Graph Scattering Transform

Rebecca Gjini, Mason Nakamura, Sam Smith, Emily Thompson

Lehigh University, Marist College, Clemson University, Southwestern University

Abstract

In the recent history of machine learning, deep, multilayered networks have been shown to outperform traditional models. In particular, Convolutional Neural Networks (CNNs) attain state of the art performance in many machine learning tasks such as image classification. The scattering transform is a mathematical model of these CNNs which allows the use of predefined wavelet filters. We apply the scattering transform to graph-structured data motivated by classification tasks. The scattering transform produces a sequence of coefficients at each layer of the network which can be used to classify different classes of graph data. Specifically, we implement the scattering transform to classify different models of random graphs. We also reproduce and improve upon the results of Gao, Wolf, & Hirn (2019) on social network data. In addition, we explore the use of principal component analysis, applied to the scattering coefficients, as a dimensionality-reduction and visualization tool.

Background

We implement the scattering transform on graph-structured data to associate the data with a series of coefficients which will be implemented into a support vector machine for classification.

We first define the random walk matrix **P** = **AD**$^{-1}$, where **A** is the adjacency matrix and **D** is the diagonal matrix of the degree vector. We let **x** be some signal vector encoding information about the graph.

We then define the graph wavelet transform as

$$\mathbf{W_j} = \mathbf{P}^{2^{j-1}} - \mathbf{P}^{2^j}, \quad 1 \le j \le J$$

The scattering transform has a multi-layered structure, with a non-linear activation function, σ.

$$\mathbf{x} \to W_1\mathbf{x} \to \sigma(W_1\mathbf{x}) \to W_2\sigma(W_1\mathbf{x}) \to \sigma(W_2\sigma(W_1\mathbf{x}))$$

We define the scattering coefficients for $0 \le j \le J$

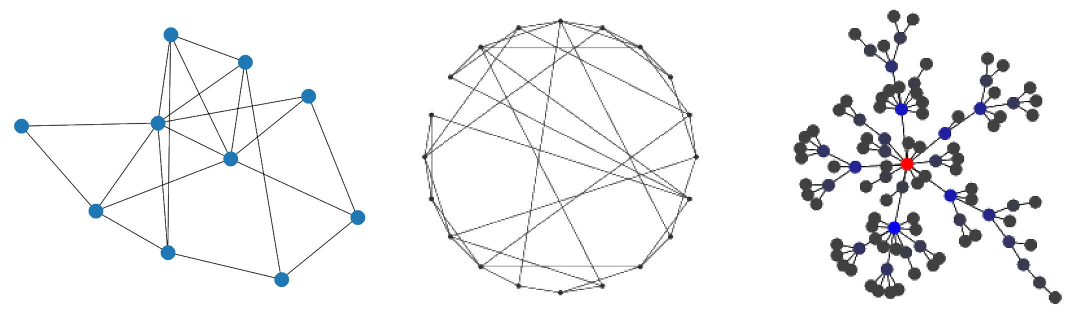
$$S[j]\mathbf{x} = \|U[j]\mathbf{x}\|_1 = \|\sigma(\mathbf{W_j}\mathbf{x})\|_1$$

We also define the second-order scattering coefficients for $0 \le j_1, j_2 \le J$

$$S[j_1, j_2]\mathbf{x} = \|U[j_1, j_2]\mathbf{x}\|_1 = \|\sigma(\mathbf{W_{j_1}}\sigma(\mathbf{W_{j_1}}\mathbf{x}))\|_1$$

Random Graphs

We first apply the scattering transform to classify three different models of random graphs: Erdős-Rényi (ER), Watts-Strogratz (WS), and Barabási-Albert (BA).



ER WS BA

We find that the scattering transform can distinguish between the geometries of the random graphs. We report the mean classification accuracies, \mu, and the standard deviations, \sigma, below.

Graph Models	μ	σ
ER vs. WS	99.5	1.98
ER vs. BA	98.7	2.56
WS vs. BA	96.3	3.68

Real-world Data

We next apply our network to a real-world social network dataset. IMDB-BINARY is a movie collaboration dataset of collected movie actor/actress and genre information. The dataset is a collection of ego networks of actors/actresses that have appeared together in any movie. The task is to classify whether these ego networks are labeled as Romance or Comedy.

We are able to reproduce and improve upon the results of Gao, Wolf, & Hirn (2019) by modifying the graph signals used. We note that choosing our graph signal as the degree vector, d, causes us to “kill off” the coefficients. That is,

$$\mathbf{\Psi_j d = 0}$$

Since the degree vector holds valuable information, we navigated this problem by instead using a modified version of **d.** In particular, we found using log(**d**) as a signal works quite well. With this modification we improved Gao, Wolf, & Hirn's (2019) results from an average accuracy μ of 71.2 to 72.6 over 50 runs.

Graph Signal	μ	σ	C	Γ	Norm	Unit V
$\mathbf{d}$	71.03	2.43	1	0.01	Yes	No
$\sqrt{\mathbf{d}}$	71.37	2.45	1	0.01	No	Yes
$log_2\mathbf{d}$	71.09	2.97	1	.01	Yes	No
$log_{10}\mathbf{d}$	72.60	2.54	10	0.001	Yes	No
$log_{100}\mathbf{d}$	72.12	2.99	10	0.001	Yes	No

Dimensionality Reduction

For both real and synthetic datasets, the scattering transform produces a high-dimensional representation of each graph. Therefore, using dimensionality reduction is beneficial for increasing the interpretability of the results. We used principal component analysis due to its use in the research conducted by Gao, Wolf, and Hirn.

Classification Accuracy with PCA using Erdős-Rényi Graphs		
Dimension	Percent Variance Retained	Accuracy
84	100	92.1
80	99.999	100
40	99.999	99
15	99.995	99
10	98.7	100
8	98.1	99.5
4	94.458	99.5
2	85.1	99.5

Results found that for Erdős-Rényi Random Graphs, PCA then SVM accuracy is greater than standard SVM accuracy for all dimensions tested, with the PCA accuracy averaging at 99.5%, and the standard SVM accuracy averaging at 92.1%. This result could be due to the large portion of the variance retained in the data, even at very low dimensions.

Classification Accuracy with PCA using the IMDB-B Dataset		
Dimension	Percent Variance Retained	Accuracy
180	100	70.6
40	99	70.7
35	98.604	70.4
30	98.149	71.2
24	97.276	68.09
8	87.97	66.8
4	74.462	62
2	53.186	60

For testing the IMDB-B dataset, each run uses the same thresholds of 99%, 90%, 80%, and 50% variance retained used in Gao, Wolf, and Hirn in order to better compare the percent variance retained at those thresholds. Those thresholds for the IMDB-BINARY dataset are 99% variance retained at 24 dimensions, 90% variance retained at 8 dimensions, 80% variance retained at 4 dimensions, and 50% variance retained at 2 dimensions. While we found similar results, the percent variance retained at the thresholds are lower than expected, with 3 out of the 4 thresholds having a lower percent variance retained.

Future Work

For random graphs, an interesting step would be to consider other random graph models other than the three we experimented with. It also should be possible to use graph scattering to synthesize new graphs from a given random graph model.

For real-world data, we hope to experiment on other datasets to see if the modified degree vector is still a valid choice of signal, or if this is a quirk of the IMDB-BINARY dataset. If it remains valid, analysis on how it works so well would be interesting.

For dimensionality reduction, experiment on larger datasets, especially those previously testing in Gao, Wolf, & Hirn (2019). One might also explore different methods of dimensionality reduction (e.g. LDA). A particularly interesting choice would be to experiment with the squeeze-fit algorithm, a method of dimensionality reduction designed to be used for classification tasks.

References

B. Feng Gao, Guy Wolf, Mathew Hirn, Geometric scattering for graph data analysis, In Proceedings of the 36th International Conference on Machine Learning, PMLR, volum 97, pages 2122-2131, 2019.

Fernando Gama, Alejandro Ribeiro, Joan Brunat Diffusion scattering transforms on graphs. International Conference on Learning Representations, 2019.

Acknowledgements

Special thanks to Michael Perlmutter, Nathan Brugnone, and Sarah McGuire for guiding us through this research! This REU is supported by the National Security Agency (NSA Award No. H98230-20-1-0006), the National Science Foundation (NSF Award No. 1852066), and Michigan State University.