Higher Order Poisson Regularization for Graph-Based Semi-Supervised Learning

Dingjun Bian bianx068@umn.edu School of Mathematics, University of Minnesota
Jeff Calder jwcalder@umn.edu School of Mathematics, University of Minnesota

Summary

We study the theoretical properties and performance of higher order Poisson regularization for graph-based semi-supervised learning (SSL). In particular, we prove the existence of solution for the higher order Poisson regularization problems. Moreover, we show that the solution is unique up to a constant function using a variational approach. We then develop an algorithm to solve for the higher order Poisson learning problems and compare the performance of the algorithm with other techniques such as Poisson learning and Laplace learning on common data sets for semi-supervised machine learning under very low label rates.

Mathematical Framework of SSL

General Mathematical Framework of Learning
Let $\mathcal{X}$ be the set of datapoints that we are interested in doing classification or regression with. In practice, $\mathcal{X}$ is typically $\mathbf{R}^d$, where d represents the number of features of the data. Let $\mathcal{Y}$ be the set of labels. Similarly, $\mathcal{Y}$ is typically $\mathbf{R}^k$ where k is the number of classes. In particular, for a k-class classification problem, the class labels are normally the standard basis vectors in $\mathbf{R}^k$ (one-hot vectors) $e_1, \cdots, e_k \in \mathcal{Y}$.

Mathematical Framework of Semi-Supervised Learning
In semi-supervised machine learning, we would be given a training subset $\Gamma := \{x_1, \cdots, x_L\} \subseteq \mathcal{X}$ with corresponding labels given as $g : \Gamma \to \mathcal{Y}$ and a subset $\{x_{L+1}, \cdots, x_{L+T}\} \subseteq \mathcal{X}$. The goal of an transductive semi-supervised machine learning algorithm is to learn a function

$$u : \{x_1, \cdots, x_{L+T}\} \to \mathcal{Y} \ \ s.t. \ \ u(x) \approx g(x) \text{ when } x \in \Gamma$$

using the unlabeled data $\{x_{L+1}, \cdots, x_{L+T}\}$ to aid learning.

Intuitions for Semi-Supervised Learning

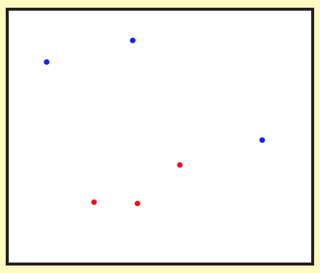

Figure 1. Before given any unlabeled datapoints

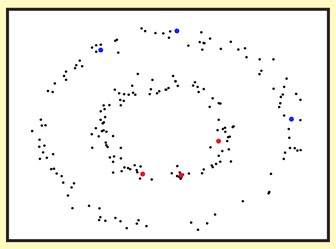

Figure 2. After given the unlabeled datapoints

We note that the the best classification discriminant that we could use is a linear discriminant that splits the blue and red labeled datapoints in Figure 1. However, after when unlabeled datapoints are given as in Figure 2, we can use a non-linear discriminant that encircles the inner ring for classification. In other words, we could now identify the general data structure and thus make a more informed decision on the decision boundary.

Graph-Based Semi-Supervised Learning
Semi-Supervised Learning models are usually constructed through graphs. That is, we are given graph $(\mathcal{X}, \mathcal{W})$ where $\mathcal{X} \subseteq \mathbf{R}^d$ are the vertices and $\mathcal{W} = (w_{xy})_{x,y\in\mathcal{X}}$ are the non-negative edge weights. Moreover, we have $w_{xy} \approx 1$ if x, y similar, and $w_{xy} \approx 0$ when dissimilar. These weights are normally calculated using the k-nearest neighbor weights. The SSL model is then constructed based on the ***semi-supervised smoothness assumption:*** Similar points $x, y \in \mathcal{X}$ in high density regions of the graph should have similar labels.

References

[1] D. Bian. Higher Order Poisson Regularization for Graph-Based Semi-Supervised Learning. University of Minnesota Honors Thesis in Mathematics, 2021.

[2] J. Calder, B. Cook, M. Thorpe, and D. Slepčev. Poisson Learning: Graph based semi-supervised learning at very low label rates. *International Conference on Machine Learning (ICML), PMLR 119:1306–1316*, 2020.

[3] X. Zhou, M. Belkin. Semi-supervised learning by higher order regularization *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, PMLR 15:892-900*, 2011.

Acknowledgments: I want to express my deepest appreciation to Professor Jeff Calder for advising me on this thesis and introducing me to the beauty of applied mathematics. I would also like to express my sincere appreciation to Professor Gilad Lerman, Professor William Leeb and Professor Li Wang for reviewing this thesis and providing me with valuable feedback.

Higher Order Poisson Learning

We propose the higher order Poisson (HOP) learning

$$\mathcal{L}^m u(x) = \sum_{y\in\Gamma} (g(y) - \overline{g})\delta_{xy} \ \text{ s.t. } \ \sum_{x\in\mathcal{X}} u(x) = 0,$$

where $\overline{g} = \frac{1}{|\Gamma|}\sum_{y\in\Gamma} g(y)$ and $m \in \mathbf{Z}^+$. The label decision for vertex $x \in \mathcal{X}$ is determined by the largest component of $u(x)$. That is, we have

$$\ell(x) = \underset{j\in\{1,\ldots,k\}}{\operatorname{argmax}} \{u_j(x)\}.$$

The Existence and Uniqueness of Solution to HOP Learning Problems

Theorem 1. *Assume that $\mathcal{G}$ is connected. Then there exists a unique minimizer $u \in \ell_0^2(\mathcal{X})$ of*

$$\min_{u\in\ell_0^2(\mathcal{X})} \left\{ \frac{1}{2}(u, \mathcal{L}^m u) - \sum_{x\in\Gamma} (g(x) - \overline{g}) \cdot u(x) \right\},$$

and furthermore it satisfies the Euler-Lagrange equation for higher order Poisson learning

$$\mathcal{L}^m u(x) = \sum_{y\in\Gamma} (g(y) - \overline{g})\delta_{xy} \ \ s.t. \ \ \sum_{x\in\mathcal{X}} u(x) = 0,$$

where $\overline{g} = \frac{1}{|\Gamma|}\sum_{y\in\Gamma} g(y)$ and $m \in \mathbf{Z}^+$.

Algorithm Development: Spectral Numerical Approximation

To solve the Euler-Lagrange equation for HOP learning problems, we need to solve a linear system with high conditional numbers. This implies that the system is ill-conditioned and the runtime for solving it is intractable. Therefore, we shall use spectral numerical approximation to reduce the computational complexity.

Theorem 2 (Spectral Numerical Approximation for HOP learning)**.** *We have by spectral decomposition that*

$$(\mathcal{L}_N^m)^{-1} \approx (\mathcal{L}_N^m)^{-1} = V_N \Lambda_{N,m} V_N^T$$

with V_N being the matrix of first N (in terms of the ascending order of eigenvalues) eigenvectors as its columns and $\Lambda_{N,m} = diag((\lambda_i^{-m})_{i=1,\cdots,N})$ being the diagonal matrix with diagonal entries equal to the inverse of the first N eigenvalues. The solution of the Euler Lagrange equation for HOP learning problems is then approximated by

$$u(x) \approx (\mathcal{L}_N^m)^{-1} \sum_{y\in\Gamma} (g(y) - \overline{g})\delta_{xy}.$$

Simulation Results

We compare here the performance of the HOP learning with the standard Poisson learning (equivalent to the special case of HOP when $m = 1$) on MNIST, Fashion-MNIST, and Cifar-10. In almost all cases, we see that HOP learning performs slightly better than the standard Poisson learning due to the flexibility of our HOP learning algorithm in choosing the spectral cutoff N and the Laplacian order m. We present here the mean accuracy score of HOP learning and the standard Poisson learning after running 100 trials.

	# Labels per class	**1**	**2**	**3**	**4**	**5**
MNIST	HOP with $N = 20$ & $m = 1.0$	**91.6 (3.9)**	**94.3 (1.6)**	**95.0 (0.8)**	95.1 (0.6)	95.3 (0.5)
	HOP with $N = 30$ & $m = 1.0$	90.9 (3.8)	94.3 (1.5)	94.9 (1.1)	**95.3 (0.9)**	**95.6 (0.8)**
	Poisson	90.2 (4.0)	93.6 (1.6)	94.5 (1.1)	94.9 (0.8)	95.3 (0.7)
	# Labels per class	**1**	**2**	**3**	**4**	**5**
Fashion-MNIST	HOP with $N = 10$ & $m = 0.4$	**61.7 (5.2)**	65.6 (4.1)	68.5 (2.6)	69.5 (2.2)	69.7 (2.1)
	HOP with $N = 20$ & $m = 0.7$	60.7 (4.8)	**65.8 (4.4)**	69.4 (2.8)	70.6 (2.6)	71.4 (2.3)
	HOP with $N = 30$ & $m = 0.8$	59.9 (4.7)	65.6 (4.2)	**69.5 (2.7)**	70.8 (2.5)	72.0 (2.3)
	HOP with $N = 30$ & $m = 0.7$	59.6 (4.7)	65.5 (4.2)	69.4 (2.7)	**70.9 (2.5)**	**72.1 (2.2)**
	Poisson	60.8 (4.6)	**66.1 (3.9)**	**69.6 (2.6)**	**71.2 (2.2)**	**72.4 (2.3)**
	# Labels per class	**1**	**2**	**3**	**4**	**5**
Cifar-10	HOP with $N = 20$ & $m = 0.3$	**41.8 (5.5)**	47.7 (5.3)	51.0 (3.9)	53.1 (3.1)	54.4 (2.7)
	HOP with $N = 20$ & $m = 0.1$	41.7 (5.5)	**47.8 (5.3)**	**51.0 (3.8)**	**53.2 (3.1)**	**54.5 (2.7)**
	Poisson	40.7 (5.5)	46.5 (5.1)	49.9 (3.4)	52.3 (3.1)	53.8 (2.6)

We also present the contour graph of our HOP learning algorithm accuracy score below. Note that we are only presenting the accuracy contour graph for MNIST and Fashion-MNIST here, but the same pattern is applicable to the contour graph of Cifar-10: The lower spectral cutoff N we use for our HOP algorithm, the more likely we will be able to obtain a higher accuracy score. One can connect this observation to the idea of spectral embedding and clustering, where the higher eigenvector we choose to include in our approximation, the more likely we will overfit our model.