SOME NEW DIRECTIONS IN GRAPH-BASED SEMI-SUPERVISED LEARNING

XIAOJIN ZHU, ANDREW B. GOLDBERG, TUSHAR KHOT

University of Wisconsin-Madison

Our position

- Current graph-based Semi-supervised learning (SSL) methods have three limitations:
  - data is restricted to live on a single manifold
  - learning must happen in batch mode
  - the target label is assumed smooth on the manifold
- We propose three new directions:
  - multiple manifolds learning
  - Online SSL
  - Compressive sensing for SSL

Background and notation

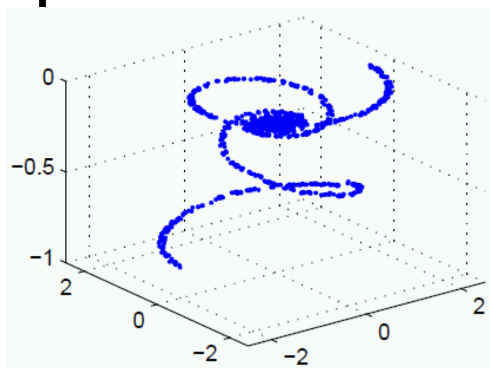
- Input: n labeled points {(xi,yi)}, m unlabeled {xi}
- Goal: learn f: X→Y
- Graph on n+m points, W_{ij} edge weight
- Assumption: large edge weight → similar label
- Weight matrix W, degree matrix D, Laplacian matrix L=D-W
- Optimization:

minimize the energy f'Lf,

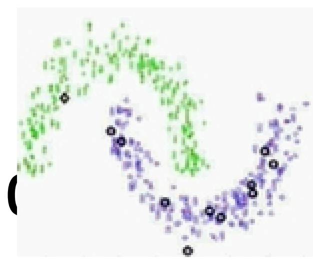
subject to given labels fi≈yi

Limitation 1: no intersecting manifolds

- Existing graph-based SSL works well on a single man



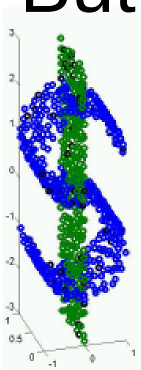
- Or on multiple well-separated manifol

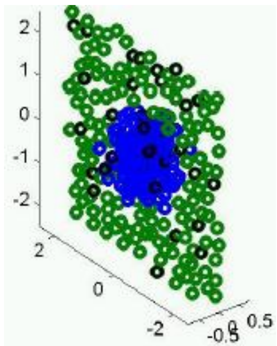


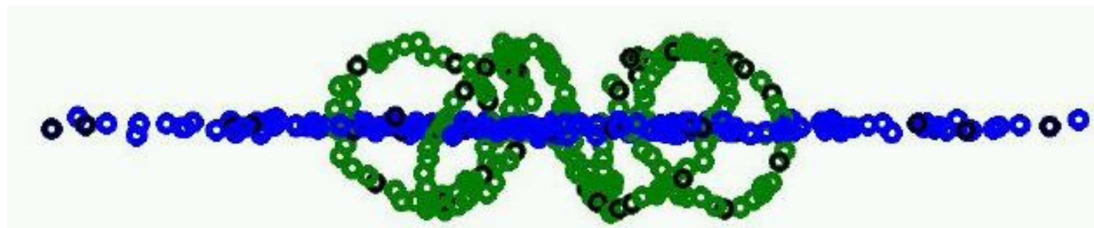
- Edge weight depends on simple (Euclidean) distance: the closer, the larger
  - RBF weight $w_{ij} = \exp\left(-\lambda d(x_i, x_j)^2\right)$
  - K nearest neighbor (1 if close, 0 otherwise)

Limitation 1: no intersecting manifolds

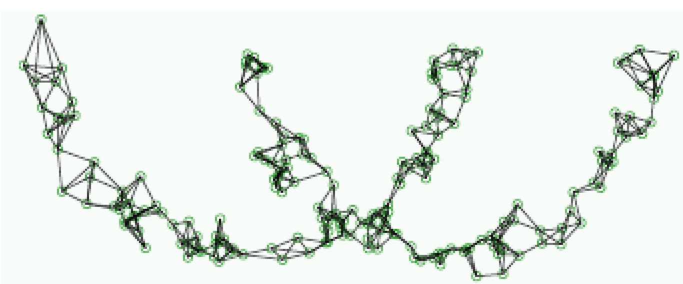
- But cannot handle intersecting manifolds:



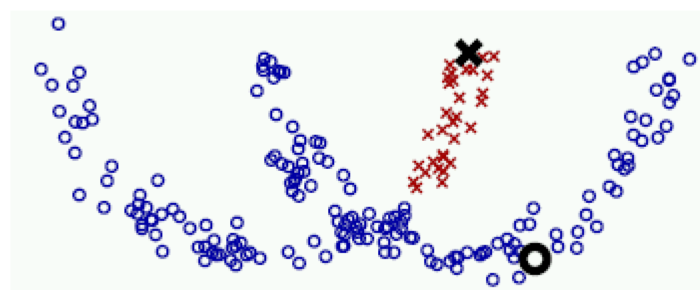




- Euclidean-distance-based weights will mix up manifolds



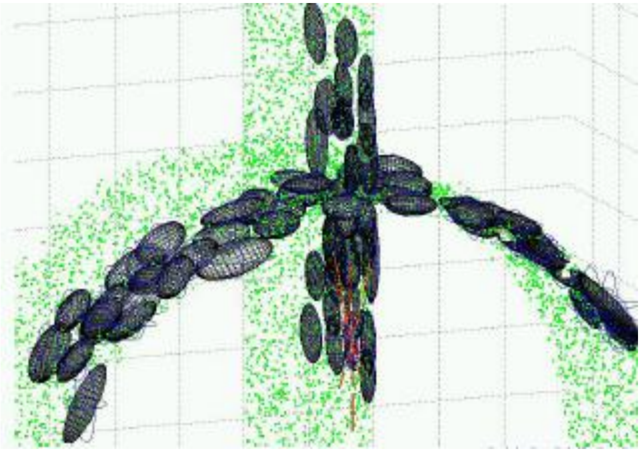
(a) 4-NN graph



(b) Harmonic function predictions

Solution: local covariance

- The sample covariance matrix (ellipsoid) captures local geometry $\Sigma_x = \frac{1}{m-1}\sum_j (x_j - \mu_x)(x_j - \mu_x)^\top$
- Similar nearby ellipsoids → large edge weight



- But how to measure covariance similarity?

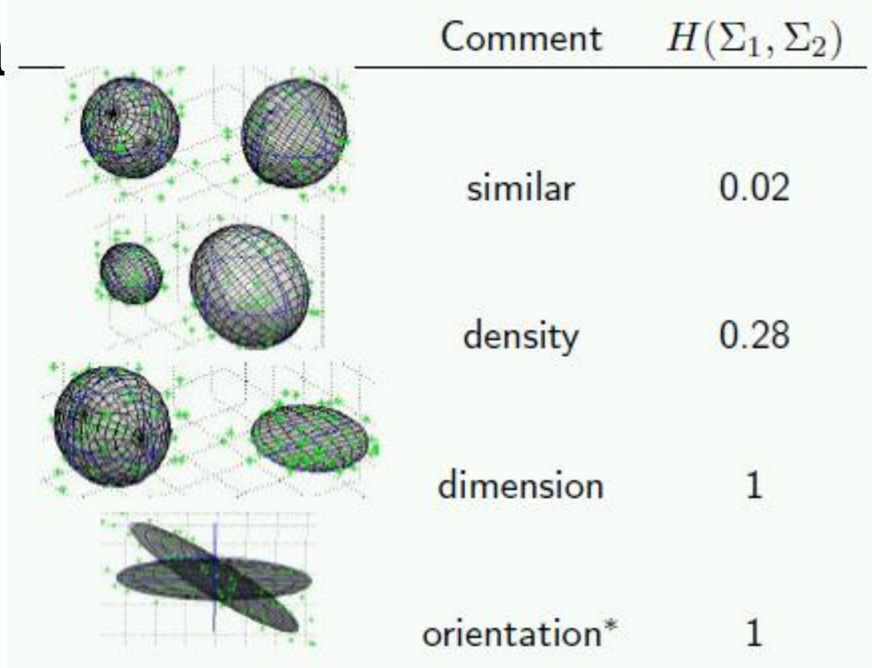
A distance on covariance matrices

- Hellinger distance $H^2(p,q) = \frac{1}{2}\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 dx$
- Symmetric, value in [0,1]
- Let p be the normal distribution at mean 0 with covariance Σ_1, similarly for q
- Define the Hellinger distance between two covariance matrices as

$$H(\Sigma_1, \Sigma_2) \equiv H(p,q) = \sqrt{1 - 2^{\frac{d}{2}} \frac{|\Sigma_1|^{\frac{1}{4}} |\Sigma_2|^{\frac{1}{4}}}{|\Sigma_1 + \Sigma_2|^{\frac{1}{2}}}}$$

Property of Hellinger distance

- Large value if the two covariance matrices are similar; close to 0, if they differ in density, dimensionality or orientation
- Ideal for tracing a manifold in a mixture of multiple ma



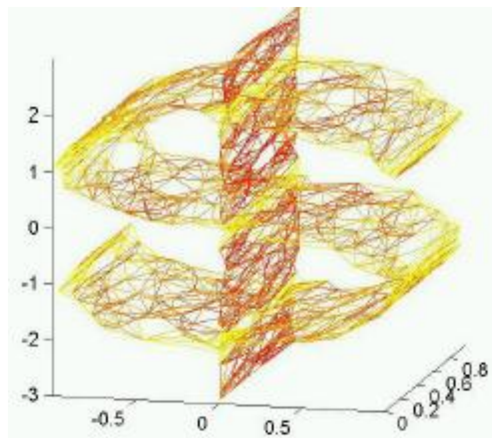
Comment	$H(\Sigma_1, \Sigma_2)$
similar	0.02
density	0.28
dimension	1
orientation*	1

Hellinger distance for multi-manifold

- Similar covariance → large weig

$$w_{ij} = e^{-\frac{H^2(\Sigma_{x_i}, \Sigma_{x_j})}{2\sigma^2}}$$

- Example: red=large weight, yellow=small weight



- Use this graph in manifold regularization – it will separate the manifolds.

Limitation 2: need all data at once

- In many applications, data stream in. Cannot store them all. Want:
  - Online processing and then discard each incoming item
  - Learn even when the item is unlabeled (different from standard online learning)
  - Tolerate adversarial concept drifts (changes in X→Y)
  - Theoretic guarantee
  - Uses only finite memory budget

Online SSL setting

1. At time t, adversary picks (x_t, y_t) not necessarily iid, shows x_t
2. Learner uses current predictor f_t to predict $f_t(x_t)$
3. With a small probability, adversary reveals y_t, otherwise it abstains (unlabeled)
4. Learner updates $f_t \rightarrow f_{t+1}$, based on x_t and y_t (if given)
5. Repeat for $t \leftarrow t+1$

Solution: online convex programming

- Batch SSL minimizes a risk functional J(f) on all data
- If J can be decomposed into a sum of instantaneous risks $J_t(f)$ on individual data item

$$J(f) = \frac{1}{T}\sum_{t=1}^{T} J_t(f)$$

$$f_{t+1} = f_t - \eta_t \left.\frac{\partial J_t(f)}{\partial f}\right|_{f_t}$$

- Then one can do gradient descent on $J_t(f)$ at each step
- Even though each $J_t(f)$ is different, one can show this gradient descent procedure optimizes something sensible: in particular,

No-regret guarantee

- In online learning with concept drift, accuracy is not a good measure, because adversary can change the true labels arbitrarily often
- Instead, measure the difference to the best batch hypothesis f* (which will also be bad if concept drifts too often), known as the regret

$$\text{regret} \equiv \frac{1}{T}\sum_{t=1}^{T} J_t(f_t) - J(f^*)$$

- [Zinkevich03] the gradient descent procedure has zero regret asymptotically.

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} J_t(f_t) - J(f^*) \leq 0$$

Online graph-based SSL

- This can be applied to graph-based SSL
- The instantaneous risk involves a subgraph from x_t to all previous points
- Limited memory version: only keep a fixed length buffer, instead of all previous points
- Open questions: better ways to define the instantaneous risk, such that the manifold structure is summarized using finite memory. (on-going work)

Limitation 3: f has to be smooth

- Eigen value/vectors of Laplacia $L = \sum_i \lambda_i \psi_i \psi_i^\top$
- Eigenvectors form orthonormal ba $\Psi = \{\psi_i\}$
- Any f can be decomposed $f = \sum_i \alpha_i \psi_i$
- Existing SSL assumption: f uses a few low frequency eigenvectors, i.e., the corresponding α_i are large (non-zero).
- Low frequency eigenvectors: whose eigenvalues are close to zero

New assumption: sparsity

- Allow f to have high frequency eigenvectors, as long as α is sparse (a few large entries)
- Recent advances in compressive sensing determine when learning can happen
  - The signal representation basis is Ψ
  - The measurement basis is the canonical basis I (identity matrix)
  - Labeled data in transductive learning = measurements made with random rows from I

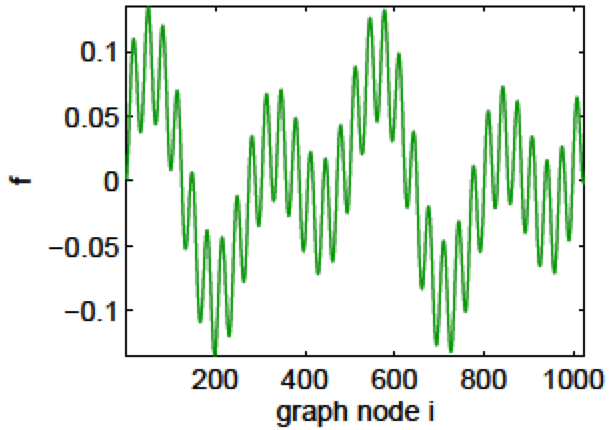
SSL as compressive sensing

- Key quantity: coherence $\mu(\mathrm{I}, \Psi)$ $\propto$ max entry in Ψ
- Theorem: let there be n labeled points, m unlabeled points. Assume α has S≤n+m non-zero entries (but could be anywhere, both low and high fr $n \geq C\mu^2(\mathrm{I}, \Psi) S \log(n+m)$

labeled points is sufficient to exactly learn f.

Example

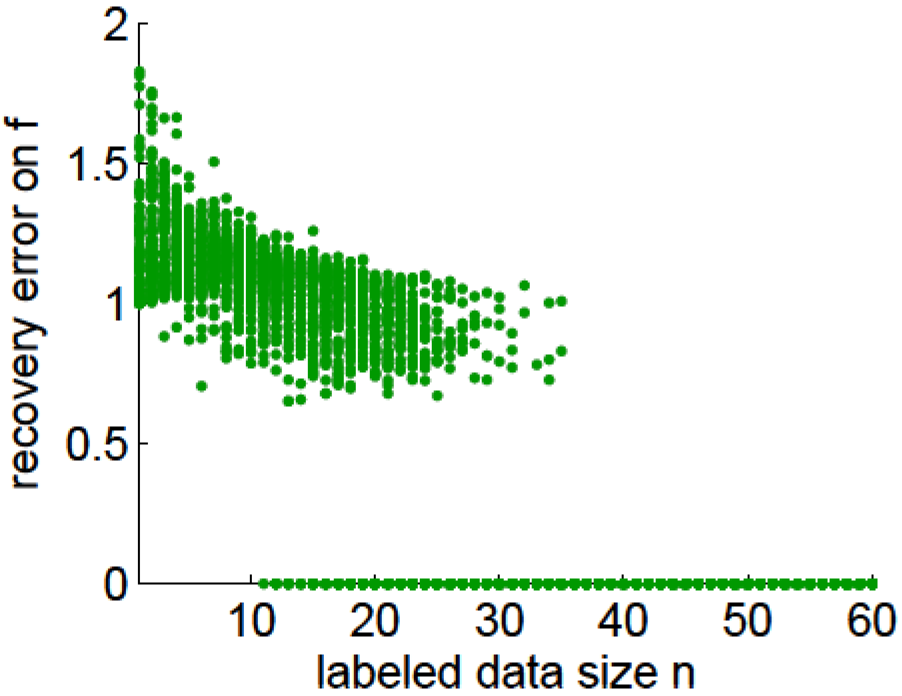
- Unweighted ring graph with 1024 nodes
- Sparsity S=3, nonsmooth funct $f = -\psi_5 - 1.3\psi_8 + \psi_{63}$



- Draw n random points to get label (true f values). Recovery f using L-1 minimization as standard in compressive sensing. Measure recovery error.
- Repeat several times for each n.

Example

- Each trial is a dot
- Exact recovery happens when n>35



- Compressive sensing → transductive learning for sparse but nonsmooth functions

Conclusions

- We have presented three new research directions for graph-based SSL
  - Multi-manifold learning
  - Online SSL
  - Compressive sensing
- We hope to inspire new research, making SSL an even more valuable tool for multimedia analysis.
- We thank the presenter, and you!