Introduction to Graph-based Semi-supervised Learning

MLSS 2007

Practical Session on Graph-based Algorithms in Machine Learning

Matthias Hein and Ulrike von Luxburg

Department of Computer Science, Saarland University, Saarbrücken, Germany
Max Planck Institute for Biological Cybernetics, Tübingen, Germany

Roadmap

- What is semi-supervised learning (SSL) ? What is transduction ?
- The cluster/manifold assumption
- graph-based SSL using regularized least squares
  1. Interpretation in terms of label propagation
  2. Interpretation in terms of a data-dependent kernel
- Experiments

Why semi-supervised learning ?

- Human labels can be expensive and time consuming,
- There is a lot of unlabeled data around us e.g. images and text on the web. The knowledge about the unlabeled data "should" be helpful to build better classifiers,

What is semi-supervised learning ?

Input space X, Output: $\{-1, 1\}$ (binary classification):

- a small set L of labeled data (X_l, Y_l),
- a large set U of unlabeled data X_u.
- notation: n=l+u, total number of data points. T denotes the set of all points.

e.g. a small number of labeled images and a huge number of unlabeled images from the internet.

Definition:

- **Transduction:** Prediction of the labels Y_u of the unlabeled data X_u,
- **SSL:** Construction of a classifier $f : X \to \{-1, 1\}$ on the whole input space (using the unlabeled data).

No !

Because:

- in order to deal with a small amount of labeled data we have to make strong assumptions about the underlying joint probability measure $\mathrm{P}(X, Y)$ e.g. a relation of $\mathrm{P}(X)$ and $\mathrm{P}(Y|X)$.

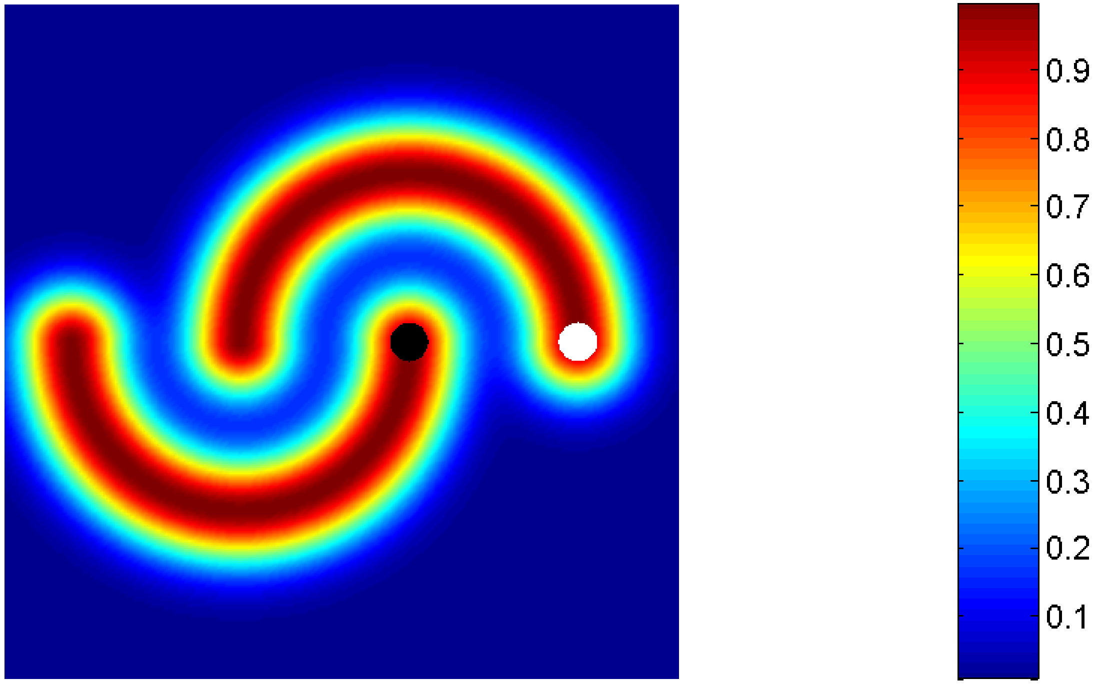
But:

- empirical success of SSL methods shows that unlabeled data can improve performance.
- nice application of SSL from an unexpected side: spectral matting (Levin et al. 2006) a kind of user-interactive segmentation (foreground / background).

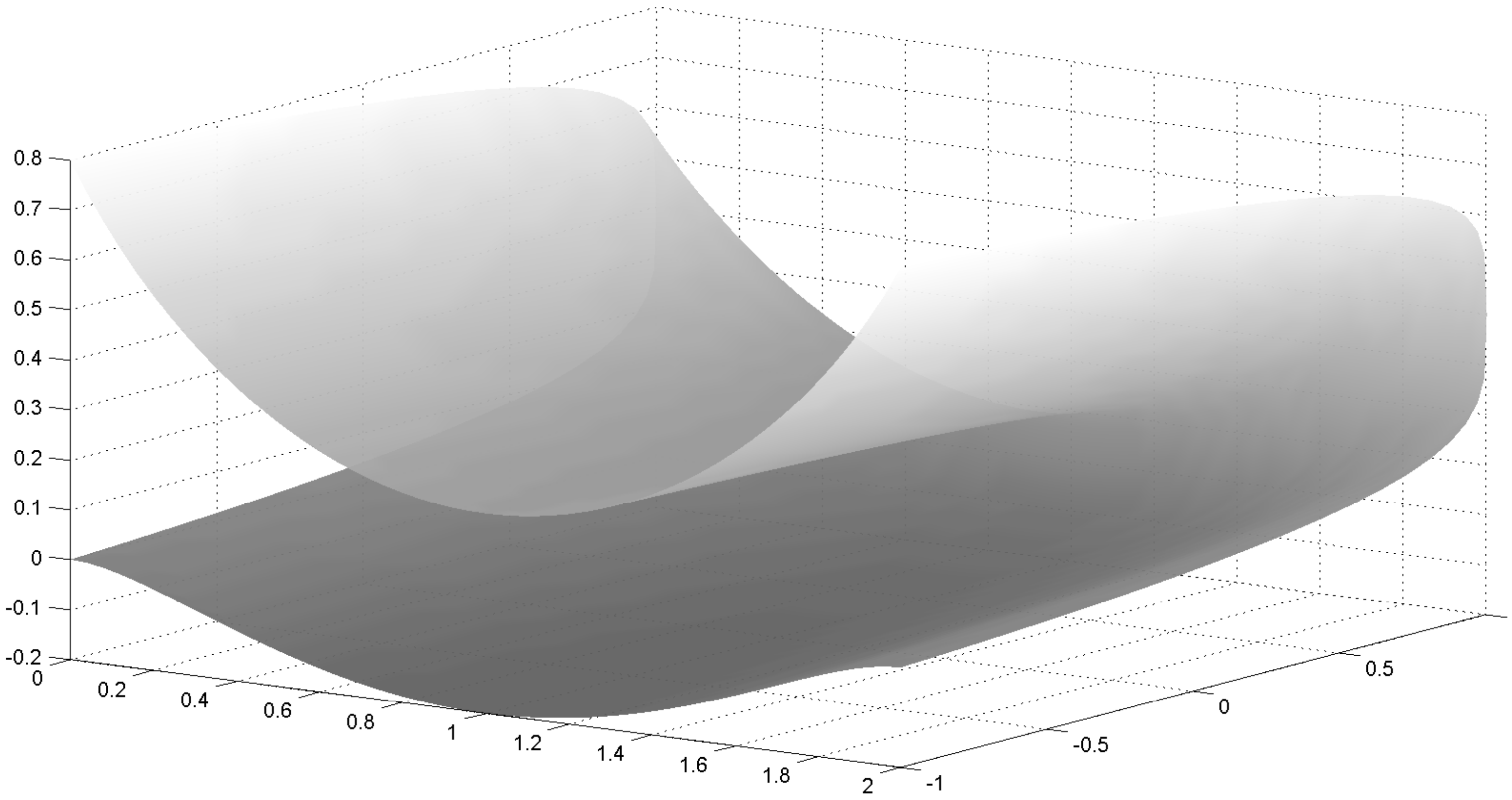
Matting

Left: Input Image with user labels, Right: Image segmentation

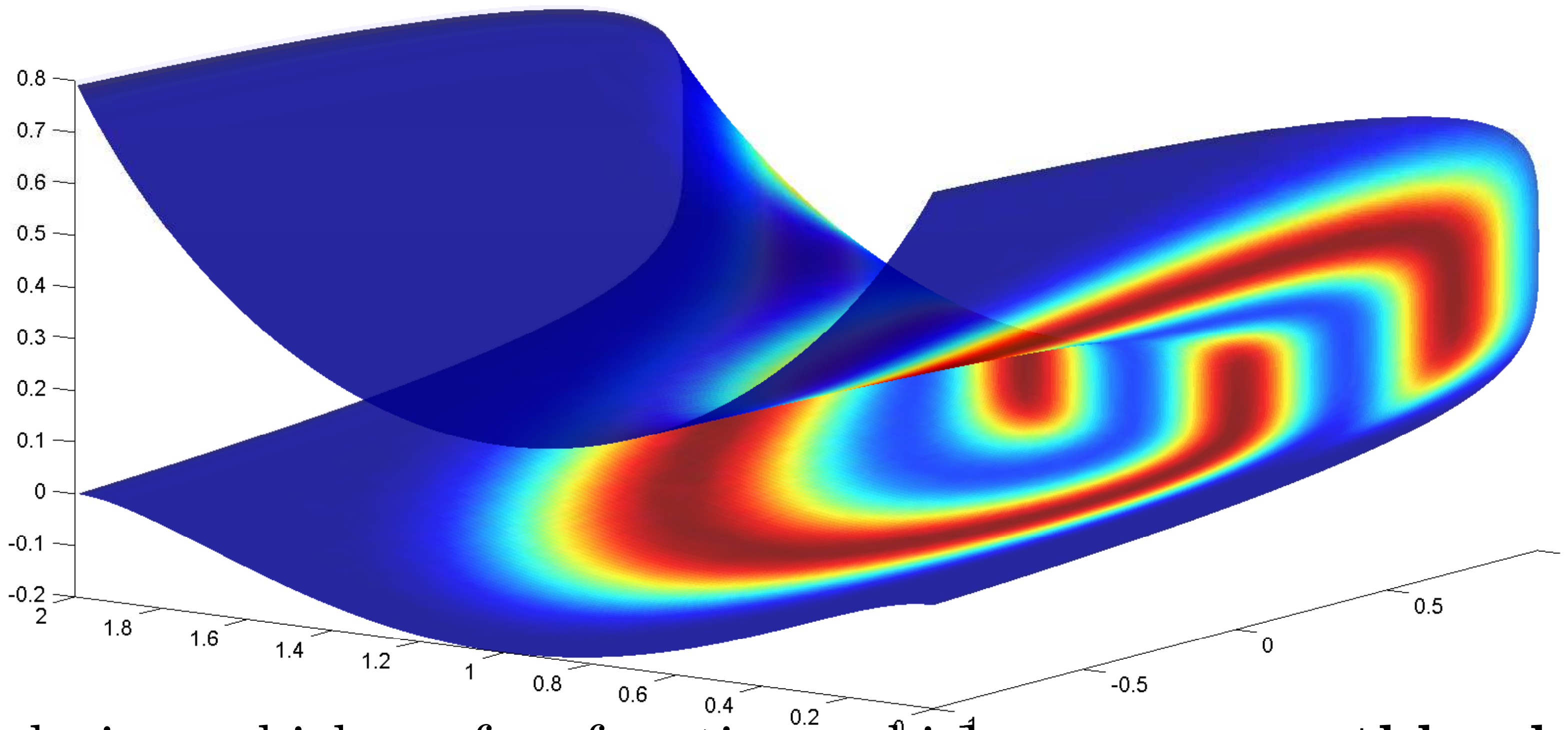
Cluster assumption: points which can be connected via (many) paths through high-density regions are likely to have the same label.



Manifold assumption: each class lies on a separate manifold.



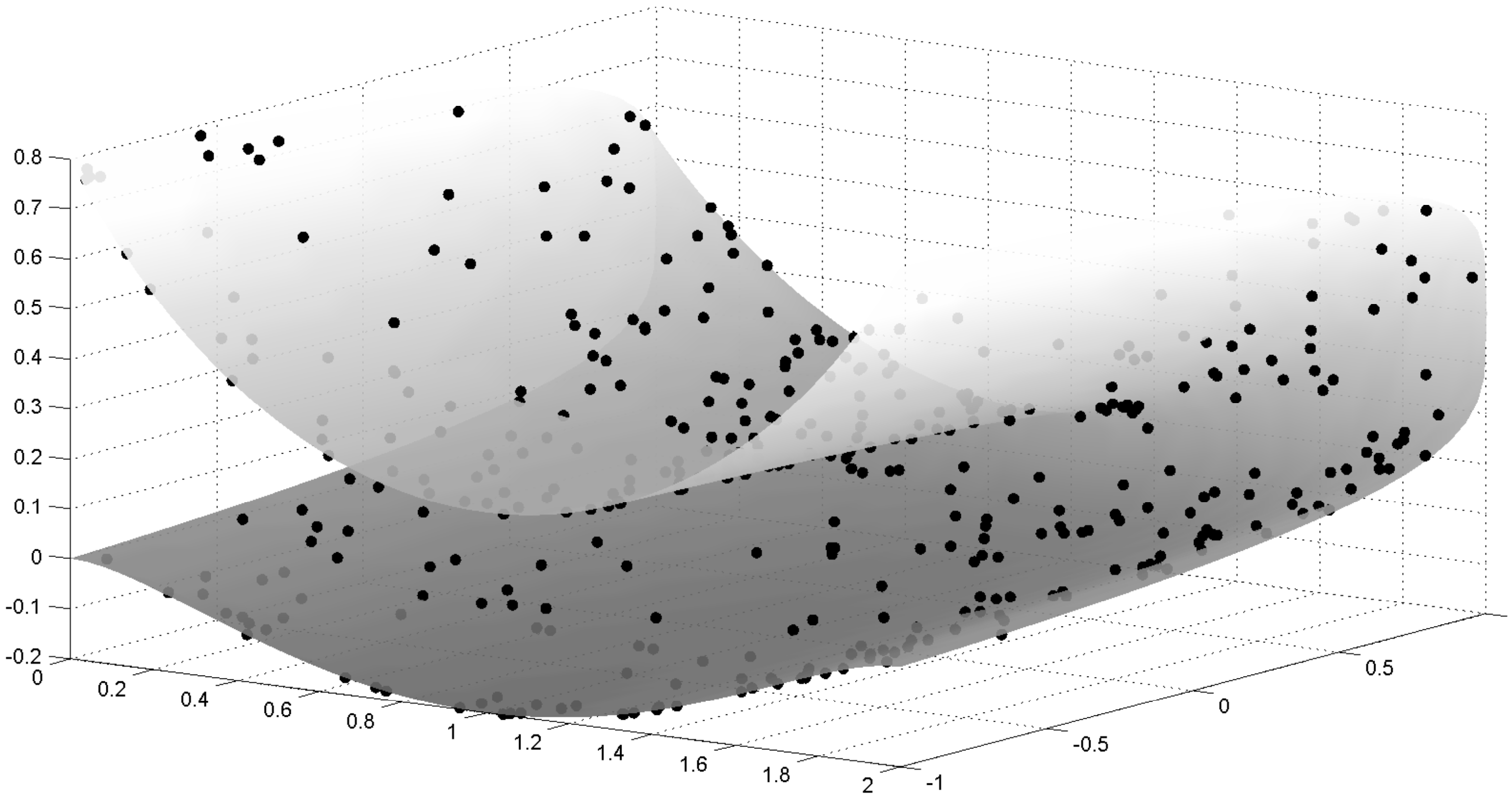
Cluster/Manifold assumption: points which can be connected via a path through high density regions on the data manifold are likely to have the same label.



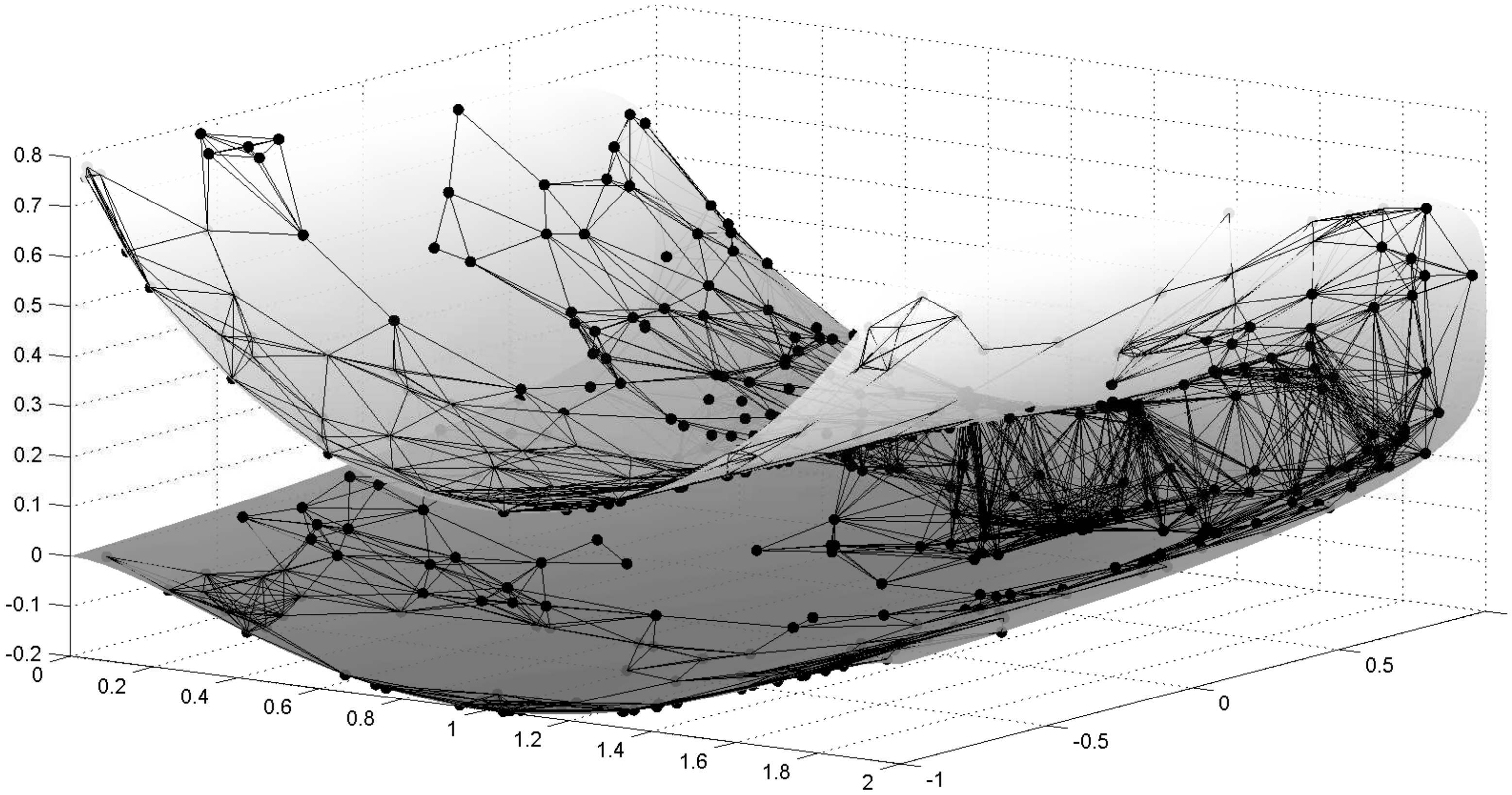
$\Longrightarrow$ Use regularizer which prefers functions which **vary smoothly along the manifold** and **do not vary in high density regions.**

The cluster/manifold-assumption II

Problem: We have only (a lot of) unlabeled and some labeled points and no information about the density and the manifold.



Approach: Use a graph to approximate the manifold (and density).



The cluster/manifold-assumption IV

Define a regularization functional which penalizes functions which vary in high-density regions.

$$\langle f, Lf \rangle = \langle f, (D - W)f \rangle = \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2,$$

where $D = d_i \delta_{ij}$ with $d_i = \sum_{j=1}^{n} w_{ij}$ and the graph Laplacian is defined as $L = D - W$.

For the ϵ-neighborhood graph one can show (Bousquet, Chapelle and Hein (2003), Hein (2006)) under certain technical conditions that as $\epsilon \to 0$ and $n\epsilon^m \to \infty$ (m is dimension of the manifold).

$$\lim_{n \to \infty} \frac{1}{n\epsilon^{m+2}} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 \sim \int_M \|\nabla f\|^2 p(x)^2 dx$$

Transductive Learning via regularized least squares:

Zhu, Ghahramani, Lafferty (2002,2003):

$$\boxed{\mathop{\arg\min}_{f\in\mathbb{R}^n,\, f_L=Y_L} \sum_{i,j\in T}^{n} w_{ij}(f_i - f_j)^2}.$$

Belkin and Niyogi (2003):

$$\boxed{\mathop{\arg\min}_{f\in\mathbb{R}^n} \sum_{i\in L}(y_i - f_i)^2 + \lambda \sum_{i,j\in T} w_{ij}(f_i - f_j)^2}.$$

Zhou, Bousquet, Lal, Weston and Schoelkopf (2003):

$$\boxed{\mathop{\arg\min}_{f\in\mathbb{R}^n} \sum_{i\in T}(y_i - f_i)^2 + \lambda \sum_{i,j\in T} w_{ij}\left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}}\right)^2},$$

where $y_i = 0$ if $i \in U$.

$$\boxed{\arg\min_{f \in \mathbb{R}^n} \sum_{i \in T} (y_i - f_i)^2 + \lambda \sum_{i,j \in T} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2,}$$

where $y_i = 0$ if $i \in U$. Note that

$$f^T (\mathbb{1} - D^{-1/2} W D^{-1/2}) f = \sum_{i,j \in T} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2.$$

The solution f^* can be found as:

$$\boxed{f^* = \left(\mathbb{1} + \lambda (\mathbb{1} - D^{-1/2} W D^{-1/2}) \right)^{-1} Y}$$

or with $S = D^{-1/2} W D^{-1/2}$ and $\alpha = \frac{\lambda}{1+\lambda}$ $(0 < \alpha < 1)$,

$$\boxed{f^* = \frac{1}{1+\lambda} \left[\mathbb{1} - \frac{\lambda}{1+\lambda} S \right]^{-1} Y = (1-\alpha)[\mathbb{1} - \alpha S]^{-1} Y,}.$$

Label Propagation

Interpretation of the solution f^* in terms of label propagation:

$$\boxed{f^* = (1-\alpha)\Big[\mathbb{1} - \alpha S\Big]^{-1} Y}$$

One can show $[\mathbb{1} - \alpha S]^{-1} = \sum_{r=0}^{\infty} \alpha^r S^r$.

$$\boxed{f^* = (1-\alpha)\Big[\mathbb{1} - \alpha S\Big]^{-1} Y = \frac{\sum_{r=0}^{\infty} \alpha^r S^r}{\sum_{r=0}^{\infty} \alpha^r} Y}$$

Solution f^* can be interpreted as the limit $f^* = \lim_{t\to\infty} f_t$ of the iterative scheme f_t with $f(0) = Y$,

$$\boxed{f_{t+1} = \alpha S f_t + (1-\alpha) Y \quad \Rightarrow \quad f_{t+1} = \alpha^t S^t Y + (1-\alpha) \sum_{r=0}^{t} (\alpha S)^r Y,}$$

where $\lim_{t\to\infty} \alpha^t S^t Y = 0$.

Relation to random walks

The solution is given by

$$\boxed{f^* = (1-\alpha)\left[\mathbb{1} - \alpha S\right]^{-1} Y = \frac{\sum_{r=0}^{\infty} \alpha^r S^r}{\sum_{r=0}^{\infty} \alpha^r} Y}$$

Using $S = D^{-1/2} W D^{-1/2}$ we get with the stochastic matrix $P = D^{-1} W$,

$$S = D^{1/2} P D^{-1/2} \quad \text{and} \quad \boxed{S^r = D^{1/2} P^r D^{-1/2}}.$$

Plugging the expression for S^r into the equation for the solution f,

$$\boxed{f^* = D^{1/2} \frac{\sum_{r=0}^{\infty} \alpha^r P^r}{\sum_{r=0}^{\infty} \alpha^r} D^{-1/2} Y}$$

Comments

- All approaches can also be interpreted as kernel machines. Let $L^\dagger$ be the pseudo-inverse of the graph Laplacian. Then

$$K = L^\dagger,$$

is a (data-dependent) kernel on n points. Let $f_i = \sum_{j=1}^n \alpha_j k(x_i, x_j)$. Then

$$f^\top L f = \alpha^\top K^T L K \alpha = \alpha^\top K \alpha.$$

- The structure of the graph influences significantly the result. For high-dimensional data one can improve the performance by using "Manifold Denoising" as a preprocessing method.

Experiments

- Run DemoSSL
- Make yourself familiar with the demo

Experiments I

Does it work ?

Use: Two Moons (Balanced/Unbalanced) in low dimensions (2-5) !

- Find the best parameters for 4 labeled points.

Questions:

- What is your test error ? How stable is it (Draw new labeled points) ?
- What happens if you increase the noise dimensions (30 and 200) ?

Experiments II

Influence of the regularization parameter:

Use: Two Moons (Balanced/Unbalanced) in low dimensions (2-10) and ≈ 10 labeled points!

- Study influence of the regularization parameter (min/max).

Questions:

- What behavior do you observe ?
- Can you explain it ?

Experiments III

The solution f^* of the SSL problem: $\boxed{f^* = D^{1/2} \frac{\sum_{r=0}^{\infty} \alpha^r P^r}{\sum_{r=0}^{\infty} \alpha^r} D^{-1/2} Y.}$

- $\lambda \to \infty$ $(\alpha \to 1)$:
  For a connected graph it holds: $\lim_{m\to\infty} \frac{1}{m} \sum_{r=0}^{m} P_{ij}^r = \pi_j$, where π is the stationary distribution of the the random walk P. This yields

$$f \longrightarrow D^{-1/2} \begin{pmatrix} \pi^T \\ \ldots \\ \pi^T \end{pmatrix} (D^{1/2} Y).$$

- $\lambda \to 0$ $(\alpha \to 0)$:

$$f \longrightarrow Y + \alpha\, S\, Y = Y + \alpha\, D^{1/2}\, P\, D^{1/2}\, Y.$$

Experiments IV

What happens if the cluster assumption is not valid ?

Use: Two Gaussians (Balanced/No Cluster) in low dimensions (2-10) !

Questions:

- How many labels do you need to get a test error below 10%.
- What happens if you increase the dimension ?

Experiments V

What happens if the graph structure is bad ?

Use: Two Gaussians (Balanced/Different variance) in high dimensions (130) with 10 labeled points !

Questions:

- What happens here ?
- compare mutual and symmetric k-nearest neighbor graph. Which is better for this dataset ?
- How could we even improve the performance ?
- What happens if you increase the dimension to 200 ?

Cross validation works (quite well) !

But:

- A lot of parameters usually lead to zero cross-validation error.
- Evaluate other characteristics of the solution (e.g. class proportions in the solution versus class proportions in the labeled set) to choose in this set of parameters.

Conclusions of the practical session

- Graph-based methods work very well if underlying assumptions are satisfied.
- Graph-structure is very important (not well studied yet in machine learning). Graph-structure is as important as variations of algorithms.
- Many applications of graph-based methods and more to come.

References

General Literature on Semi-supervised Learning:

- O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning.* MIT Press, Cambridge, 2006.
- X. Zhu. Semi-supervised learning literature survey. TR-1530. University of Wisconsin-Madison Department of Computer Science, 2005.
- M. Seeger. Learning with labeled and unlabeled data. Technical Report. University of Edinburgh, 2001.

Semi-supervised Learning based on the graph Laplacian:

- X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, 2002.
- X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In T. Fawcett and N. Mishra, editors, *Proc. of the 20nd Int. Conf. on Machine Learning (ICML)*, 2003.
- M. Belkin and P. Niyogi. Semi-supervised learning on manifolds. *Machine Learning*, 56:209–239, 2004.
- D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Adv. in Neur. Inf. Proc. Syst. (NIPS)*, volume 16, pages 321–328, 2004.
- O. Bousquet, O. Chapelle, and M. Hein. Measure based regularization. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Adv. in Neur. Inf. Proc. Syst. (NIPS)*, volume 16. MIT Press, 2004.

Generalizations:

- D. Zhou, B. Schölkopf, and T. Hofmann. Semi-supervised learning on directed graphs. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Adv. in Neur. Inf. Proc. Syst. (NIPS)*, volume 17, pages 1633–1640. MIT Press, 2005.
- D. Zhou and B. Schölkopf. Regularization on discrete spaces. In *27th Pattern Recognition Symposium (DAGM)*, pages 361–368, Berlin, 2005. Springer.

References

Other approaches:

- O. Chapelle and A. Zien. Low density separation. In Z. Ghahramani and R. Cowell, editors, *Proc. of the 10th Int. Workshop on Art. Int. and Stat. (AISTATS)*, 2005.

Theoretical analysis of the graph Laplacian and the induced smoothness functional

- M. Hein, J.-Y. Audibert, and U. von Luxburg. From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. In *Proc. of the 18th Conf. on Learning Theory (COLT)*, pages 486–500, 2005.
- M. Belkin and P. Niyogi. Towards a theoretical foundation for Laplacian-based manifold methods. In P. Auer and R. Meir, editors, *Proc. of the 18th Conf. on Learning Theory (COLT)*, pages 470–485, Berlin, 2005. Springer.
- M. Hein. Uniform convergence of adaptive graph-based regularization. In G. Lugosi and H. Simon, editors, *Proc. of the 19th Conf. on Learning Theory (COLT)*, pages 50–64, 2006.
- M. Hein, J.-Y. Audibert, and U. von Luxburg. Graph Laplacians and their convergence on random neighborhood graphs, 2006. accepted at JMLR, available at arXiv:math.ST/0608522.

Denoising as Preprocessing:

- M. Hein, M. Maier Manifold denoising. In *Adv. in Neural Inf. Proc. Syst. 19 (NIPS)*, 2007.

Matting:

- A. Levin, A. Rav-Acha, D. Lischinski A Closed Form Solution to Natural Image Matting. CVPR 2006.