Journal of Machine Learning **ISSN**: 2790-2048(e), 2790-203X(p)

Spectral Neural Networks: Approximation Theory and Optimization Landscape

Chenghui Li * [1], Rishi Sonthalia † [2], and Nicolás García Trillos ‡ [1]

[1]Department of Statistics, University of Wisconsin Madison, Madison, Wisconsin 53706, USA.
[2]Department of Mathematics, Boston College, Boston, MA 02109, USA.

Abstract. There is a large variety of machine learning methodologies that are based on the extraction of spectral geometric information from data. However, the implementations of many of these methods often depend on traditional eigensolvers, which present limitations when applied in practical online big data scenarios. To address some of these challenges, researchers have proposed different strategies for training neural networks as alternatives to traditional eigensolvers, with one such approach known as spectral neural network (SNN). In this paper, we investigate key theoretical aspects of SNN. First, we present quantitative insights into the tradeoff between the number of neurons and the amount of spectral geometric information a neural network learns. Second, we initiate a theoretical exploration of the optimization landscape of SNN's objective to shed light on the training dynamics of SNN. Unlike typical studies of convergence to global solutions of NN training dynamics, SNN presents an additional complexity due to its non-convex ambient loss function, a feature that is common in unsupervised learning settings.

Keywords:
Spectral neural networks,
Manifold learning,
Riemannian optimization,
Approximation with neural networks,
Graph Laplacian.

Article Info.:
Volume: 5
Number: 1
Pages: 19 - 70
Date: March/2026
doi.org/10.4208/jml.240322

Article History:
Received: 22/03/2024
Accepted: 31/10/2025

Communicated by:
Weijie Su

1 Introduction

In the past decades, researchers from a variety of disciplines have studied the use of spectral geometric methods to process, analyze, and learn from data. These methods have been used in supervised learning [3, 8, 64], clustering [56, 72], dimensionality reduction [7, 20], and contrastive learning [34]. While the aforementioned methods have strong theoretical foundations, their algorithmic implementations often depend on traditional eigensolvers. These eigensolvers tend to underperform in practical big data scenarios due to high computational demands and memory constraints. Moreover, they are particularly vulnerable in online settings since the introduction of new data typically necessitates a full computation from scratch.

To overcome some of the drawbacks of traditional eigensolvers, new frameworks for learning from spectral geometric information that are based on the training of neural networks have emerged. To begin discussing some of popular training strategies, consider a data set $\mathcal{X}_n = \{x_1, \ldots, x_n\}$ in $\mathbb{R}^d$ and a $n \times n$ adjacency matrix $\mathcal{A}_\mathbf{n}$ describing similarity

*`cli539@stat.wisc.edu`
†`sonthal@bc.edu`
‡Corresponding author. `garciatrillo@wisc.edu`

among points in $\mathcal{X}_n$. One could start by computing the eigendecomposition of $\mathcal{A}_n$ using traditional eigensolvers and get eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$. Then, to generalize these eigenvectors to points outside of $\mathcal{X}_n$, one can minimize the following ℓ_2 loss:

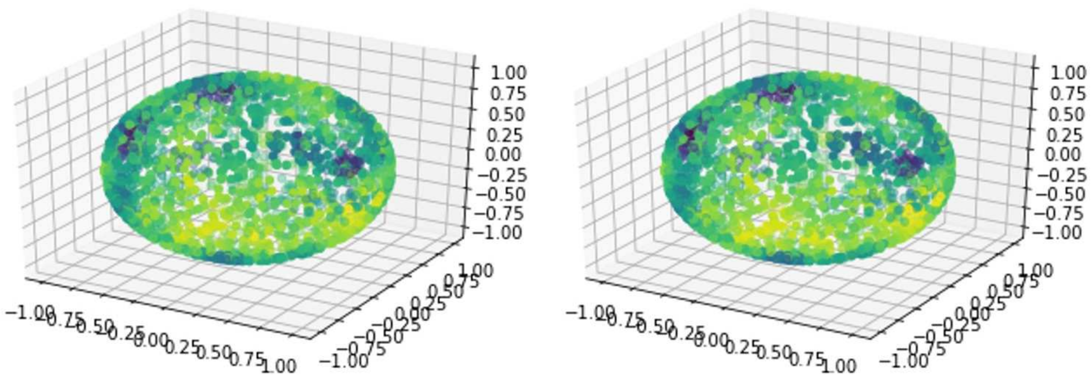
$$\min_{\theta} \|f_\theta(\mathcal{X}_n) - \mathbf{v}\|^2, \tag{1.1}$$

where $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r]$ and θ denotes the parameters of a neural network. This approach, referred to as Eigensolver net, is a natural way to extend the geometric information contained in the similarity matrix of a finite collection of points to out-of-sample data and can be used even when the matrix $\mathcal{A}_n$ is not positive semidefinite (PSD). On the other hand, the Eigensolver net has some drawbacks. Specifically, one still needs to compute the eigendecomposition using traditional eigensolvers, which is precisely what one may want to avoid.

Spectralnet [61] and spectral neural network [34] have been proposed to overcome this issue. In these approaches, the goal is to find neural networks that can approximate the spectrum of a large target matrix, and the differences among the approaches lie mostly in the specific loss functions used for training; here we focus on SNN, and provide some details on Spectralnet in Appendix A.2. A SNN is trained by minimizing the spectral contrastive loss function

$$\min_{\theta \in \Theta} L(\theta) \overset{\text{def}}{=} \ell(\mathbf{Y}_\theta), \quad \ell(\mathbf{Y}) \overset{\text{def}}{=} \left\|\mathbf{Y}\mathbf{Y}^\top - \mathcal{A}_\mathbf{n}\right\|_\mathrm{F}^2, \quad \mathbf{Y} \in \mathbb{R}^{n \times r} \tag{1.2}$$

through first-order optimization methods. Fig. 1.1 illustrates that SNNs can well approximate the desired eigenvectors associated to a proximity based similarity matrix. In the above and in the sequel, θ represents the vector of parameters of the neural network $f_\theta : \mathbb{R}^d \to \mathbb{R}^r$, the matrix $\mathbf{Y}_\theta$ is the $n \times r$ matrix whose rows are the outputs $f_\theta(x_1), \ldots, f_\theta(x_n)$, and $\|\cdot\|_\mathrm{F}$ is the Frobenius norm. The mapping f_θ can be interpreted



(A) (B)

Figure 1.1: (B) shows the first eigenvector for the Laplacian of a proximity graph from data points sampled from S^2 obtained using an eigensolver. (A) shows the same eigenvector but obtained using SNN. The difference between the two figures is minor, showing that the neural network learns the eigenvector of the graph Laplacian well. See details in Appendix B.1.