\; \widehat{R}_N(\boldsymbol{\ell}^\star) + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4), \quad \beta \;\leq\; 2\left[\frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l}\frac{1 - c_u}{c_u}\frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1}\right]$$

Does the bound say anything useful?

# Transductive Generalization Bounds

$$R_N(\ell^\star) \;\leq\; \widehat{R}_N(\ell^\star) + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4), \quad \beta \;\leq\; 2\left[\frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l}\frac{1-c_u}{c_u}\frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1}\right]$$

Does the bound say anything useful?

**1)** The error is controlled. as it is not infinite we can regularize a lot or get more data

# Transductive Generalization Bounds

$$R_N(\boldsymbol{\ell}^\star) \;\leq\; \widehat{R}_N(\boldsymbol{\ell}^\star) + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4), \quad \beta \;\leq\; 2\left[\frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l}\frac{1-c_u}{c_u}\frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1}\right]$$

Does the bound say anything useful?

**1)** The error is controlled.

**2)** Practical when error $\Delta_T(\beta, n_l, \delta)$ decreases at rate $O(n_l^{-\frac{1}{2}})$

# Transductive Generalization Bounds

$$R_N(\boldsymbol{\ell}^\star) \;\leq\; \widehat{R}_N(\boldsymbol{\ell}^\star) + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4), \quad \beta \;\leq\; 2\left[\frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l}\frac{1 - c_u}{c_u}\frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1}\right]$$

Does the bound say anything useful?

**1)** The error is controlled.

**2)** Practical when error $\Delta_T(\beta, n_l, \delta)$ decreases at rate $O(n_l^{-\frac{1}{2}})$

↳ Achieved when $\beta = O(1/n_l)$. That is, $\gamma_g = \Omega(n_l^{\frac{3}{2}})$.

# Transductive Generalization Bounds

$$R_N(\ell^\star) \leq \widehat{R}_N(\ell^\star) + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4), \quad \beta \leq 2\left[\frac{\sqrt{2}}{\gamma_g + 1} + \sqrt{2n_l}\frac{1-c_u}{c_u}\frac{\lambda_M(\mathbf{L}) + \gamma_g}{\gamma_g^2 + 1}\right]$$

Does the bound say anything useful?

**1)** The error is controlled.

**2)** Practical when error $\Delta_T(\beta, n_l, \delta)$ decreases at rate $O(n_l^{-\frac{1}{2}})$

↳ Achieved when $\beta = O(1/n_l)$. That is, $\gamma_g = \Omega(n_l^{\frac{3}{2}})$.

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

What about inductive error?

# SSL with Graphs: Laplacian SVMs

$$f^\star = \arg\min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i}^{n_l} \max\left(0, 1 - yf\left(\mathbf{x}\right)\right) + \gamma_1 \|f\|_\mathcal{K}^2 + \gamma_2 \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

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

# SSL with Graphs: Laplacian SVMs

$$f^\star = \arg\min_{f\in\mathcal{H}_\mathcal{K}} \sum_i^{n_l} \max\left(0, 1 - yf\left(\mathbf{x}\right)\right) + \gamma_1 \|f\|_\mathcal{K}^2 + \gamma_2 \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

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

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

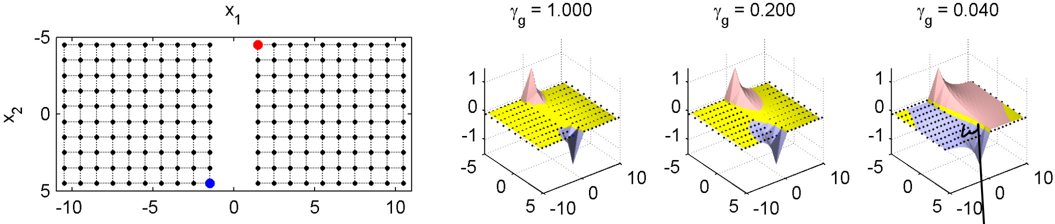
# SSL with Graphs: Laplacian SVMs

$$f^{\star} = \arg\min_{f \in \mathcal{H}_{\mathcal{K}}} \sum_{i}^{n_l} \max\left(0, 1 - yf\left(\mathbf{x}\right)\right) + \gamma_1 \|f\|_{\mathcal{K}}^2 + \gamma_2 \mathbf{f}^{\mathsf{T}} \mathbf{L} \mathbf{f}$$

$\mathcal{H}_{\mathcal{K}}$ 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 $\mathbf{f}^{\mathsf{T}}\mathbf{L}\mathbf{f}$ can perform badly.



# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$. → space of function with 2 param we like to learn

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1, \alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1$$

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^{\mathsf{T}} \mathbf{L} \mathbf{f}$$

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

For this simple case (only vertical/horizontal edges) we can write down $\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$ explicitly.

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

For this simple case (only vertical/horizontal edges) we can write down $\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$ explicitly.

$$\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1, \alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^\mathsf{T} \mathbf{L} \mathbf{f}$$

For this simple case (only vertical/horizontal edges) we can write down $\mathbf{f}^\mathsf{T} \mathbf{L} \mathbf{f}$ explicitly.

$$\mathbf{f}^\mathsf{T} \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} (f(\mathbf{x}_i) - f(\mathbf{x}_j))^2$$

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

For this simple case (only vertical/horizontal edges) we can write down $\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$ explicitly.

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

# SSL with Graphs: Laplacian SVMs

Linear $\mathcal{K} \equiv$ functions $f(x_i) = \alpha_1 x_{i1} + \alpha_2 x_{i2}$.

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \gamma_1 \left[\alpha_1^2 + \alpha_2^2\right] + \gamma_2 \mathbf{f}^\mathsf{T} \mathbf{L} \mathbf{f}$$

For this simple case (only vertical/horizontal edges) we can write down $\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$ explicitly.

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

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right) [\alpha_1^2 + \alpha_2^2]$$

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right) [\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)[\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \overline{\gamma}[\alpha_1^2 + \alpha_2^2]$$

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)[\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \overline{\gamma}[\alpha_1^2 + \alpha_2^2]$$

What does this objective function correspond to?

the laplacian should add smoothness but it fails here as $\Delta \simeq$ cst

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right) [\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \overline{\gamma}[\alpha_1^2 + \alpha_2^2]$$

What does this objective function correspond to?

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)[\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \overline{\gamma}[\alpha_1^2 + \alpha_2^2]$$

What does this objective function correspond to?

The only influence of unlabeled data is through $\overline{\gamma}$.

# SSL with Graphs: Laplacian SVMs

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

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)[\alpha_1^2 + \alpha_2^2]$$

Setting $\overline{\gamma} = \left(\gamma_1 + \frac{\gamma_2 \Delta}{2}\right)$:

$$\min_{\alpha_1,\alpha_2} \sum_{i}^{n_l} V(f, \mathbf{x}_i, y_i) + \overline{\gamma}[\alpha_1^2 + \alpha_2^2]$$

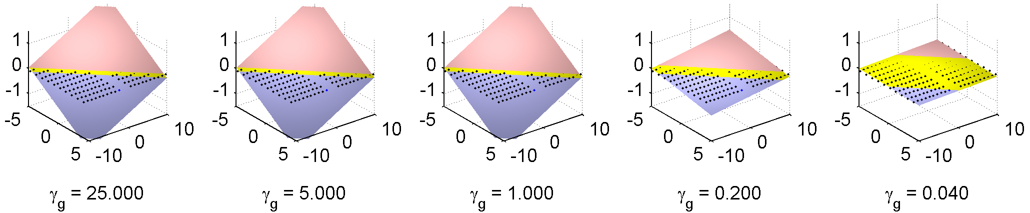
What does this objective function correspond to?

The only influence of unlabeled data is through $\overline{\gamma}$.

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

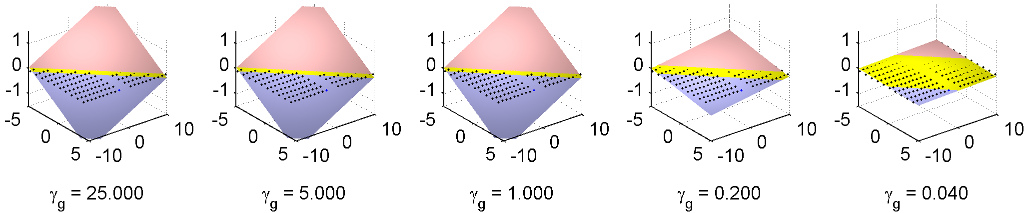
# SSL with Graphs: Laplacian SVMs

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



# SSL with Graphs: Laplacian SVMs

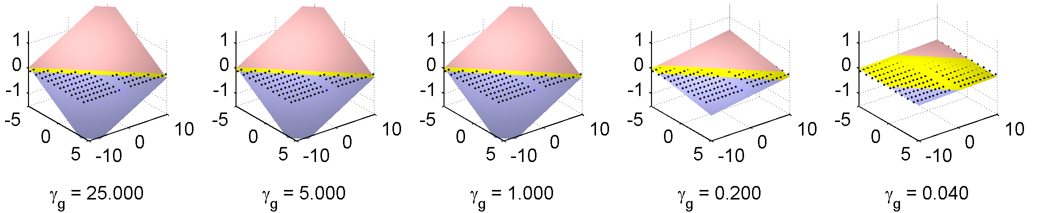
LSVM for 2D data and linear $\mathcal{K}$ only changes the slope



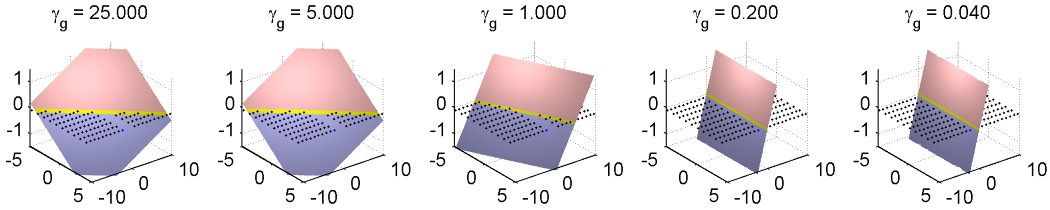

What would we like to see?

# SSL with Graphs: Laplacian SVMs

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

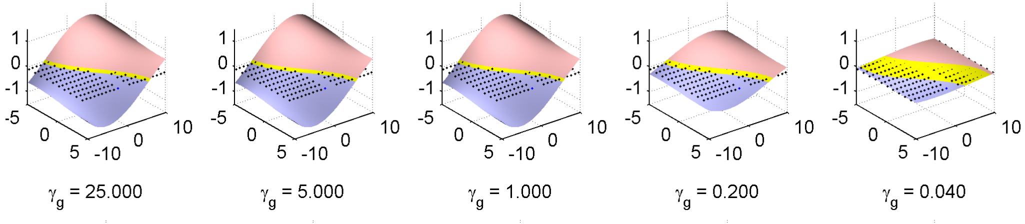



What would we like to see?



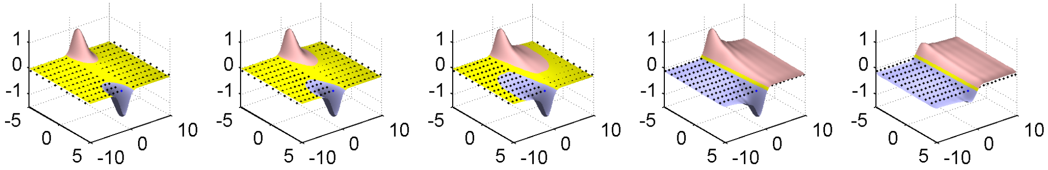
# SSL with Graphs: LapSVMs

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



# SSL with Graphs: LapSVMs

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



# SSL with Graphs: Max-Margin Graph Cuts

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

# SSL with Graphs: Max-Margin Graph Cuts

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

involve more thing in error.

$$f^\star = \min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i : |\ell_i^\star| \geq \varepsilon} V(f, \mathbf{x}_i, \mathrm{sgn}(\ell_i^\star)) + \gamma \|f\|_\mathcal{K}^2$$

$$\text{s.t.} \quad \boldsymbol{\ell}^\star = \arg\min_{\boldsymbol{\ell} \in \mathbb{R}^N} \boldsymbol{\ell}^\mathsf{T} (\mathbf{L} + \gamma_g \mathbf{I}) \boldsymbol{\ell}$$

$$\text{s.t. } \ell_i = y_i \text{ for all } i = 1, \ldots, n_l$$

if we have probability to belong to a class we put it in our problem.

# SSL with Graphs: Max-Margin Graph Cuts

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

$$f^\star = \min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i: |\ell_i^\star| \geq \varepsilon} V(f, \mathbf{x}_i, \operatorname{sgn}(\ell_i^\star)) + \gamma \|f\|_\mathcal{K}^2$$

$$\text{s.t.} \quad \boldsymbol{\ell}^\star = \arg\min_{\boldsymbol{\ell} \in \mathbb{R}^N} \boldsymbol{\ell}^\mathsf{T} (\mathbf{L} + \gamma_g \mathbf{I}) \boldsymbol{\ell}$$

$$\text{s.t. } \ell_i = y_i \text{ for all } i = 1, \dots, n_l$$

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

# SSL with Graphs: Max-Margin Graph Cuts

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

$$f^\star = \min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i : |\ell_i^\star| \geq \varepsilon} V(f, \mathbf{x}_i, \mathrm{sgn}(\ell_i^\star)) + \gamma \|f\|_\mathcal{K}^2$$

$$\text{s.t.} \quad \boldsymbol{\ell}^\star = \arg\min_{\boldsymbol{\ell} \in \mathbb{R}^N} \boldsymbol{\ell}^\mathsf{T}(\mathbf{L} + \gamma_g \mathbf{I})\boldsymbol{\ell}$$

$$\text{s.t. } \ell_i = y_i \text{ for all } i = 1, \dots, n_l$$

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

Will we get into the same trouble?

# SSL with Graphs: Max-Margin Graph Cuts

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

$$f^\star = \min_{f \in \mathcal{H}_\mathcal{K}} \sum_{i:|\ell_i^\star| \geq \varepsilon} V(f, \mathbf{x}_i, \operatorname{sgn}(\ell_i^\star)) + \gamma \|f\|_\mathcal{K}^2$$

$$\text{s.t.} \quad \boldsymbol{\ell}^\star = \arg\min_{\boldsymbol{\ell} \in \mathbb{R}^N} \boldsymbol{\ell}^\mathsf{T}(\mathbf{L} + \gamma_g \mathbf{I})\boldsymbol{\ell}$$

$$\text{s.t. } \ell_i = y_i \text{ for all } i = 1, \dots, n_l$$

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

Will we get into the same trouble?

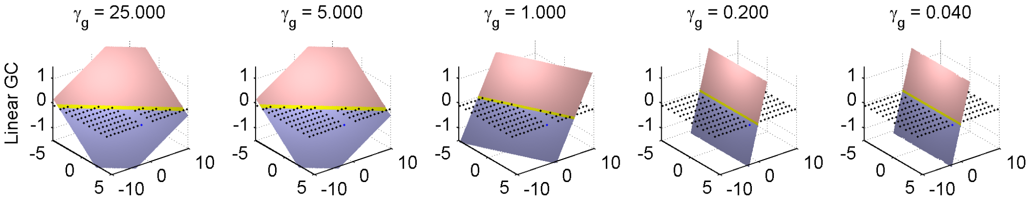
Representer theorem is still cool:

confidence?

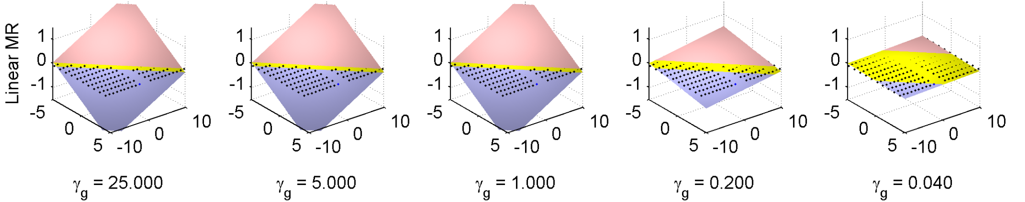
$$f^\star(\mathbf{x}) = \sum_{i:|f_i^\star| \geq \varepsilon} \alpha_i^\star \mathcal{K}(\mathbf{x}_i, \mathbf{x})$$

# SSL with Graphs: LapSVMs and MM Graph Cuts

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

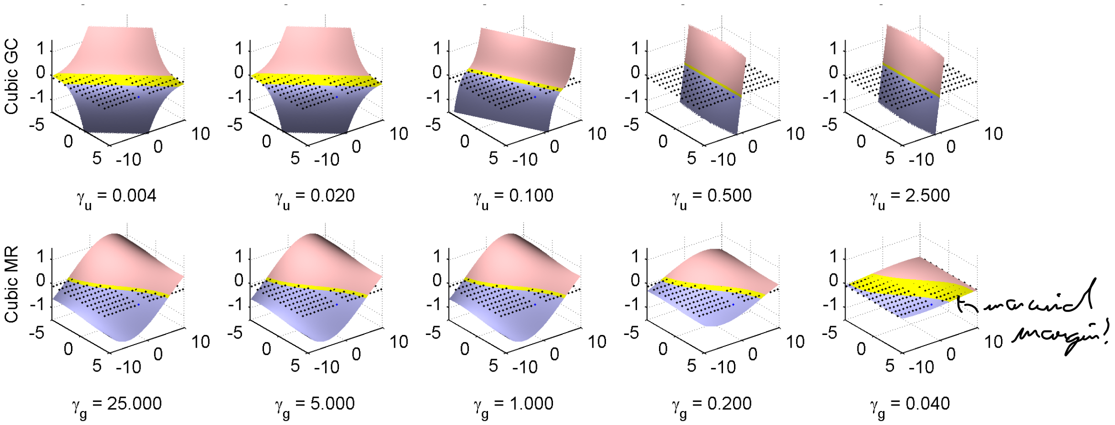



LapSVM for 2D data and **linear** $\mathcal{K}$ only changes the slope

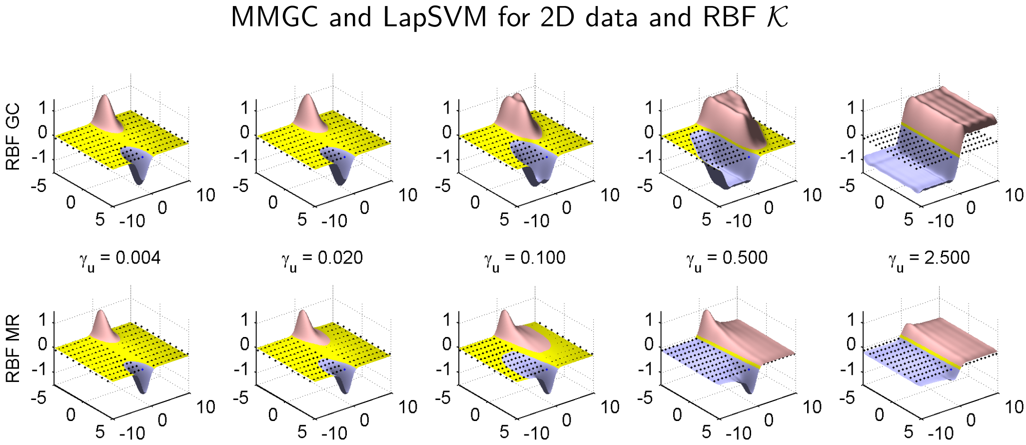


# SSL with Graphs: LapSVMs and MM Graph Cuts

MMGC for 2D data and **cubic** $\mathcal{K}$ is good



# SSL with Graphs: LapSVMs and MM Graph Cuts



# Inductive Generalization Bounds

We may want to bound the **classification risk** coming from laplacian reg graph

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

↳ classif. loss

Three steps:

1) From expected to empirical error (SLT)
2) Convexify classification loss into squared loss
3) From empirical loss to inductive bound

# Inductive Generalization Bounds

Using classical SLT tools (Equations 3.15 and 3.24 [Vap95]), with probability $1 - \eta$

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

# Inductive Generalization Bounds

Using classical SLT tools (Equations 3.15 and 3.24 [Vap95]), with probability $1 - \eta$

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

$N \equiv$ number of samples

# Inductive Generalization Bounds

Using classical SLT tools (Equations 3.15 and 3.24 [Vap95]), with probability $1 - \eta$

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

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

# Inductive Generalization Bounds

Using classical SLT tools (Equations 3.15 and 3.24 [Vap95]), with probability $1 - \eta$

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

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

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

# Inductive Generalization Bounds

Using classical SLT tools (Equations 3.15 and 3.24 [Vap95]), with probability $1 - \eta$

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

$N \equiv$ 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 $\mathcal{L}(f(\mathbf{x}_i), y_i)$?

# Inductive Generalization Bounds

For any $y_i \in \{-1, 1\}$ and $\ell_i^\star$

$$\frac{1}{N}\sum_i \mathcal{L}(f(\mathbf{x}_i), y_i)$$

# Inductive Generalization Bounds

For any $y_i \in \{-1, 1\}$ and $\ell_i^\star$

$$\frac{1}{N}\sum_i \mathcal{L}(f(\mathbf{x}_i), y_i)$$

loss btw SVM and GFS

d

$$\overset{\text{convexity}}{\leq} \frac{1}{N}\sum_i \mathcal{L}(f(\mathbf{x}_i), \mathrm{sgn}(\ell_i^\star)) + \underbrace{\frac{1}{N}\sum_i (\ell_i^\star - y_i)^2}_{\text{unknown...}}$$

# Inductive Generalization Bounds

For any $y_i \in \{-1, 1\}$ and $\ell_i^\star$

$$
\begin{aligned}
&\frac{1}{N} \sum_i \mathcal{L}(f(\mathbf{x}_i), y_i) \\
&\overset{\text{convexity}}{\leq} \frac{1}{N} \sum_i \mathcal{L}(f(\mathbf{x}_i), \operatorname{sgn}(\ell_i^\star)) + \frac{1}{N} \sum_i (\ell_i^\star - y_i)^2 \\
&\overset{\text{stability bound}}{\leq} \left( \frac{1}{N} \sum_i \mathcal{L}(f(\mathbf{x}_i), \operatorname{sgn}(\ell_i^\star)) \right) + \widehat{R}_P(\boldsymbol{\ell}^\star) + \Delta_T(\beta, n_l, \delta)
\end{aligned}
$$

# Inductive Generalization Bounds

**Combining inductive + transductive error**

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

$$
\begin{aligned}
R_P(f) \;\leq\; & \frac{1}{N} \sum_i \mathcal{L}(f(\mathbf{x}_i), \operatorname{sgn}(\ell_i^\star)) + \\
& \widehat{R}_P(\boldsymbol{\ell}^\star) + \Delta_T(\beta, n_l, \delta) + \Delta_I(h, N, \eta)
\end{aligned}
$$

# Inductive Generalization Bounds

**Combining inductive + transductive error**

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

$$
\begin{aligned}
R_P(f) \;\leq\; & \frac{1}{N}\sum_i \mathcal{L}(f(\mathbf{x}_i), \operatorname{sgn}(\ell_i^\star)) + \\
& \widehat{R}_P(\boldsymbol{\ell}^\star) + \Delta_T(\beta, n_l, \delta) + \Delta_I(h, N, \eta)
\end{aligned}
$$

↪ up to making sure the rest don't blow up

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

$$
\begin{aligned}
R_P(f) \leq & \frac{1}{N}\sum_{i:|\ell_i^\star|\geq\varepsilon} \mathcal{L}(f(\mathbf{x}_i), \operatorname{sgn}(\ell_i^\star)) + \frac{2\varepsilon n_\varepsilon}{N} + \\
& \widehat{R}_P(\boldsymbol{\ell}^\star) + \Delta_T(\beta, n_l, \delta) + \Delta_I(h, N, \eta)
\end{aligned}
$$

# Inductive Generalization Bounds

**Combining inductive + transductive error**

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

$$R_P(f) \;\leq\; \frac{1}{N}\sum_i \mathcal{L}(f(\mathbf{x}_i), \mathrm{sgn}(\ell_i^\star)) + \widehat{R}_P(\boldsymbol{\ell}^\star) + \Delta_T(\beta, n_l, \delta) + \Delta_I(h, N, \eta)$$

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

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

in order to get balance term

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

# SSL with Graphs: What is behind it?

Why and when it helps?

semi-supervised ↗ ↗ supervised

Can we guarantee benefit of SSL over SL?

$SL \rightarrow \frac{1}{\sqrt{n_l}}$

$SSL: \frac{1}{\sqrt{n_l}} \times \beta(L)$

can be lower than 1

# SSL with Graphs: What is behind it?

Why and when it helps?

Can we guarantee benefit of SSL over SL?

Are there cases when **manifold** SSL is provably helpful?

Say $\mathcal{X}$ is supported on manifold $\mathcal{M}$. Compare two cases:

- SL: does not know about $\mathcal{M}$ and only knows $(\mathbf{x}_i, y_i)$
- SSL: perfect knowledge of $\mathcal{M}$

SSL has access ~~to~~ the manifold of the data

# SSL with Graphs: What is behind it?

Why and when it helps?

Can we guarantee benefit of SSL over SL?

Are there cases when **manifold** SSL is provably helpful?

Say $\mathcal{X}$ is supported on manifold $\mathcal{M}$. Compare two cases:

- SL: does not know about $\mathcal{M}$ and only knows $(\mathbf{x}_i, y_i)$
- SSL: perfect knowledge of $\mathcal{M}$ $\equiv$ humongous amounts of $\mathbf{x}_i$

http://people.cs.uchicago.edu/~niyogi/papersps/ssminimax2.pdf

# SSL with Graphs: What is behind it?

Set of learning problems - collections $\mathcal{P}$ of probability distributions:

$\mathcal{P}$

# SSL with Graphs: What is behind it?

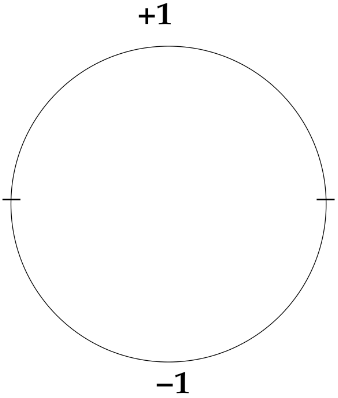
Set of learning problems - collections $\mathcal{P}$ of probability distributions:

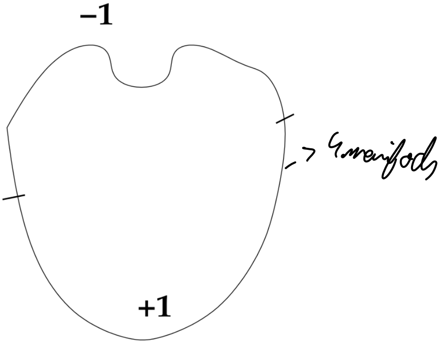
$$\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$$

# SSL with Graphs: What is behind it?

Set of learning problems - collections $\mathcal{P}$ of probability distributions:

$$\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \cup_{\mathcal{M}} \{p \in \mathcal{P} | p_{\mathcal{X}} \text{ is uniform on } \mathcal{M}\}$$



# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}} \text{ is uniform on } \mathcal{M}\}$

**Regression function** $m_p = \mathbb{E}[y | x]$ when $x \in \mathcal{M}$

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}} \text{ is uniform on } \mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}[y \,|x]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\bar{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}[y|x]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\overline{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{x}})} \right]$$

Worst manifold but best algo

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}} \text{ is uniform on } \mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}\left[y \,|x\right]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\overline{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

Since $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}\left[y \,|x\right]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\overline{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

Since $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{\mathcal{M}} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

worst point inside the manifold

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}[y | x]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\overline{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \| A(\overline{z}) - m_p \|_{L^2(p_{\mathbf{X}})} \right]$$

Since $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{\mathcal{M}} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\overline{z}} \left[ \| A(\overline{z}) - m_p \|_{L^2(p_{\mathbf{X}})} \right]$$

(SSL) When $A$ is allowed to know $\mathcal{M}$

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}[y\,|x]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\bar{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

Since $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{\mathcal{M}} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

(SSL) When $A$ is allowed to know $\mathcal{M}$

$$Q(n_l, \mathcal{P}) = \sup_{\mathcal{M}} \inf_{A} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

↳ because SSL check the manifolds

# SSL with Graphs: What is behind it?

**Set of problems** $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}} = \{p \in \mathcal{P} | p_{\mathcal{X}}$ is uniform on $\mathcal{M}\}$
**Regression function** $m_p = \mathbb{E}[y\,|x]$ when $x \in \mathcal{M}$
**Algorithm** $A$ and **labeled examples** $\bar{z} = \{z_i\}_{i=1}^{n_l} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_l}$
**Minimax rate**

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

Since $\mathcal{P} = \cup_{\mathcal{M}} \mathcal{P}_{\mathcal{M}}$

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{\mathcal{M}} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

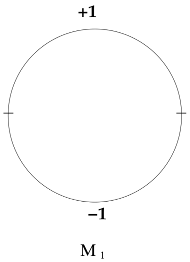
(SSL) When $A$ is allowed to know $\mathcal{M}$

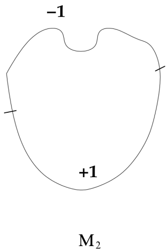
$$Q(n_l, \mathcal{P}) = \sup_{\mathcal{M}} \inf_{A} \sup_{p \in \mathcal{P}_{\mathcal{M}}} \mathbb{E}_{\bar{z}} \left[ \|A(\bar{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right]$$

In which cases there is a gap between $Q(n_l, \mathcal{P})$ and $R(n_l, \mathcal{P})$?

# SSL with Graphs: What is behind it?

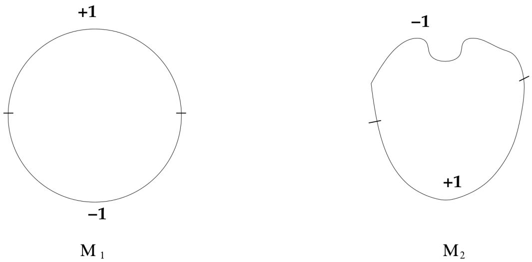
**Hypothesis space** $\mathcal{H}$: half of the circle as $+1$ and the rest as $-1$



# SSL with Graphs: What is behind it?

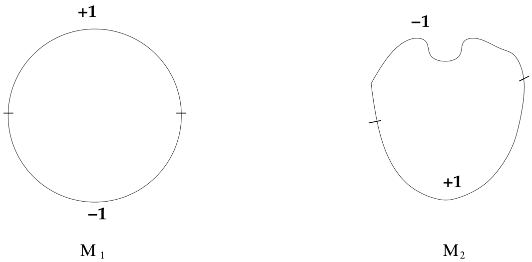
**Hypothesis space** $\mathcal{H}$: half of the circle as $+1$ and the rest as $-1$




**Case 1:** $\mathcal{M}$ is known to the learner ($\mathcal{H}_{\mathcal{M}}$)

# SSL with Graphs: What is behind it?

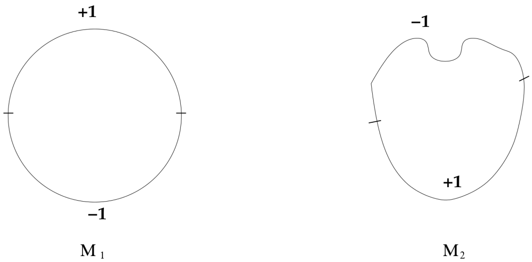
**Hypothesis space** $\mathcal{H}$: half of the circle as $+1$ and the rest as $-1$




**Case 1:** $\mathcal{M}$ is known to the learner ($\mathcal{H}_{\mathcal{M}}$) What is a VC dimension of $\mathcal{H}_{\mathcal{M}}$?

# SSL with Graphs: What is behind it?

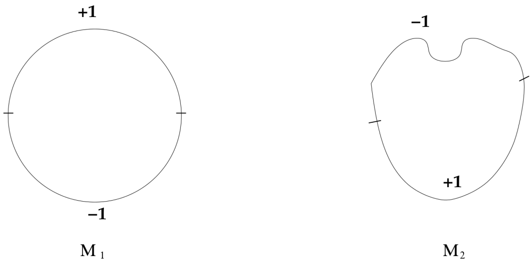
**Hypothesis space** $\mathcal{H}$: half of the circle as $+1$ and the rest as $-1$




**Case 1:** $\mathcal{M}$ is known to the learner ($\mathcal{H}_{\mathcal{M}}$) What is a VC dimension of $\mathcal{H}_{\mathcal{M}}$?

# SSL with Graphs: What is behind it?

**Hypothesis space** $\mathcal{H}$: half of the circle as $+1$ and the rest as $-1$




**Case 1:** $\mathcal{M}$ is known to the learner ($\mathcal{H}_{\mathcal{M}}$) What is a VC dimension of $\mathcal{H}_{\mathcal{M}}$?

$$\text{Optimal rate } Q(n, \mathcal{P}) \leq 2\sqrt{\frac{3 \log n_l}{n_l}}$$

# SSL with Graphs: What is behind it?

**Case 2:** $\mathcal{M}$ is **unknown** to the learner

# SSL with Graphs: What is behind it?

**Case 2:** $\mathcal{M}$ is **unknown** to the learner

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{X}})} \right] =$$

## SSL with Graphs: What is behind it?

**Case 2:** $\mathcal{M}$ is **unknown** to the learner

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \| A(\overline{z}) - m_p \|_{L^2(p_{\mathbf{X}})} \right] = \Omega(1)$$

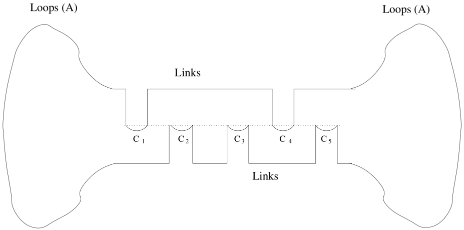
# SSL with Graphs: What is behind it?

**Case 2:** $\mathcal{M}$ is **unknown** to the learner

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \| A(\overline{z}) - m_p \|_{L^2(p_{\mathbf{X}})} \right] = \Omega(1)$$

We consider $2^d$ manifolds of the form

$$\mathcal{M} = \text{Loops} \cup \text{Links} \cup C \text{ where } C = \cup_{i=1}^{d} C_i$$



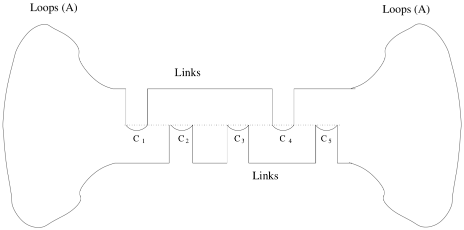
# SSL with Graphs: What is behind it?

**Case 2:** $\mathcal{M}$ is **unknown** to the learner

$$R(n_l, \mathcal{P}) = \inf_{A} \sup_{p \in \mathcal{P}} \mathbb{E}_{\overline{z}} \left[ \|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{x}})} \right] = \Omega(1)$$

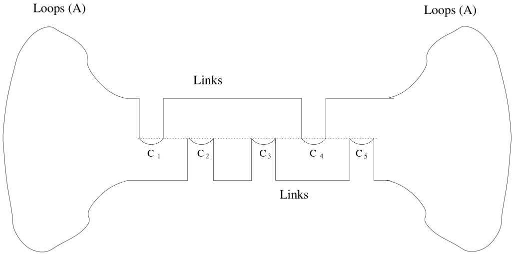
We consider $2^d$ manifolds of the form

$$\mathcal{M} = \text{Loops} \cup \text{Links} \cup C \text{ where } C = \cup_{i=1}^{d} C_i$$




**Main idea**: $d$ segments in $C$, $d - l$ with no data, $2^l$ possible choices for labels, which helps us to lower bound $\|A(\overline{z}) - m_p\|_{L^2(p_{\mathbf{x}})}$

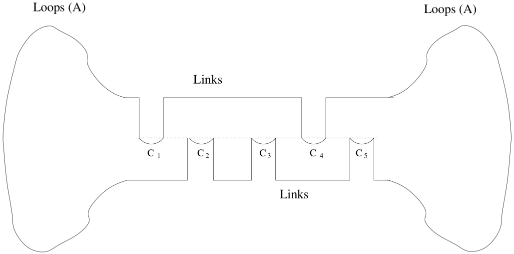
# SSL with Graphs: What is behind it?




## Knowing the manifold helps

- $C_1$ and $C_4$ are close

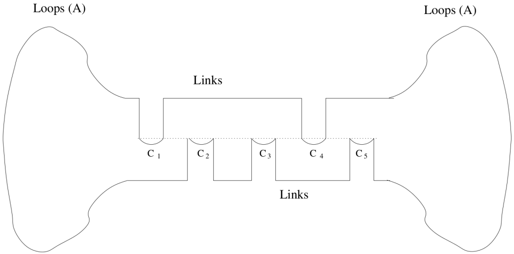
# SSL with Graphs: What is behind it?




**Knowing the manifold helps**

- $C_1$ and $C_4$ are close
- $C_1$ and $C_3$ are far

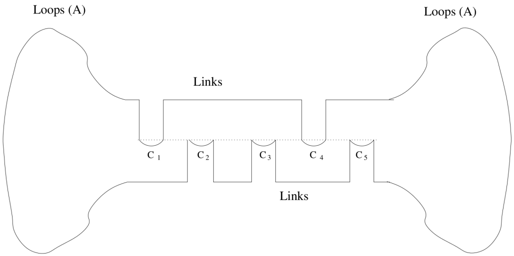
# SSL with Graphs: What is behind it?




**Knowing the manifold helps**

- $C_1$ and $C_4$ are close
- $C_1$ and $C_3$ are far
- we also need: **target function varies smoothly**

# SSL with Graphs: What is behind it?




**Knowing the manifold helps**

- $C_1$ and $C_4$ are close
- $C_1$ and $C_3$ are far
- we also need: **target function varies smoothly**
- altogether: **closeness on manifold $\rightarrow$ similarity in labels**

# SSL with Graphs: What is behind it?

What does it mean to **know** $\mathcal{M}$?

# SSL with Graphs: What is behind it?

What does it mean to **know** $\mathcal{M}$?

**Different degrees of knowing $\mathcal{M}$**

- set membership oracle: $\mathbf{x} \overset{?}{\in} \mathcal{M}$
- approximate oracle
- knowing the harmonic functions on $\mathcal{M}$
- knowing the Laplacian $\mathcal{L}_{\mathcal{M}}$
- knowing eigenvalues and eigen*functions*
- topological invariants, e.g., dimension
- metric information: geodesic distance

# OnlineSSL($\mathcal{G}$)

when we can't access future $\mathbf{x}$

...and we want the results in real time

# 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 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 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 $\mathbb{R}^d$

↪ what if we want to learn faces while watching!

↪ also useful for computational reasons

# 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 $\mathbf{x}_t$

# 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**

```
while new unlabeled example x_t comes do
    Add x_t to graph G(W)
```

# Online SSL with Graphs

Online HFS: Straightforward solution

**while** new unlabeled example $\mathbf{x}_t$ comes **do**
    Add $\mathbf{x}_t$ to graph $G(\mathbf{W})$
    Update $\mathbf{L}_t$

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

# 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 $\widehat{y}_t =$

# Online SSL with Graphs

**Online HFS: Straightforward solution**

**while** new unlabeled example $\mathbf{x}_t$ comes **do**
    Add $\mathbf{x}_t$ to graph $G(\mathbf{W})$
    Update $\mathbf{L}_t$
    Infer labels $\mathbf{f}_u = (\mathbf{L}_{uu} + \gamma_g \mathbf{I})^{-1} (\mathbf{W}_{ul}\mathbf{f}_l)$
    Predict $\widehat{y}_t = \operatorname{sgn}(\mathbf{f}_u(t))$
**end while**

# 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 $\widehat{y}_t = \operatorname{sgn}(\mathbf{f}_u(t))$
6: **end while**

What is wrong with this solution?

# Online SSL with Graphs

**Online HFS: Straightforward solution**

**while** new unlabeled example $\mathbf{x}_t$ comes **do**
  Add $\mathbf{x}_t$ to graph $G(\mathbf{W})$
  Update $\mathbf{L}_t$
  Infer labels $\mathbf{f}_u = (\mathbf{L}_{uu} + \gamma_g \mathbf{I})^{-1} (\mathbf{W}_{ul}\mathbf{f}_l)$
  Predict $\widehat{y}_t = \operatorname{sgn}(\mathbf{f}_u(t))$
**end while**

What is wrong with this solution?

The cost and memory of the operations.

# Online SSL with Graphs

**Online HFS: Straightforward solution**

- **while** new unlabeled example $\mathbf{x}_t$ comes **do**
  - Add $\mathbf{x}_t$ to graph $G(\mathbf{W})$
  - Update $\mathbf{L}_t$
  - Infer labels $\mathbf{f}_u = (\mathbf{L}_{uu} + \gamma_g \mathbf{I})^{-1} (\mathbf{W}_{ul}\mathbf{f}_l)$
  - Predict $\widehat{y}_t = \operatorname{sgn}(\mathbf{f}_u(t))$
- **end while**

What is wrong with this solution?

The cost and memory of the operations.

Do not scale!

What can we do?

# Online SSL with Graphs

Let's keep only $k$ vertices!

# Online SSL with Graphs

Let's keep only $k$ vertices!

Limit memory to $k$ **centroids** with $\underbrace{\widetilde{\mathbf{W}}^{\mathrm{q}} \text{ weights.}}_{\text{between each other } (k^2 \text{ edges})}$

# Online SSL with Graphs

Let's keep only $k$ vertices!

Limit memory to $k$ **centroids** with $\widetilde{\mathbf{W}}^{\mathrm{q}}$ weights. → centroids will need to represent bunch of points

Each centroid represents *several* others.

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

Each centroid represents *several* others.

Diagonal $\mathbf{V} \equiv$ **multiplicity**. $\rightarrow$ how many points in a centroid

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

Each centroid represents *several* others.

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

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

# Online SSL with Graphs

**Let's keep only $k$ vertices!**

Limit memory to $k$ **centroids** with $\widetilde{\mathbf{W}}^{\mathrm{q}}$ 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. We can (thus do harmonic solve) replace with centroids.

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

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

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

Proof?

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

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

Proof? Using electric circuits.

# Online SSL with Graphs

Let's keep only $k$ vertices!

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

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

Proof? Using electric circuits.

Why do we keep the multiplicities?

# Online SSL with Graphs

quantization

**Online HFS with Graph Quantization**

**Input**

$k$ number of representative nodes

# Online SSL with Graphs

Online HFS with Graph Quantization

**Input**
    $k$ number of representative nodes
**Initialization**
    $\mathbf{V}$ matrix of multiplicities with 1 on diagonal

# Online SSL with Graphs

Online HFS with Graph Quantization

**Input**
    $k$ number of representative nodes
**Initialization**
    $\mathbf{V}$ matrix of multiplicities with 1 on diagonal
**while** new unlabeled example $\mathbf{x}_t$ comes **do**
    Add $\mathbf{x}_t$ to graph $G$
    **if** $\#$ nodes $> k$ **then**
        quantize $G$
    **end if**

# Online SSL with Graphs

**Online HFS with Graph Quantization**

**Input**
  $k$ number of representative nodes
**Initialization**
  $\mathbf{V}$ matrix of multiplicities with 1 on diagonal
**while** new unlabeled example $\mathbf{x}_t$ comes **do**
  Add $\mathbf{x}_t$ to graph $G$
  **if** $\#$ nodes $> k$ **then**
    quantize $G$
  **end if**
  Update $\mathbf{L}_t$ of $G(\mathbf{VWV})$

# Online SSL with Graphs

we don't want all (we want each label data to be in the graph)

**Online HFS with Graph Quantization**

**Input**
  $k$ number of representative nodes
**Initialization**
  $\mathbf{V}$ matrix of multiplicities with 1 on diagonal
**while** new unlabeled example $\mathbf{x}_t$ comes **do**
  Add $\mathbf{x}_t$ to graph $G$
  **if** $\#$ nodes $> k$ **then**
    quantize $G$
  **end if**
  Update $\mathbf{L}_t$ of $G(\mathbf{VWV})$
  Infer labels
  Predict $\widehat{y}_t = \operatorname{sgn}(\mathbf{f}_u(t))$
**end while**

# Online SSL with Graphs: Graph Quantization

An idea: incremental $k$-centers

# Online SSL with Graphs: Graph Quantization

An idea: incremental $k$-centers

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

# 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

# Online SSL with Graphs: Graph Quantization

An idea: incremental $k$-centers

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

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

- Distance $\mathbf{c}_i, \mathbf{c}_j \in C_t$ is at least $\geq R$ → try to keep some variance

# Online SSL with Graphs: Graph Quantization

An idea: incremental $k$-centers

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

Keeps up to $k$ centers $C_t = \{\mathbf{c}_1, \mathbf{c}_2, \ldots\}$ 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 $R$. → every time a new point arrives, we want to have a center in radius R. If not, create a cluster

# 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 $R$.
- $|C_t| \leq k$

# Online SSL with Graphs: Graph Quantization

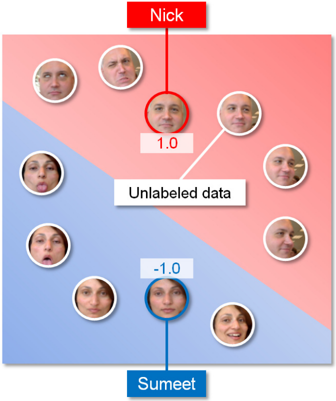
An idea: incremental $k$-centers

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

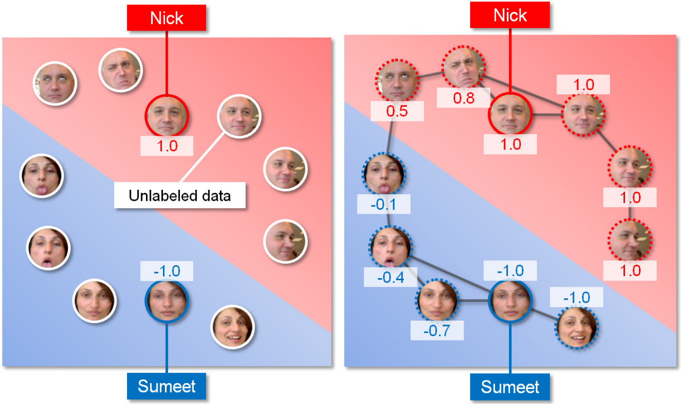
Keeps up to $k$ centers $C_t = \{\mathbf{c}_1, \mathbf{c}_2, \dots\}$ with

- 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

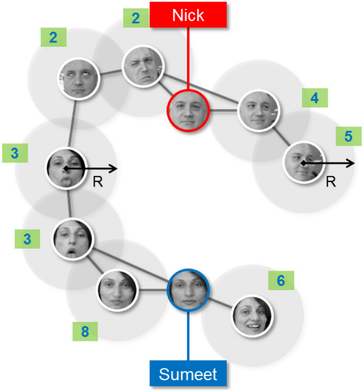
# Online SSL with Graphs: Graph Quantization



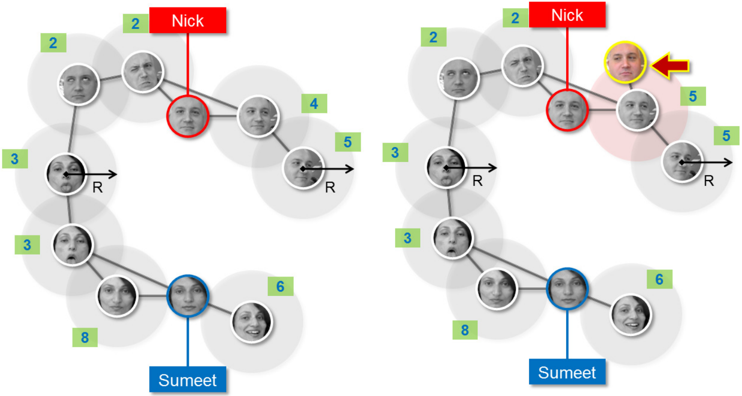
# Online SSL with Graphs: Graph Quantization



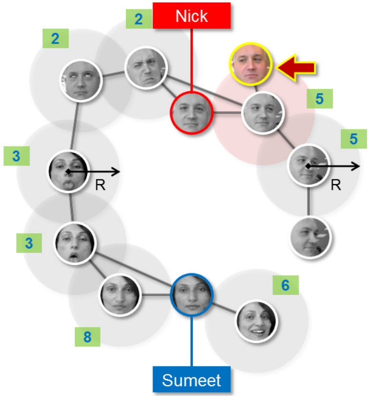
# Online SSL with Graphs: Graph Quantization



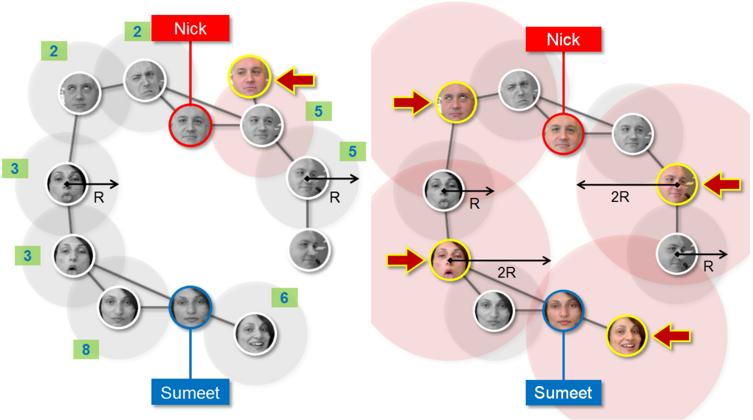
# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization



# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

1: an unlabeled $\mathbf{x}_t$, a set of centroids $C_{t-1}$, multiplicities $\mathbf{v}_{t-1}$

# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

an unlabeled $\mathbf{x}_t$, a set of centroids $C_{t-1}$, multiplicities $\mathbf{v}_{t-1}$

**if** $(|C_{t-1}| = k + 1)$ **then**

  $R \leftarrow mR$

# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

an unlabeled $\mathbf{x}_t$, a set of centroids $C_{t-1}$, multiplicities $\mathbf{v}_{t-1}$

**if** $(|C_{t-1}| = k+1)$ **then**

  $R \leftarrow mR$

  greedily repartition $C_{t-1}$ into $C_t$ such that:

    no two vertices in $C_t$ are closer than $R$

    for any $\mathbf{c}_i \in C_{t-1}$ exists $\mathbf{c}_j \in C_t$ such that $d(\mathbf{c}_i, \mathbf{c}_j) < R$

# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

```
an unlabeled x_t, a set of centroids C_{t-1}, multiplicities v_{t-1}
if (|C_{t-1}| = k + 1) then
    R ← mR
    greedily repartition C_{t-1} into C_t such that:
        no two vertices in C_t are closer than R
        for any c_i ∈ C_{t-1} exists c_j ∈ C_t such that d(c_i, c_j) < R
    update v_t to reflect the new partitioning
else
    C_t ← C_{t-1}
    v_t ← v_{t-1}
```

# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

```
an unlabeled x_t, a set of centroids C_{t-1}, multiplicities v_{t-1}
if (|C_{t-1}| = k + 1) then
    R ← mR
    greedily repartition C_{t-1} into C_t such that:
        no two vertices in C_t are closer than R
        for any c_i ∈ C_{t-1} exists c_j ∈ C_t such that d(c_i, c_j) < R
    update v_t to reflect the new partitioning
else
    C_t ← C_{t-1}
    v_t ← v_{t-1}
end if
if x_t is closer than R to any c_i ∈ C_t then
    v_t(i) ← v_t(i) + 1
else
```

# Online SSL with Graphs: Graph Quantization

**Online $k$-centers**

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

# Online SSL with Graphs: Graph Quantization

Doubling algorithm [Cha+97]

# Online SSL with Graphs: Graph Quantization

Doubling algorithm [Cha+97]

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

# 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 $\mathbf{c}$?

# 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 $\mathbf{c}$?

$R$

# 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 $\mathbf{c}$?

$$R + \frac{R}{m}$$

# 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 $\mathbf{c}$?

$$R + \frac{R}{m} + \frac{R}{m^2}$$

# 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 $\mathbf{c}$?

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

# 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 $\mathbf{c}$?

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

# 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 $\mathbf{c}$?

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

Guarantees

# 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 $\mathbf{c}$?

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

Guarantees: 8-approximation algorithm.

# Online SSL with Graphs: Graph Quantization

If kmeans (k) performs 0.634
then k-m online $\leq 8 \times 0.634$

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 $\mathbf{c}$?**

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

Guarantees: 8-approximation algorithm.

**Why not incremental $k$-means?**

We can more stability with the potential variance

# Online SSL with Graphs

**Video examples**

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

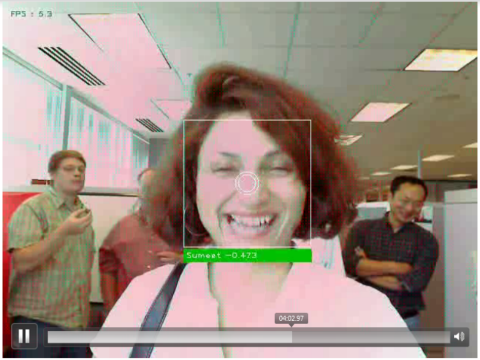
http://www.bkveton.com/videos/Ad.mp4

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

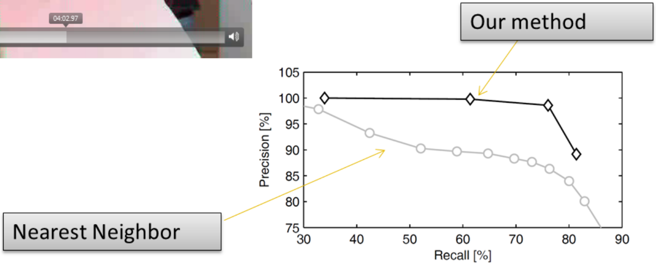
http://researchers.lille.inria.fr/~valko/hp/serve.php?what=publications/kveton2009nipsdemo.officespace.mov

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

# SSL with Graphs: Some experimental results

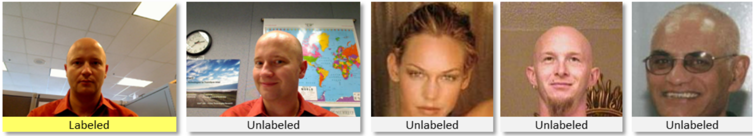


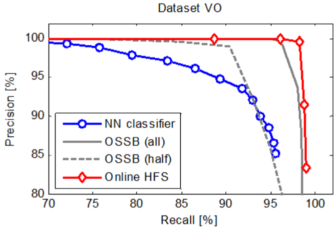

- 8 people classification
- Making funny faces
- 4 faces/person are labeled

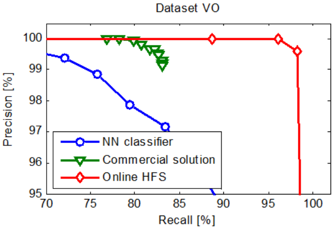


# SSL with Graphs: Some experimental results

- 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)

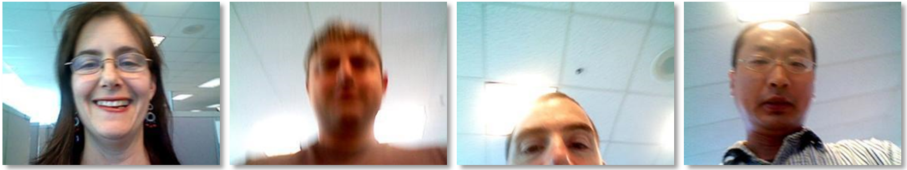
Online HFS yields better results than a commercial solution at 20% of the computational cost

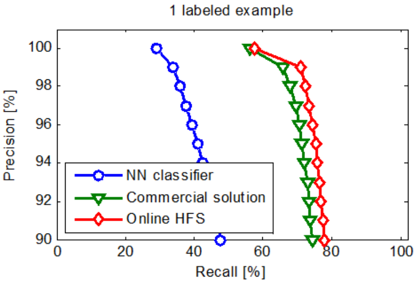
# SSL with Graphs: Some experimental results

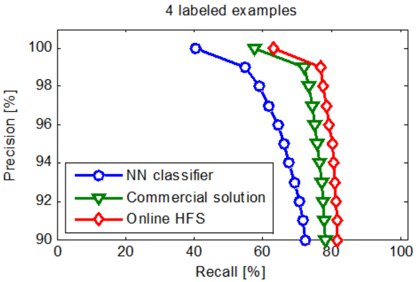
- **Logging in** with faces instead of password
- Able to **learn** and improve


# SSL with Graphs: Some experimental results

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





# Online SSL with Graphs: Analysis

Want to bound $\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-y_t)^2$

What can we guarantee?

# Online SSL with Graphs: Analysis

Want to bound $\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-y_t)^2$

What can we guarantee?

Three sources of error

## Online SSL with Graphs: Analysis

Want to bound $\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-y_t)^2$

What can we guarantee?

Three sources of error

- generalization error — if all data: $(\ell_t^\star - y_t)^2$ offline vs world
- online error — data only incrementally: $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$ online vs offline
- quantization error — memory limitation: $(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2$ quantized vs online

# Online SSL with Graphs: Analysis

Want to bound $\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-y_t)^2$

What can we guarantee?

Three sources of error

- generalization error — if all data: $(\ell_t^\star - y_t)^2$
- online error — data only incrementally: $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$
- quantization error — memory limitation: $(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2$

All together:

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-y_t)^2 \leq \frac{9}{2N}\sum_{t=1}^{N}(\ell_t^\star-y_t)^2+\frac{9}{2N}\sum_{t=1}^{N}(\ell_t^{\mathrm{o}}[t]-\ell_t^\star)^2+\frac{9}{2N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t])^2$$

# Online SSL with Graphs: Analysis

**Bounding transduction error** $(\ell_t^\star - y_t)^2$

# Online SSL with Graphs: Analysis

**Bounding** transduction error $(\ell_t^\star - y_t)^2$

If all labeled examples $l$ are i.i.d., $c_l = 1$ and $c_l \gg c_u$, then

$$R(\boldsymbol{\ell}^\star) \leq \widehat{R}(\boldsymbol{\ell}^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$

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

holds with the probability of $1 - \delta$, where

$$R(\boldsymbol{\ell}^\star) = \frac{1}{N}\sum_t (\ell_t^\star - y_t)^2 \quad \text{and} \quad \widehat{R}(\boldsymbol{\ell}^\star) = \frac{1}{n_l}\sum_{t \in l}(\ell_t^\star - y_t)^2$$

# Online SSL with Graphs: Analysis

**Bounding** transduction error $(\ell_t^\star - y_t)^2$

If all labeled examples $l$ are i.i.d., $c_l = 1$ and $c_l \gg c_u$, then

$$R(\boldsymbol{\ell}^\star) \;\leq\; \widehat{R}(\boldsymbol{\ell}^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$

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

holds with the probability of $1 - \delta$, where

$$R(\boldsymbol{\ell}^\star) = \frac{1}{N}\sum_t (\ell_t^\star - y_t)^2 \quad \text{and} \quad \widehat{R}(\boldsymbol{\ell}^\star) = \frac{1}{n_l}\sum_{t \in l}(\ell_t^\star - y_t)^2$$

How should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** transduction error $(\ell_t^\star - y_t)^2$

If all labeled examples $l$ are i.i.d., $c_l = 1$ and $c_l \gg c_u$, then

$$R(\boldsymbol{\ell}^\star) \leq \widehat{R}(\boldsymbol{\ell}^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$

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

holds with the probability of $1 - \delta$, where

$$R(\boldsymbol{\ell}^\star) = \frac{1}{N}\sum_t (\ell_t^\star - y_t)^2 \quad \text{and} \quad \widehat{R}(\boldsymbol{\ell}^\star) = \frac{1}{n_l}\sum_{t \in l}(\ell_t^\star - y_t)^2$$

How should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

### Bounding transduction error $(\ell_t^\star - y_t)^2$

If all labeled examples $l$ are i.i.d., $c_l = 1$ and $c_l \gg c_u$, then

$$R(\boldsymbol{\ell}^\star) \leq \widehat{R}(\boldsymbol{\ell}^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$

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

holds with the probability of $1 - \delta$, where

$$R(\boldsymbol{\ell}^\star) = \frac{1}{N}\sum_t (\ell_t^\star - y_t)^2 \quad \text{and} \quad \widehat{R}(\boldsymbol{\ell}^\star) = \frac{1}{n_l}\sum_{t \in l} (\ell_t^\star - y_t)^2$$

How should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** **transduction error** $(\ell_t^\star - y_t)^2$

If all labeled examples $l$ are i.i.d., $c_l = 1$ and $c_l \gg c_u$, then

$$R(\boldsymbol{\ell}^\star) \leq \widehat{R}(\boldsymbol{\ell}^\star) + \underbrace{\beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta + 4)}_{\text{transductive error } \Delta_T(\beta, n_l, \delta)}$$

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

holds with the probability of $1 - \delta$, where

$$R(\boldsymbol{\ell}^\star) = \frac{1}{N}\sum_t (\ell_t^\star - y_t)^2 \quad \text{and} \quad \widehat{R}(\boldsymbol{\ell}^\star) = \frac{1}{n_l}\sum_{t \in l}(\ell_t^\star - y_t)^2$$

How should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\mathrm{o}}[t] - \ell_t^{\star})^2$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell_t^{\text{o}}[t] - \ell_t^{\star})^2$

i.e. $\gamma_g$ is big

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\text{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell_t^{\mathrm{o}}[t] - \ell_t^{\star})^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq$$

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\text{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\text{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})}$$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1}$$

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\text{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\text{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$$

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2 \leq \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_\infty^2$$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell^{\mathrm{o}}_t[t] - \ell^\star_t)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \le \|A\mathbf{v}\|_2 \le \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \le \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \le \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell^{\mathrm{o}}_t[t] - \ell^\star_t)^2 \le \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_\infty^2 \le \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_2^2$$

# Online SSL with Graphs: Analysis

**Bounding online error** $(\ell_t^{\text{o}}[t] - \ell_t^{\star})^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\text{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n \times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell_t^{\text{o}}[t] - \ell_t^{\star})^2 \leq \|\boldsymbol{\ell}^{\text{o}}[t] - \boldsymbol{\ell}^{\star}\|_\infty^2 \leq \|\boldsymbol{\ell}^{\text{o}}[t] - \boldsymbol{\ell}^{\star}\|_2^2 \leq \left(\frac{2\sqrt{n_l}}{\gamma_g + 1}\right)^2$$

# Online SSL with Graphs: Analysis

**Bounding** **online error** $(\ell^{\mathrm{o}}_t[t] - \ell^\star_t)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\boldsymbol{\ell} = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell^{\mathrm{o}}_t[t] - \ell^\star_t)^2 \leq \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_\infty^2 \leq \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_2^2 \leq \left(\frac{2\sqrt{n_l}}{\gamma_g + 1}\right)^2$$

Again, how should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\text{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\text{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\ell = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \le \|A\mathbf{v}\|_2 \le \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \le \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \le \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell_t^{\text{o}}[t] - \ell_t^\star)^2 \le \|\boldsymbol{\ell}^{\text{o}}[t] - \boldsymbol{\ell}^\star\|_\infty^2 \le \|\boldsymbol{\ell}^{\text{o}}[t] - \boldsymbol{\ell}^\star\|_2^2 \le \left(\frac{2\sqrt{n_l}}{\gamma_g + 1}\right)^2$$

Again, how should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** online error $(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2$

Idea: If $\mathbf{L}$ and $\mathbf{L}^{\mathrm{o}}$ are regularized, then HFSs get closer together.

since they get closer to zero

Recall $\ell = (\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})^{-1}\mathbf{y}$, where $\mathbf{Q} = \mathbf{L} + \gamma_g \mathbf{I}$

and also $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\lambda_m(A)\|\mathbf{v}\|_2 \leq \|A\mathbf{v}\|_2 \leq \lambda_M(A)\|\mathbf{v}\|_2$

$$\|\boldsymbol{\ell}\|_2 \leq \frac{\|\mathbf{y}\|_2}{\lambda_m(\mathbf{C}^{-1}\mathbf{Q} + \mathbf{I})} = \frac{\|\mathbf{y}\|_2}{\frac{\lambda_m(\mathbf{Q})}{\lambda_M(\mathbf{C})} + 1} \leq \frac{\sqrt{n_l}}{\gamma_g + 1}$$

Difference between offline and online solutions:

$$(\ell_t^{\mathrm{o}}[t] - \ell_t^\star)^2 \leq \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_\infty^2 \leq \|\boldsymbol{\ell}^{\mathrm{o}}[t] - \boldsymbol{\ell}^\star\|_2^2 \leq \left(\frac{2\sqrt{n_l}}{\gamma_g + 1}\right)^2$$

Again, how should we set $\gamma_g$?

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

How are the quantized and full solution different?

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

# Online SSL with Graphs: Analysis

with quantization, we replace a bunch of points with repetitions of centroids: i.e. the corresponding laplacian in blocks

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

How are the quantized and full solution different?

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

In **Q**!

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

How are the quantized and full solution different?

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

In **Q**! $\mathbf{Q}^{\mathrm{o}}$ (online) vs. $\mathbf{Q}^{\mathrm{q}}$ (quantized)

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

How are the quantized and full solution different?

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

In **Q**! $\mathbf{Q}^{\mathrm{o}}$ (online) vs. $\mathbf{Q}^{\mathrm{q}}$ (quantized)

We have: $\boldsymbol{\ell}^{\mathrm{o}} = (\mathbf{C}^{-1}\mathbf{Q}^{\mathrm{o}} + \mathbf{I})^{-1}\mathbf{y}$ vs. $\boldsymbol{\ell}^{\mathrm{q}} = (\mathbf{C}^{-1}\mathbf{Q}^{\mathrm{q}} + \mathbf{I})^{-1}\mathbf{y}$

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

How are the quantized and full solution different?

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

In **Q**! $\mathbf{Q}^{\mathrm{o}}$ (online) vs. $\mathbf{Q}^{\mathrm{q}}$ (quantized)

We have: $\boldsymbol{\ell}^{\mathrm{o}} = (\mathbf{C}^{-1}\mathbf{Q}^{\mathrm{o}} + \mathbf{I})^{-1}\mathbf{y}$ vs. $\boldsymbol{\ell}^{\mathrm{q}} = (\mathbf{C}^{-1}\mathbf{Q}^{\mathrm{q}} + \mathbf{I})^{-1}\mathbf{y}$

With linear algebra we get

$$\|\boldsymbol{\ell}^{\mathrm{q}} - \boldsymbol{\ell}^{\mathrm{o}}\|_2 \leq \frac{\sqrt{n_l}}{c_u\gamma_g^2}\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F$$

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

The quantization error depends on $\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F = \|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$.

When can we keep $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$ under control?

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

The quantization error depends on $\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F = \|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$.

When can we keep $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$ under control?

Charikar guarantees **distortion** error of at most $Rm/(m-1)$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

The quantization error depends on $\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F = \|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$.

When can we keep $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$ under control?

Charikar guarantees **distortion** error of at most $Rm/(m-1)$

For what kind of data $\{\mathbf{x}_i\}_{i=1,\dots,n}$ is the distortion small?

# Online SSL with Graphs: Analysis

distortion: How far away are old points in history from centroids?

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

The quantization error depends on $\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F = \|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$.

When can we keep $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$ under control?

Charikar guarantees **distortion** error of at most $Rm/(m-1)$

For what kind of data $\{\mathbf{x}_i\}_{i=1,\ldots,n}$ is the distortion small?

Assume manifold $\mathcal{M}$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

The quantization error depends on $\|\mathbf{Q}^{\mathrm{q}} - \mathbf{Q}^{\mathrm{o}}\|_F = \|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$.

When can we keep $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F$ under control?

Charikar guarantees **distortion** error of at most $Rm/(m-1)$

For what kind of data $\{\mathbf{x}_i\}_{i=1,\ldots,n}$ is the distortion small?

Assume manifold $\mathcal{M}$

- all $\{\mathbf{x}_i\}_{i\geq 1}$ lie on a smooth $d$-dimensional compact $\mathcal{M}$
- with boundary of bounded geometry Def. 11 of Hein [HAL07]
  - has finite volume $V$
  - has finite surface area $A$
  - should not intersect itself
  - should not fold back onto itself

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

Bounding $\left\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\right\|_F$ when $\mathbf{x}_i \in \mathcal{M}$

Consider $k$-sphere packing* of radius $r$ with centers contained in $\mathcal{M}$. *only the centers are packed, not necessarily the entire ball

If $k$ is large $\rightarrow r <$ **injectivity radius** of $\mathcal{M}$ [HAL07]

# Online SSL with Graphs: Analysis

## **Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

Bounding $\left\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\right\|_F$ when $\mathbf{x}_i \in \mathcal{M}$

Consider $k$-sphere packing$^\star$ of radius $r$ with centers contained in $\mathcal{M}$. $^\star$only the centers are packed, not necessarily the entire ball

If $k$ is large $\rightarrow r <$ **injectivity radius** of $\mathcal{M}$ [HAL07] and $r < 1$:

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

Bounding $\left\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\right\|_F$ when $\mathbf{x}_i \in \mathcal{M}$

Consider $k$-sphere packing$^\star$ of radius $r$ with centers contained in $\mathcal{M}$. $^\star$only the centers are packed, not necessarily the entire ball

If $k$ is large $\to r <$ **injectivity radius** of $\mathcal{M}$ [HAL07] and $r < 1$:

$$r < \left(\frac{V + Ac_{\mathcal{M}}}{kc_d}\right)^{1/d} = \mathcal{O}\left(\frac{1}{k^{1/d}}\right)$$

$r$-packing is a $2r$-covering:

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

Bounding $\left\|\mathbf{L}^{\mathrm{q}}-\mathbf{L}^{\mathrm{o}}\right\|_F$ when $\mathbf{x}_i \in \mathcal{M}$

Consider $k$-sphere packing* of radius $r$ with centers contained in $\mathcal{M}$. *only the centers are packed, not necessarily the entire ball

If $k$ is large $\rightarrow r <$ **injectivity radius** of $\mathcal{M}$ [HAL07] and $r < 1$:

$$r < \left(\frac{V + Ac_{\mathcal{M}}}{kc_d}\right)^{1/d} = \mathcal{O}\left(\frac{1}{k^{1/d}}\right)$$

$r$-packing is a $2r$-covering:

$$\max_{i=1,\ldots,N} \|\mathbf{x}_i - \mathbf{c}\|_2 \leq R\frac{m}{m-1} \leq 2\mathcal{O}\left(k^{-1/d}\right) = \mathcal{O}\left(k^{-1/d}\right)$$

But what about $\left\|\mathbf{L}^{\mathrm{q}}-\mathbf{L}^{\mathrm{o}}\right\|_F$?

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

If similarity is $M$-Lipschitz

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

If similarity is $M$-Lipschitz, $\mathbf{L}$ is normalized

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

If similarity is $M$-Lipschitz, $\mathbf{L}$ is normalized, $\forall i, j \ \frac{\sqrt{\mathbf{D}_{ii}^{\mathrm{o}} \mathbf{D}_{jj}^{\mathrm{o}}}}{N} > c_{min}$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

If similarity is $M$-Lipschitz, $\mathbf{L}$ is normalized, $\forall i, j$ $\frac{\sqrt{\mathbf{D}^{\mathrm{o}}_{ii}\mathbf{D}^{\mathrm{o}}_{jj}}}{N} > c_{min}$

$$\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(MR^2/c_{min}) = \mathcal{O}(k^{-2/d}).$$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

If similarity is $M$-Lipschitz, $\mathbf{L}$ is normalized, $\forall i, j \ \frac{\sqrt{\mathbf{D}_{ii}^{\mathrm{o}}\mathbf{D}_{jj}^{\mathrm{o}}}}{N} > c_{min}$

$$\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(MR^2/c_{min}) = \mathcal{O}(k^{-2/d}).$$

Are the assumptions reasonable?

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d})$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2 \leq \frac{n_l}{c_u^2\gamma_g^4}\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2$$

# Online SSL with Graphs: Analysis

**Bounding** **quantization error** $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2 \leq \frac{n_l}{c_u^2\gamma_g^4}\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \frac{n_l}{c_u^2\gamma_g^4}$$

# Online SSL with Graphs: Analysis

**Bounding quantization error** $\left(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2 \leq \frac{n_l}{c_u^2\gamma_g^4}\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \frac{n_l}{c_u^2\gamma_g^4}$$

With properly setting $\gamma_g$, e.g., $\gamma_g = \Omega(n_l^{3/8})$, we can have

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2 \leq \frac{n_l}{c_u^2\gamma_g^4}\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \frac{n_l}{c_u^2\gamma_g^4}$$

With properly setting $\gamma_g$, e.g., $\gamma_g = \Omega(n_l^{3/8})$, we can have

$$\frac{1}{N}\sum_{t=1}^{N}\left(\ell_t^{\mathrm{q}}[t] - y_t\right)^2 = \mathcal{O}\left(n_l^{-1/2}\right).$$

# Online SSL with Graphs: Analysis

**Bounding** quantization error $\left(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t]\right)^2$

We showed $\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \mathcal{O}(k^{-2/d}) = \mathcal{O}(1)$.

$$\frac{1}{N}\sum_{t=1}^{N}(\ell_t^{\mathrm{q}}[t] - \ell_t^{\mathrm{o}}[t])^2 \leq \frac{n_l}{c_u^2 \gamma_g^4}\|\mathbf{L}^{\mathrm{q}} - \mathbf{L}^{\mathrm{o}}\|_F^2 \leq \frac{n_l}{c_u^2 \gamma_g^4}$$

With properly setting $\gamma_g$, e.g., $\gamma_g = \Omega(n_l^{3/8})$, we can have

$$\frac{1}{N}\sum_{t=1}^{N}\left(\ell_t^{\mathrm{q}}[t] - y_t\right)^2 = \mathcal{O}\left(n_l^{-1/2}\right).$$

do not do worse than SSL offline

What does that mean?

*Daniele Calandriello*

dcalandriello@google.com

ENS Paris-Saclay, MVA 2022/2023

https://sites.google.com/view/daniele-calandriello/