Semi-Supervised Learning by Higher Order Regularization

Xueyuan Zhou\textsuperscript{1} and Mikhail Belkin\textsuperscript{2}

\textsuperscript{1}Computer Science, University of Chicago
\textsuperscript{2}Computer Science and Engineering, The Ohio State University
AISTATS 2011, Ft. Lauderdale, FL, USA
Outline

- Semi-Supervised Learning
- Graph Laplacian Regularization
- Problem of Graph Laplacian Regularization
- Solution: Iterated Laplacian Regularization
- Experiments
- Summary
Semi-Supervised Learning

- Semi-Supervised Learning (SSL)
  - A few labeled examples, with an additional very large unlabeled example set (partially observed data)

- Goal
  - Inductive inference: estimate an unknown function over the whole domain
  - Transductive inference: estimate the values of an unknown function at particular points
Semi-Supervised Learning

- **Semi-Supervised Learning (SSL)**
  - **Given**
    \[ X_L = \{ x_1, x_2, \cdots, x_l \} \]
    \[ Y_L = \{ y_1, y_2, \cdots, y_l \} \]
    \[ X_U = \{ x_{l+1}, x_{l+2}, \cdots, x_{l+u} \} \]
    \[ |X_L| << |X_U|, X = X_L \cup X_U, n = l + u \]
  - **SSL**: find a function \( f(x) \) on the whole domain (inductive), or on the unlabeled set \( Y_U = \{ y_{l+1}, y_{l+2}, \cdots, y_{l+u} \} \) (transductive)
Semi-Supervised Learning

- **Semi-Supervised Learning (SSL)**
  - Given
    \[ X_L = \{x_1, x_2, \cdots, x_l\} \]
    \[ Y_L = \{y_1, y_2, \cdots, y_l\} \]
    \[ X_U = \{x_{l+1}, x_{l+2}, \cdots, x_{l+u}\} \]
    \[ |X_L| << |X_U|, X = X_L \cup X_U, n = l + u \]
  - SSL: find a function \( f(x) \) on the whole domain (inductive), or on the unlabeled set \( Y_U = \{y_{l+1}, y_{l+2}, \cdots, y_{l+u}\} \) (transductive)

- **Model:**
  - \( x_i \in \Omega \subset \mathbb{R}^d, (\Omega \text{ can be a manifold } \mathcal{M}) \)
  - \( x_i \sim p(x), (x_i, y_i) \sim p(x, y) \)
  - \( 0 < a \leq p(x) \leq b < +\infty \)
Semi-Supervised Learning

Examples

- Internet search queries/text/image classification
  - x: query/text document/image, y: predefined class
- Estimation of functions on social network: Linkedin
  - x: user profile, y: e.g., interested in machine learning job
- Information retrieval: image ranking as regression
  - x: images, y: ranking score
- etc
Graph Laplacian Regularization

- A popular algorithm: find a “smooth” function
  - Similar $x \rightarrow$ similar $y$
    \[
    \min_{f \in \mathbb{R}^n} \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2
    \]
    \[
    \text{s.t. } f(X_L) = Y_L
    \]
  - Similarity weight $w(x, y)$
  - Alternatively
    \[
    \min_{f \in \mathbb{R}^n} \sum_{x_i \in X_L} (f(x_i) - y_i)^2 + \mu \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2
    \]
Graph Laplacian Regularization

- A popular algorithm: find a “smooth” function
  - Similar x \(\Rightarrow\) similar y
    \[
    \min_{f \in \mathbb{R}^n} \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2 \\
    \text{s.t. } f(X_L) = Y_L
    \]
  - Similarity weight \(w(x, y)\)
  - Alternatively
    \[
    \min_{f \in \mathbb{R}^n} \sum_{x_i \in X_L} (f(x_i) - y_i)^2 + \mu \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2
    \]

- Estimation on an undirected graph \(G(V,E)\)
  - \(V = X\)
  - \(E = \{e_{ij} : (x_i, x_j)\}, w(x_i, x_j) = \frac{1}{t^{d/2}}e^{-\frac{||x_i-x_j||^2}{t}}\)
  - E.g., kNN graphs

Figure from, X. Zhu et al., ICML 2003.
Graph Laplacian Regularization

- Graph Laplacian Regularizer as a Semi-norm

\[
\frac{1}{2} \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2 = f^T L f
\]

where \( L = D - W \) is called the graph Laplacian
Graph Laplacian Regularization

- **Graph Laplacian Regularizer as a Semi-norm**
  \[
  \frac{1}{2} \sum_{x_i, x_j \in X} w(x_i, x_j)(f(x_i) - f(x_j))^2 = f^T L f
  \]
  where \( L = D - W \) is called the graph Laplacian

- **Analysis in An Asymptotic Setting**
  - Fixed labeled set, increasing unlabeled points
  - Limit as \( n \) increases, \( t \) decreases

\[
\frac{1}{n^{2t(d/2+1)}} f^T L f \xrightarrow{a.s.} c \int_{\Omega} \|\nabla f(x)\|^2 p^2(x) dx
\]
Graph Laplacian Regularization

- SSL with Infinite Unlabeled Data and Fixed Labeled Data

\[
\min \int_{\Omega} \| \nabla f(x) \|^2 p^2(x) dx \\
\text{s.t. } f(X_L) = Y_L
\]

- “Smooth”: small penalty/semi-norm, \textit{intuitively}
- Use \textit{sign}(f) to obtain the class in classification
- Same analysis to regularized Least Squares
Graph Laplacian Regularization

- SSL with Infinite Unlabeled Data and Fixed Labeled Data

\[
\min \int_{\Omega} \| \nabla f(x) \|^2 p^2(x) \, dx \\
\text{s.t. } f(X_L) = Y_L
\]

- “Smooth”: small penalty/semi-norm, **intuitively**
- Use \( \text{sign}(f) \) to obtain the class in classification
- Same analysis to regularized Least Squares

- Tip of the iceberg: \( \min_{f \in ?} \)
Problem of Graph Laplacian Regularization

- In SSL, the more unlabeled data, the better results we expect
Problem of Graph Laplacian Regularization

- In SSL, the more unlabeled data, the better results we expect
- However...

“Indicator” functions of labeled points.
Problem of Graph Laplacian Regularization

- In SSL, the more unlabeled data, the better results we expect
- However...

The more unlabeled data we have, the less stable the classifier gets, and the worse the results become. The opposite of the intuition “smooth”!
Problem of Graph Laplacian Regularization

- Reason: is the solution “smooth”? 

\[ I(f) = \int_{\Omega} \| \nabla f(x) \|^2 dx = \int_{\Omega} \sum_{i=1}^{d} \left| \frac{\partial f(x)}{\partial x_i} \right|^2 dx \approx d \left( \frac{1}{h} \right)^2 hh \cdots h = O(h^{d-2}) \]

- For \( d > 2 \) as \( h \to 0 \), \( I(f) \to 0 \), but \( |\nabla f(x)\| \to +\infty \), same for \( d = 2, \ldots \); this means for the solutions (indicator functions), \( I(f) = 0 \) (the penalty is small!)

- Indicator functions: nearby points have quite different function values, so not “smooth”, contrary to the intuition!
Problem of Graph Laplacian Regularization

Reason: is the solution “smooth”?

\[ I(f) = \int \Omega \| \nabla f(x) \|^2 dx = \int \Omega \sum_{i=1}^{d} |\frac{\partial f(x)}{\partial x_i}|^2 dx \approx d\left(\frac{1}{h}\right)^2 hh \cdots h = O(h^{d-2}) \]

For \( d > 2 \) as \( h \to 0 \), \( I(f) \to 0 \), but \( |\nabla f(x)| \to +\infty \), same for \( d = 2, \ldots \); this means for the solutions (indicator functions), \( I(f) = 0 \) (the penalty is small!)

Indicator functions: nearby points have quite different function values, so not “smooth”, contrary to the intuition!

The semi-normed space is too rich when \( d \geq 2 \)

\[ \{ f : f \in L_2, \int \Omega \| \nabla f(x) \|^2 dx < +\infty \} \]

Note: in \( d = 1 \), no such problem.
Solution: Iterated Laplacian Regularization

- Problem: Semi-normed (solution) space is too large
  - Sobolev Space

\[ H^m(\Omega) = \{ f : D^\alpha f \in L_2(\Omega), \forall \alpha, \text{s.t.} |\alpha| \leq m \}, \quad D^\alpha f = \frac{\partial |\alpha| f}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \]

\[ \{ f : f \in L_2, \int_\Omega \| \nabla f(x) \|^2 dx < +\infty \} \leftrightarrow H^1(\Omega) \]
Solution: Iterated Laplacian Regularization

- Problem: Semi-normed (solution) space is too large
  - Sobolev Space
    \[ H^m(\Omega) = \{ f : D^\alpha f \in L_2(\Omega), \forall \alpha, \text{s.t.} |\alpha| \leq m \}, \quad D^\alpha f = \frac{\partial|\alpha|f}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \]
    \[ \{ f : f \in L_2, \int_\Omega \| \nabla f(x) \|^2 dx < +\infty \} \leftrightarrow H^1(\Omega) \]

- Solution: Shrink the solution space
  \[ H^j(\Omega) \subset H^k(\Omega), \quad j > k \]

- Need a Sobolev norm of higher order to shrink the solution space; the Sobolev embedding theorem tells more.
Solution: Iterated Laplacian Regularization

- Problem: Semi-normed (solution) space is too large
  - Sobolev Space
    \[ H^m(\Omega) = \{ f : D^\alpha f \in L_2(\Omega), \forall \alpha, s.t. |\alpha| \leq m \}, D^\alpha f = \frac{\partial|\alpha|f}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \]
    \[ \{ f : f \in L_2, \int_\Omega \| \nabla f(x) \|^2 dx < +\infty \} \leftrightarrow H^1(\Omega) \]

- Solution: Shrink the solution space
  \[ H^j(\Omega) \subset H^k(\Omega), j > k \]
  - Need a Sobolev norm of higher order to shrink the solution space; the Sobolev embedding theorem tells more.

- Key: How to implement this idea in a practical algorithm
Solution: Iterated Laplacian Regularization

- How to implement Sobolev Norm
  - Original definition: $\| D^m f \|^2$, not easy to implement for random samples
Solution: Iterated Laplacian Regularization

- **How to implement Sobolev Norm**
  - Original definition: $\|D^m f\|^2$, *not easy to implement* for random samples
  - *Easy to implement* using iterated Laplacian semi-norm

$$I_s^d(f) = \int_\Omega f(x) \Delta^s f(x) \, dx = \sum_{i=1}^\infty |\hat{f}_i|^2 \lambda_i^s, \ s \geq 0$$

$$\Delta[\Delta^{s-1} \phi_i] = \lambda_i^s \phi_i$$

- Equivalent definition using Fourier basis: eigenfunctions of a Laplacian form $L_2$ basis

$$f \in L_2, f = \sum_{i=1}^\infty \hat{f}_i \phi_i, \ \hat{f}_i = \langle f, \phi_i \rangle, \ \Delta \phi_i = \lambda_i \phi_i$$

- **Note**, $\lambda_i$ is increasing
Solution: Iterated Laplacian Regularization

- **Iterated Laplacian Semi-normed Space**

  \[ D_s(\Omega) = \{ f \in L_2(\Omega) : I_s^d(f) < \infty \} \]

  \[ D_s(\Omega) \subset H^s(\Omega) \]

- Alternative way of smoothness controlled by \( s \), compared to \( C^k \) functions controlled by \( k \).
- From point-wise functions to \( L_2 \) functions.
Solution: Iterated Laplacian Regularization

- Least Squares by Higher Order Regularization Using the iterated graph Laplacian (least squares or interpolation)

\[ \min_{f \in \mathbb{R}^n} \sum_{x_i \in X} (f(x_i) - y_i)^2 + \mu f^T L^m f \]

- Limit of \( I_{m,n}^d(f) = f^T L^m f \)
  - For uniform density, \( I_m^d(f) = \int_\Omega f(x) \Delta^m f(x) dx \)
  - For nonuniform density, use weighted Laplacian
  - Need proper boundary conditions when domains/manifolds have boundaries
A Toy Example

Mixture of two unit variance Gaussians in $\mathbb{R}^{20}$ at $(\pm 1.5, 0, 0, \ldots, 0)$, one labeled point for each Gaussian

\[
x_1 \leq 0 \ (y = +1) \\
x_1 > 0 \ (y = -1)
\]
Experiments

More Examples: Regular vs Iterated Laplacian Regularization

\[ \min_{f \in \mathbb{R}^n} \sum_{x_i \in X_L} (f(x_i) - y_i)^2 + \mu f^T L^m f \]

Classification errors % with std for \( m = 1 \) and \( m = 4 \).

<table>
<thead>
<tr>
<th>DATA SET</th>
<th>( m = 1 )</th>
<th>( m = 4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST 3vs8</td>
<td>7.5 ± 1.5</td>
<td>5.6 ± 1.3</td>
</tr>
<tr>
<td>MNIST 4vs9</td>
<td>12.2 ± 2.9</td>
<td>8.0 ± 2.6</td>
</tr>
<tr>
<td>PCMAC</td>
<td>16.6 ± 2.4</td>
<td>11.5 ± 1.4</td>
</tr>
<tr>
<td>AUT-AVN</td>
<td>13.7 ± 2.6</td>
<td>10.1 ± 1.3</td>
</tr>
<tr>
<td>REAL-SIM</td>
<td>9.4 ± 3.3</td>
<td>5.8 ± 0.8</td>
</tr>
<tr>
<td>CCAT</td>
<td>24.0 ± 2.8</td>
<td>21.5 ± 2.8</td>
</tr>
<tr>
<td>GCMAT</td>
<td>13.1 ± 1.5</td>
<td>12.0 ± 1.3</td>
</tr>
<tr>
<td>GENE-P</td>
<td>39.3 ± 9.1</td>
<td>29.0 ± 8.9</td>
</tr>
<tr>
<td>GENE-B</td>
<td>45.3 ± 7.3</td>
<td>41.9 ± 9.2</td>
</tr>
</tbody>
</table>

Noniterated Laplacian  Iterated Laplacian
# Experiments

## More Examples:

Classification errors % for SSL Benchmark.

<table>
<thead>
<tr>
<th></th>
<th>QC+CMN</th>
<th>LapRLS</th>
<th>Best</th>
<th>IterLap</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>X_L</td>
<td>= 10$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>g241c</td>
<td>39.96</td>
<td>43.95</td>
<td>22.76</td>
<td>18.01</td>
</tr>
<tr>
<td>g241d</td>
<td>46.55</td>
<td>45.68</td>
<td>18.64</td>
<td>20.99</td>
</tr>
<tr>
<td>Digit1</td>
<td>9.80</td>
<td>5.44</td>
<td>5.44</td>
<td>6.54</td>
</tr>
<tr>
<td>USPS</td>
<td>13.61</td>
<td>18.99</td>
<td>13.61</td>
<td>13.10</td>
</tr>
<tr>
<td>BCI</td>
<td>50.36</td>
<td>48.97</td>
<td>46.90</td>
<td>46.71</td>
</tr>
<tr>
<td>Text</td>
<td>40.79</td>
<td>33.68</td>
<td>27.15</td>
<td>38.84</td>
</tr>
<tr>
<td>$</td>
<td>X_L</td>
<td>= 100$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>g241c</td>
<td>22.05</td>
<td>24.36</td>
<td>13.49</td>
<td>14.82</td>
</tr>
<tr>
<td>g241d</td>
<td>28.20</td>
<td>26.46</td>
<td>4.95</td>
<td>10.55</td>
</tr>
<tr>
<td>Digit1</td>
<td>3.15</td>
<td>2.92</td>
<td>2.44</td>
<td>2.22</td>
</tr>
<tr>
<td>USPS</td>
<td>6.36</td>
<td>4.68</td>
<td>4.68</td>
<td>3.96</td>
</tr>
<tr>
<td>BCI</td>
<td>46.22</td>
<td>31.36</td>
<td>31.36</td>
<td>43.78</td>
</tr>
<tr>
<td>Text</td>
<td>25.71</td>
<td>23.57</td>
<td>23.09</td>
<td>25.77</td>
</tr>
</tbody>
</table>


$$
\min_{f \in \mathbb{R}^n} \sum_{x_i \in X_L} (f(x_i) - y_i)^2 + \mu f^T L^m f + \epsilon \|f\|_{L_2}^2
$$

**QC+CMN**: Chapter 11, $m=1$.

**LapRLS**: Chapter 12.

**Best**: The best from 13 different algorithms.

**IterLap**: iterated Laplacian, $m>1$

$$
\|f\|_{L_2}^2 \leftrightarrow L_2(\Omega) = H^0(\Omega)
$$
Summary

- Use regularizer $f^T L^m f$ instead of $f^T L f$ in most applications
  - Sound theory support + improvement in practice

- Intuition is **IMPORTANT**, but we need **MORE**: careful mathematical analysis
  - $\int_{\Omega} \|\nabla f(x)\|^2 dx$ is not “good” to describe smoothness in $d \geq 2$
Summary

- Use regularizer $f^T L^m f$ instead of $f^T L f$ in most applications
  - Sound theory support + improvement in practice

- Intuition is \textit{IMPORTANT}, but we need \textit{MORE}: careful mathematical analysis
  - $\int_{\Omega} \| \nabla f(x) \|^2 dx$ is not “good” to describe smoothness in $d \geq 2$

Thanks to Todd Dupont, Nathan Srebro, Boaz Nadler for helpful discussions.
Summary

- Use regularizer $f^T L^m f$ instead of $f^T L f$ in most applications
  - Sound theory support + improvement in practice

- Intuition is *IMPORTANT*, but we need *MORE*: careful mathematical analysis
  - $\int_{\Omega} \|\nabla f(x)\|^2 dx$ is not “good” to describe smoothness in $d \geq 2$

Thank you!

Thanks to Todd Dupont, Nathan Srebro, Boaz Nadler for helpful discussions.