Multi-Scale and Multi-Representation Learning on Graphs and Manifolds

Zhizhen Jane Zhao

Department of Electrical and Computer Engineering
University of Illinois Urbana-Champaign

Seminar on Applied Mathematics and Data Science,
Department of Mathematics, HKUST
Sep. 26, 2019
Table of contents

1. Geometric Deep Learning

2. LanczosNet

3. Unsupervised Learning on Graphs and Manifolds

4. Summary

**Collaborators:** Renjie Liao (Grad student, University of Toronto), Richard Zemel (University of Toronto), Raquel Urtasun (University of Toronto), Yifeng Fan (Grad student at UIUC), Tingran Gao (William H. Kruskal Instructor U Chicago)
Source: M. Bronstein Geometric Deep Learning SIAM 2018 Tutorial
Applications of geometric deep learning

- Recommender system
- Neutrino detection
- LHC

- Fake news detection
- Drug repurposing
- Chemistry

Source: M. Bronstein Geometric Deep Learning SIAM 2018 Tutorial
Calculus on graphs

- **Graph** $G = (V, E)$
- **Vertices** $V = \{1, \ldots, n\}$
- **Edges** $E \subseteq V \times V$
  - undirected: $(i, j) \in E$ iff $(j, i) \in E$

Vertex weights $a_i > 0$ for $i \in V$

Edge weights $w_{ij} > 0$ for $(i, j) \in E$
Calculus on graphs

- **Graph** $G = (V, E)$
- **Vertices** $V = \{1, \ldots, n\}$
- **Edges** $E \subseteq V \times V$
  undirected: $(i, j) \in E$ iff $(j, i) \in E$
- **Vertex weights** $a_i > 0$ for $i \in V$
Calculus on graphs

- **Graph** $G = (V, E)$
- **Vertices** $V = \{1, \ldots, n\}$
- **Edges** $E \subseteq V \times V$
  
  undirected: $(i, j) \in E$ iff $(j, i) \in E$
- **Vertex weights** $a_i > 0$ for $i \in V$
- **Edge weights** $w_{ij} \geq 0$ for $(i, j) \in E$
Calculus on graphs: vertex- and edge-fields

- **Vertex field** \( f : V \rightarrow \mathbb{R} \)
Calculus on graphs: vertex- and edge-fields

- **Vertex field** $f : V \to \mathbb{R}$
- **Edge field** $F : E \to \mathbb{R}$ assumed alternating $F_{ij} = -F_{ji}$
Calculus on graphs: vertex- and edge-fields

- **Vertex field** $f: V \to \mathbb{R}$
- **Edge field** $F: E \to \mathbb{R}$ assumed alternating $F_{ij} = -F_{ji}$
- **Hilbert space** with inner products
  \[
  \langle f, g \rangle_{L^2(V)} = \sum_{i \in V} a_i f_i g_i
  \]
  \[
  \langle F, G \rangle_{L^2(E)} = \sum_{i \in E} w_{ij} F_{ij} G_{ij}
  \]
Calculus on graphs: gradient and divergence

- **Gradient** operator $\nabla : L^2(V) \rightarrow L^2(E)$

$$(\nabla f)_{ij} = f_i - f_j$$
Calculus on graphs: gradient and divergence

- **Gradient** operator $\nabla : L^2(V) \rightarrow L^2(E)$

  $$(\nabla f)_{ij} = f_i - f_j$$

- **Divergence** operator

  $\text{div} : L^2(E) \rightarrow L^2(V)$

  $$(\text{div} F)_i = \frac{1}{a_i} \sum_{j: (i,j) \in E} w_{ij} F_{ij}$$
Calculus on graphs: gradient and divergence

**Gradient** operator $\nabla : L^2(V) \rightarrow L^2(E)$

$$(\nabla f)_{ij} = f_i - f_j$$

**Divergence** operator

$\text{div} : L^2(E) \rightarrow L^2(V)$

$$(\text{div} F)_i = \frac{1}{a_i} \sum_{j: (i,j) \in E} w_{ij} F_{ij}$$

adjoint to the gradient operator

$$\langle F, \nabla f \rangle_{L^2(E)} = \langle \nabla^* F, f \rangle_{L^2(V)} = \langle -\text{div} F, f \rangle_{L^2(V)}$$
Laplacian operator $L : L^2(V) \to L^2(V)$

$$(Lf)_i = \frac{1}{a_i} \sum_{j : (i,j) \in E}^{} w_{ij} (f_i - f_j)$$

difference between $f$ and its local average
Calculus on graphs: graph Laplacian

- **Laplacian** operator \( L : L^2(V) \to L^2(V) \)

\[
(Lf)_i = \frac{1}{a_i} \sum_{j : (i,j) \in E} w_{ij} (f_i - f_j)
\]

difference between \( f \) and its local average

- Represented as a **positive semi-definite** \( n \times n \) matrix

\[
L = A^{-1}(D - W)
\]
LanczosNet
Basic notations

Graph $G = (V, E, W)$, where $V$, $E$, $W$ are set of vertices, set of edges and edge weight matrix.

Graph Laplacian

- Combinatorial definition: $L = D - W$, where $D$ is degree matrix, $D_{ii} = \sum_j W_{ij}$
- Random walk normalized definition: $L = I - D^{-1}W$
- Symmetric normalized definition: $L = I - D^{-1/2}W(D^{-1/2})'$

\[
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 2 & 0 & 0 & 0 & 0 \\
2 & 0 & 2 & 0 & 0 & 0 \\
3 & 0 & 0 & 2 & 0 & 0 \\
4 & 0 & 0 & 0 & 1 & 0 \\
5 & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\quad - \quad
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 0 & 1 & 1 & 0 & 0 \\
2 & 1 & 0 & 1 & 0 & 0 \\
3 & 1 & 1 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 1 & 0 \\
5 & 0 & 0 & 0 & 1 & 0 \\
\end{array}
\quad = \quad
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 2 & -1 & -1 & 0 & 0 \\
2 & -1 & 2 & -1 & 0 & 0 \\
3 & -1 & -1 & 2 & 0 & 0 \\
4 & 0 & 0 & 0 & 1 & -1 \\
5 & 0 & 0 & 0 & -1 & 1 \\
\end{array}
\]
Graph Fourier transform (Shuman et al. 2013)

- Input signal $X \in \mathbb{R}^{n \times 1}$
- Spectral decomposition: $L = U \Lambda U^\top$
- Graph Fourier transform and its inverse:
  $$Y = U^\top X \quad \quad X = U Y$$
- Spectral filtering
  $$Y = g_\theta(L)X = Ug_\theta(\Lambda)U^\top X$$
  where $g_\theta(\Lambda)$ is the filter and $\theta$ are learnable parameters.
- Polynomial localized filter: $g_\theta(\Lambda) = \sum_{k=0}^{K} \theta_k \Lambda^k$

$$\begin{array}{ccc}
1 & 0 & 0 \\
A & B & C
\end{array}$$

$$\begin{array}{c}
X^2 = L \\ x^1 = L \\ x^0
\end{array}$$
Chebyshev networks

ChebyNet (Defferraard et al. 2016) avoids full graph Fourier transform via $K$-th order Chebyshev polynomial:

\[ y = g_{\theta}(L)X = \sum_{k=0}^{K} \theta_k T_k(\tilde{L})X \]

where \( \tilde{L} = 2L/\lambda_{\text{max}} - I \), \( \tilde{X}_0 = X \), \( \tilde{X}_1 = \tilde{L}X \) and

\[ \tilde{X}_k = T_k(\tilde{L})X = 2\tilde{L}\tilde{X}_{k-1} - \tilde{X}_{k-2} \]

Final localized filtering is,

\[ y = \theta^T \tilde{X} = [\theta_0, \theta_1, \ldots, \theta_K]^T [\tilde{X}_0, \tilde{X}_1, \ldots, \tilde{X}_K] \]

\[ x \in \mathbb{R}^{12} \quad y \in \mathbb{R}^{6} \quad z \in \mathbb{R}^{3} \]

Figure: Graph Coarsening and Pooling \(^1\)

\(^1\)Image credit: Defferrard et al. 2016.
Graph convolutional networks

- GCNs (Kipf et al, 2016) simplify ChebyNet by: (1) 1-localized, i.e., \( K=1 \); (2) \( \lambda_{\text{max}} = 2 \); (3) \( \theta = \theta_0 = -\theta_1 \).
- GCNs stack multiple simple convolution layers \(^2\):

\[
y = \text{softmax} \left( \tilde{W} \ \text{ReLU} \ (\tilde{W} X \tilde{W}_1) \ \tilde{W}_2 \right)
\]

where \( \tilde{W} = \tilde{D}^{-1/2} \tilde{W} \tilde{D}^{-1/2}, \tilde{W} = W + I, \tilde{D}_{ii} = \sum_j \tilde{W}_{i,j} \).

\(^2\)Image credit: Kipf et al. 2016
LanczosNet: Multi-scale graph convolution

- LanczosNet (Liao et al. 2019) uses Lanczos algorithm to obtain low-rank approximation of \( S = D^{-1/2} W D^{-1/2} = I - L \):

\[
S \approx QTQ^\top = QV\Lambda(QV)^\top
\]

where \( T \in \mathbb{R}^{K \times K} \) is a tridiagonal matrix
- \( Q \in \mathbb{R}^{D \times K} \) has orthonormal columns
- \( QV \in \mathbb{R}^{D \times K} \) has orthonormal columns
- \( \Lambda \in \mathbb{R}^{K \times K} \) is a diagonal matrix

(\( \Lambda, QV \), i.e., Ritz values and vectors, are approximations of eigenvalues and eigenvectors.

- m-th order polynomial localized filter can be efficiently computed:

\[
g(S^m) = QV\Lambda^m(QV)^\top
\]
LanczosNet: Multi-scale graph convolution

Learnable Multi-Scale Spectral Filter

- Neural networks based nonlinear filtering:

\[ \tilde{\lambda}_{i,j} = f_{\theta_j}([\lambda_i^{I_1}, \ldots, \lambda_i^{I_N}]) \quad \forall j = 1, \ldots, N \]

where \( \lambda_i = \Lambda_{i,i} \), \( f_{\theta_j} \) is a neural network and \( I \) is a set of \( N \) exponents.

- For example, \( I = \{10, 50\} \) allows us to leverage the information propagated for 10 and 50 steps.

- Construct the filtered eigenvalues:

\[ \tilde{\Lambda}_j = \text{diag}(\tilde{\lambda}_{1,j}, \ldots, \tilde{\lambda}_{K,j}) \quad \forall j = 1, \ldots, N \]

where \( \tilde{\Lambda}_j \) is a diagonal matrix.
LanczosNet: Multi-scale graph convolution

Graph Convolution Layer

- Short scale

\[ Y_{\text{short}} = \left[ L^{S_1} X, \ldots, L^{S_M} X \right], \]

where \( S \) is a set of \( M \) small exponents, e.g., \( S = \{1, 3\} \).

- Long scale

\[ Y_{\text{long}} = \left[ QV\tilde{\Lambda}_1(QV)^\top X, \ldots, QV\tilde{\Lambda}_N(QV)^\top X \right], \]

- Graph Convolution

\[ Y = \text{ReLU} \left( [Y_{\text{short}}, Y_{\text{long}}] XW_1 \right) \]

- Lanczos algorithm can be back-propagated to facilitate graph kernel and node embedding learning
Experiments
Semi-supervised node classification on citation networks
Semi-supervised node classification on citation networks

- **Input:** Citation graphs (nodes are documents, edges are citation links), class labels of a subset (percentage is list below) of nodes.
- **Output:** Class labels of a separate (much larger) subset of nodes.

<table>
<thead>
<tr>
<th></th>
<th>GCN-FP</th>
<th>GGNN</th>
<th>DCNN</th>
<th>ChebyNet</th>
<th>GCN</th>
<th>MPNN</th>
<th>GraphSAGE</th>
<th>GAT</th>
<th>LNet</th>
<th>AdaLNet</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cora</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Public</td>
<td>74.6 ± 0.7</td>
<td>77.6 ± 1.7</td>
<td>79.7 ± 0.8</td>
<td>78.0 ± 1.2</td>
<td>80.5 ± 0.8</td>
<td>78.0 ± 1.1</td>
<td>74.5 ± 0.8</td>
<td><strong>82.6 ± 0.7</strong></td>
<td>79.5 ± 1.8</td>
<td>80.4 ± 1.1</td>
</tr>
<tr>
<td>3%</td>
<td>71.7 ± 2.4</td>
<td>73.1 ± 2.3</td>
<td>76.7 ± 2.5</td>
<td>62.1 ± 6.7</td>
<td>74.0 ± 2.8</td>
<td>72.0 ± 4.6</td>
<td>64.2 ± 4.0</td>
<td>56.8 ± 7.9</td>
<td>76.3 ± 2.3</td>
<td><strong>77.7 ± 2.4</strong></td>
</tr>
<tr>
<td>1%</td>
<td>59.6 ± 6.5</td>
<td>60.5 ± 7.1</td>
<td>66.4 ± 8.2</td>
<td>44.2 ± 5.6</td>
<td>61.0 ± 7.2</td>
<td>56.7 ± 5.9</td>
<td>49.0 ± 5.8</td>
<td>48.6 ± 8.0</td>
<td>66.1 ± 8.2</td>
<td><strong>67.5 ± 8.7</strong></td>
</tr>
<tr>
<td>0.5%</td>
<td>50.5 ± 6.0</td>
<td>48.2 ± 5.7</td>
<td>59.0 ± 10.7</td>
<td>33.9 ± 5.0</td>
<td>52.9 ± 7.4</td>
<td>46.5 ± 7.5</td>
<td>37.5 ± 5.4</td>
<td>41.4 ± 6.9</td>
<td>58.1 ± 8.2</td>
<td><strong>60.8 ± 9.0</strong></td>
</tr>
<tr>
<td><strong>Citeseer</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Public</td>
<td>61.5 ± 0.9</td>
<td>64.6 ± 1.3</td>
<td>69.4 ± 1.3</td>
<td>70.1 ± 0.8</td>
<td>68.1 ± 1.3</td>
<td>64.0 ± 1.9</td>
<td>67.2 ± 1.0</td>
<td><strong>72.2 ± 0.9</strong></td>
<td>66.2 ± 1.9</td>
<td>68.7 ± 1.0</td>
</tr>
<tr>
<td>1%</td>
<td>54.3 ± 4.4</td>
<td>56.0 ± 3.4</td>
<td>62.2 ± 2.5</td>
<td>59.4 ± 5.4</td>
<td>58.3 ± 4.0</td>
<td>54.3 ± 3.5</td>
<td>51.0 ± 5.7</td>
<td>46.5 ± 9.3</td>
<td>61.3 ± 3.9</td>
<td><strong>63.3 ± 1.8</strong></td>
</tr>
<tr>
<td>0.5%</td>
<td>43.9 ± 4.2</td>
<td>44.3 ± 3.8</td>
<td>53.1 ± 4.4</td>
<td>45.3 ± 6.6</td>
<td>47.7 ± 4.4</td>
<td>41.8 ± 5.0</td>
<td>33.8 ± 7.0</td>
<td>38.2 ± 7.1</td>
<td>53.2 ± 4.0</td>
<td><strong>53.8 ± 4.7</strong></td>
</tr>
<tr>
<td>0.3%</td>
<td>38.4 ± 5.8</td>
<td>36.5 ± 5.1</td>
<td>44.3 ± 5.1</td>
<td>39.3 ± 4.9</td>
<td>39.2 ± 6.3</td>
<td>36.0 ± 6.1</td>
<td>25.7 ± 6.1</td>
<td>30.9 ± 6.9</td>
<td>44.4 ± 4.5</td>
<td><strong>46.7 ± 5.6</strong></td>
</tr>
<tr>
<td><strong>Pubmed</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Public</td>
<td>76.0 ± 0.7</td>
<td>75.8 ± 0.9</td>
<td>76.8 ± 0.8</td>
<td>69.8 ± 1.1</td>
<td>77.8 ± 0.7</td>
<td>75.6 ± 1.0</td>
<td>76.8 ± 0.6</td>
<td>76.7 ± 0.5</td>
<td><strong>78.3 ± 0.3</strong></td>
<td>78.1 ± 0.4</td>
</tr>
<tr>
<td>0.1%</td>
<td>70.3 ± 4.7</td>
<td>70.4 ± 4.5</td>
<td>73.1 ± 4.7</td>
<td>55.2 ± 6.8</td>
<td>73.0 ± 5.5</td>
<td>67.3 ± 4.7</td>
<td>65.4 ± 6.2</td>
<td>59.6 ± 9.5</td>
<td><strong>73.4 ± 5.1</strong></td>
<td>72.8 ± 4.6</td>
</tr>
<tr>
<td>0.05%</td>
<td>63.2 ± 4.7</td>
<td>63.3 ± 4.0</td>
<td>66.7 ± 5.3</td>
<td>48.2 ± 7.4</td>
<td>64.6 ± 7.5</td>
<td>59.6 ± 4.0</td>
<td>53.0 ± 8.0</td>
<td>50.4 ± 9.7</td>
<td><strong>68.8 ± 5.6</strong></td>
<td>66.0 ± 4.5</td>
</tr>
<tr>
<td>0.03%</td>
<td>56.2 ± 7.7</td>
<td>55.8 ± 7.7</td>
<td>60.9 ± 8.2</td>
<td>45.3 ± 4.5</td>
<td>57.9 ± 8.1</td>
<td>53.9 ± 6.9</td>
<td>45.4 ± 5.5</td>
<td>50.9 ± 8.8</td>
<td>60.4 ± 8.6</td>
<td><strong>61.0 ± 8.7</strong></td>
</tr>
</tbody>
</table>

**Table:** Test accuracy with 10 runs. The public splits in Cora, Citeseer and Pubmed contain 5.2%, 3.6% and 0.3% labeled examples respectively.
Graph convolution for quantum chemistry

DFT \rightarrow \text{Targets} \quad E, \omega_0, \ldots

\sim 10^3 \text{ seconds}

\sim 10^{-2} \text{ seconds}

Message Passing Neural Net
Graph Regression on QM8 Quantum Chemistry Dataset

- **Input:** Molecule graphs (nodes are atoms, edges are chemical bonds, and multiple types of chemical bonds exist.)
- **Output:** Electronic spectra and excited state energy

<table>
<thead>
<tr>
<th>Methods</th>
<th>Validation MAE ($\times 1.0e^{-3}$)</th>
<th>Test MAE ($\times 1.0e^{-3}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GCN-FP</td>
<td>15.06 ± 0.04</td>
<td>14.80 ± 0.09</td>
</tr>
<tr>
<td>GGNN</td>
<td>12.94 ± 0.05</td>
<td>12.67 ± 0.22</td>
</tr>
<tr>
<td>DCNN</td>
<td>10.14 ± 0.05</td>
<td>9.97 ± 0.09</td>
</tr>
<tr>
<td>ChebyNet</td>
<td>10.24 ± 0.06</td>
<td>10.07 ± 0.09</td>
</tr>
<tr>
<td>GCN</td>
<td>11.68 ± 0.09</td>
<td>11.41 ± 0.10</td>
</tr>
<tr>
<td>MPNN</td>
<td>11.16 ± 0.13</td>
<td>11.08 ± 0.11</td>
</tr>
<tr>
<td>GraphSAGE</td>
<td>13.19 ± 0.04</td>
<td>12.95 ± 0.11</td>
</tr>
<tr>
<td>GPNN</td>
<td>12.81 ± 0.80</td>
<td>12.39 ± 0.77</td>
</tr>
<tr>
<td>GAT</td>
<td>11.39 ± 0.09</td>
<td>11.02 ± 0.06</td>
</tr>
<tr>
<td>LanczosNet</td>
<td>9.65 ± 0.19</td>
<td>9.58 ± 0.14</td>
</tr>
<tr>
<td>AdaLanczosNet</td>
<td>10.10 ± 0.22</td>
<td>9.97 ± 0.20</td>
</tr>
</tbody>
</table>

**Table:** Mean absolute error (MAE) on QM8 dataset.
The code for LanczosNet is available at https://github.com/lrjconan/LanczosNetwork
Unsupervised Learning on Graphs
Cryo-electron microscopy single particle reconstruction

Nobel Prize in Chemistry 2017

Different posed samples

Micrograph

Particle Images

Initial 3D volume

Atomic model

3D density map

Refinement
Simplified image formation model for a 3D electron density map $V$ and $g \in SO(3)$:

$$I = P(g \cdot V) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{I})$$
Extremely noisy images
Extremely noisy images

Class averaging: classify images with similar viewing directions, register and average to improve their signal-to-noise ratio (SNR).
Extremely noisy images

**Class averaging**: classify images with similar viewing directions, register and average to improve their signal-to-noise ratio (SNR).

Clean  $I_i$  $I_j$  Average

**Crystallization in silico**
Large high-dimensional data sets

- **Large \( n \):**

  The number of images can be over 1 million.

- **High dimensional data:**

  The typical size of Eukaryotic ribosome is 250 – 300 Å in diameter and recent EM camera pixel spacing can be as small as 0.6 Å. Therefore, a single particle image can be about 500 × 500 pixels.

- **Crystallization in silico:** requires efficient and accurate algorithms.
Data geometry

- Assume that each projection image is centered.
- Each image $I$ corresponds to an unknown $g \in SO(3)$ describing the particle orientation.
- Represented by a $3 \times 3$ rotation matrix $R = \begin{pmatrix} R^1 & R^2 & R^3 \\ \end{pmatrix}$ with $RR^\top = R^\top R = I$ and $\det(R) = 1$.
- The projection image lies on a tangent plane to the two dimensional unit sphere $S^2$ at the viewing angle $v = v(R) = R^3$. 
Results

noisy

closest match

denoised
Results

- Crucial step: correctly identify nearest neighbors.
Multi-Frequency Vector Diffusion Maps
Geometry revisited

Geometry of cryo-electron microscopy single particle images:

Nonlinear dimensionality reduction:
- Locally linear embedding (LLE), ISOMAP, Hessian LLE, Laplacian eigenmaps, Diffusion maps (DM).
- Vector diffusion maps (VDM) generalizes diffusion maps (DM) to define heat kernels for vector fields on the manifold.
\( \mathcal{G} \)-invariant distances

- Given a dataset \( x_i \in \mathbb{R}^l \) for \( i = 1, \ldots, n \):

  \( \mathcal{G} \)-invariant distance: \( d_{ij} = \min_{g \in \mathcal{G}} \| x_i - g \cdot x_j \| \),
  
  optimal alignment: \( g_{ij} = \arg \min_{g \in \mathcal{G}} \| x_i - g \cdot x_j \| \).

- Data points lie on or close to a low-dimensional manifold \( \mathcal{M} \) and we define \( \mathcal{B} = \mathcal{M} / \mathcal{G} \).

- Define **neighborhood graph based on the invariant distance**: \( \mathcal{G} = (V, E) \) by \((i, j) \in E \iff d_{ij} \leq \epsilon\), with the associated alignment \( g_{ij} \in \mathcal{G} \).

- In cryo-EM single particle images example, \( \mathcal{G} = \text{SO}(2) \), which is the in-plane rotation within each image.
Multi-frequency vector diffusion maps

- **Challenge**: Noisy data induces inaccurate low-dimensional embedding.

- **Goal**: Robustly learn the nonlinear geometrical structure of data from noisy measurements to improve nearest neighbor search and alignment.

- **Our work**: Multi-frequency vector diffusion maps (MFVDM).
  1. Extend VDM by using *multiple irreducible representation*.
  2. Achieve more accurate nearest neighbor identification and alignment.

![Diagram showing clean and noisy data with added and removed edges.](image)
Laplacian eigenmap and diffusion maps

- Symmetric $n \times n$ matrix $W_0$:

$$W_0(i, j) = \begin{cases} w_{ij} & (i, j) \in E \\ 0 & (i, j) \notin E \end{cases}$$

- Diagonal degree matrix $D_0$:

$$D_0(i, i) = \text{deg}(i) = \sum_{j : (i, j) \in E} w_{ij}.$$

- Graph Laplacian, Normalized graph Laplacian and random walk matrix:

$$L_0 = D_0 - W_0, \quad \mathcal{L}_0 = I - D_0^{-1/2} W_0 D_0^{-1/2}, \quad A_0 = D_0^{-1} W_0$$

- The diffusion map $\Phi_t$ is defined in terms of the eigenvectors of $A_0$:

$$A_0 \phi_l = \lambda_l \phi_l, \quad l = 1, \ldots, n$$

$$\Phi_t : i \mapsto (\lambda_t^l \phi_l(i))_{l=1}^n$$
Multi-frequency vector diffusion maps

- **Intuition**: For neighbor points in $B$, the alignments should have **cycle consistency across multiple irreducible representations**, e.g., for neighbor nodes $i, j$ and $l$, for each $k \in \mathbb{Z}$,

$$k(\alpha_{ij} + \alpha_{jl} + \alpha_{li}) \approx 0 \mod 2\pi.$$

- In the VDM framework, we define the affinity between $i$ and $j$ by considering all paths of length $t$ connecting them, but instead of just summing the weights of all paths, we sum the transformations.

- Every path from $j$ to $i$ may result in a different transformation (like parallel transport due to curvature).
MFVDM builds a series of weight matrices $W_k$ for $k = 1, \ldots, k_{\text{max}}$:

$$W_k(i,j) = \begin{cases} w_{ij} \rho_k(g_{ij}) & (i,j) \in E, \\ 0 & \text{otherwise}, \end{cases}$$

The dimension of the irreducible representation of $\rho_k$ is $d_k$.

Degree matrix $D_k(i,i) = \sum_{j:(i,j) \in E} w_{ij} I_{d_k \times d_k}$.

In the application in cryo-EM image analysis, $\rho_k(g) = e^{ik\alpha}$ and $d_k = 1$ for all $k$. 
Averaging operator for vector fields

- The Hilbert space $\mathcal{H}$, as a unitary representation of the compact Lie group $G$, admits an isotypic decomposition $\mathcal{H} = \bigoplus \mathcal{H}_k$, where a function $f$ is in $\mathcal{H}_k$ if and only if $f(xg) = g^k f(x)$.

- For each frequency $k$, we construct a normalized matrix $A_k = D_k^{-1} W_k$, which is an averaging operator for vector fields in $\mathcal{H}_k$.

\[
(A_k z_k)(i) = \frac{1}{\text{deg}(i)} \sum_{j: (i,j) \in E} w_{ij} \rho_k(g_{ij}) z_k(j).
\]
Averaging operator for vector fields

- At each frequency $k$, the affinity between $i$ and $j$ is defined as the consistency between these transformations.

- $A_k = D_k^{-1} W_k$ is similar to the Hermitian matrix

$$\tilde{A}_k = D_k^{-1/2} W_k D_k^{-1/2}$$

- We define the affinity between $i$ and $j$ as

$$\left\| \tilde{A}_k^{2t}(i,j) \right\|_{HS}^2 = \frac{\deg(i)}{\deg(j)} \left\| (D_k^{-1} W_k)^{2t}(i,j) \right\|_{HS}^2.$$
VDM at frequency $k$

- Define the affinity matrix $\tilde{A}_k$ for frequency $k$:

\[
\tilde{A}_k = \sum_{l=1}^{nd_k} \lambda_l^{(k)} u_l^{(k)}(i) \overline{u_l^{(k)}(j)}, \quad \tilde{A}^{2t}_k = \sum_{l=1}^{nd_k} \left(\lambda_l^{(k)}\right)^{2t} u_l^{(k)}(i) \overline{u_l^{(k)}(j)}
\]

with $|\lambda_1^{(k)}| \geq |\lambda_2^{(k)}| \geq \ldots \geq |\lambda_{nd_k}^{(k)}|$.

- The affinity between $i$ and $j$ is given as $\|\tilde{A}^{2t}_k(i, j)\|_{HS}^2$.

- VDM mapping for frequency $k$:

\[
\hat{V}_t^{(k)} : i \mapsto \left(\left(\lambda_l^{(k)} \lambda_r^{(k)}\right)^{t} \langle u_l^{(k)}(i), u_r^{(k)}(i)\rangle\right)_{l, r=1}^{m_k}
\]

We call this frequency-$k$-VDM, $m_k \ll nd_k$ is a truncation parameter.
Group Equivariant Property of Eigenvectors

- The eigenvectors of $\tilde{A}_k$ are group equivariant: $u_l^{(k)}(R_\alpha \cdot l_1) = u_l^{(k)}(i)e^{-ik\alpha}$.

- For images of the same views $v_i = v_j$, the corresponding entries of eigenvalues are vectors in the complex plane and,

$$u_l^{(k)}(i) = e^{ik\alpha ij}u_l^{(k)}(j), \quad \forall l = 1, \ldots, n.$$ 

- To estimate the in-plane rotational alignment angles for images of similar views, we

$$\hat{\alpha}_{ij} = \arg \max_{\alpha} \sum_{k=1}^{k_{\text{max}}} \sum_{l=1}^{m} \left(\lambda_l^{(k)}\right)^2 u_l^{(k)}(i)u_l^{(k)}(j)e^{-ik\alpha}.$$ 

- Efficiently estimated using FFT.
Multi-frequency vector diffusion maps

- **Multi-frequency vector diffusion maps**: Concatenate $\hat{V}_t^{(k)}$ for $k = 1, \ldots, k_{\text{max}}$:

$$
\hat{V}_t(i) : i \mapsto (\hat{V}_t^{(1)}(i); \hat{V}_t^{(2)}(i); \ldots; \hat{V}_t^{(k_{\text{max}})}(i)).
$$

- **Multi-frequency vector diffusion distance**:

$$
d_{\text{MFVDM}, t}^2(i, j) = \left\| \frac{\hat{V}_t(i)}{\|\hat{V}_t(i)\|} - \frac{\hat{V}_t(j)}{\|\hat{V}_t(j)\|} \right\|_2^2.
$$

- Using multiple irreducible representation leads to a **highly robust measure of neighbor points on $B$**.

![Diagram](image)
Multi-Frequancy Class Averaging: Spectral Properties

- Related to the application in cryo-EM image analysis, we assume that the data points $x_i$ are uniformly distributed over $SO(3)$ according to the Haar measure.

- The base manifold characterized by the viewing directions $v_i$’s is a unit two sphere $S^2$ and the pairwise alignment group is $SO(2)$.

- Then $e^{i k \alpha_{ij}}$ approximates the local parallel transport operator from $T_{v_j}S^2$ to $T_{v_i}S^2$, whenever $x_i$ and $x_j$ have similar viewing directions $v_i$ and $v_j$ that satisfy $\langle v_i, v_j \rangle \geq 1 - h$.

- The matrices $W_k^{\text{clean}}$ approximate the local parallel transport operators $T_h^{(k)}$, which are integral operators over $SO(3)$.
Theorem (Gao, Fan, Z. 2019, Eigenvalues of $T_h^{(k)}$)

The operator $T_h^{(k)}$ has a discrete spectrum $\lambda_n^{(k)}(h)$ for all $n \in \mathbb{N}$, and $\lambda_n^{(k)} = 0$ for all $0 \leq n < |k|$. For $n \geq |k|$ and $h \in (0, 2]$, the dimension of the eigenspace of $T_h^{(k)}$ corresponding to $\lambda_n^{(k)}$ is $2n + 1$. More precisely, in the regime $h \ll 1$, the eigenvalue $\lambda_n^{(k)}(h)$ ($n \geq |k|$) adopts asymptotic expansion

$$\lambda_n^{(k)}(h) = \frac{1}{2} h - \frac{1}{8} (n^2 + n - k^2) + O(h^3).$$

Moreover, we show that $\lambda_n^{(k)}(h)$ is a polynomial in $h$ of degree $(n + 1)$ whenever $n \geq |k|$. 

Spectral Properties
Examples with $k = 1$ and $k = 2$

The largest three eigenvalues for cases $k = 1$ and $k = 2$ can be explicitly written out as

\[
\begin{align*}
\lambda_1^{(1)}(h) &= \frac{1}{2} h - \frac{1}{8} h^2, \\
\lambda_2^{(1)}(h) &= \frac{1}{2} h - \frac{5}{8} h^2 + \frac{1}{6} h^3, \\
\lambda_3^{(1)}(h) &= \frac{1}{2} h - \frac{11}{8} h^2 + \frac{25}{24} h^3 - \frac{15}{64} h^4, \\
\lambda_2^{(2)}(h) &= \frac{1}{2} h - \frac{1}{4} h^2 + \frac{1}{24} h^3, \\
\lambda_3^{(2)}(h) &= \frac{1}{2} h - h^2 + \frac{13}{24} h^3 - \frac{3}{32} h^4, \\
\lambda_4^{(2)}(h) &= \frac{1}{2} h - 2 h^2 + \frac{57}{24} h^3 - \frac{70}{64} h^4 + \frac{7}{40} h^5.
\end{align*}
\]
Spectral Gap

We have the following characterization of the spectral gap for $T^{(k)}_h$ in the regime $0 < h \ll 1$ with $\Delta_k := \frac{1}{k+1}$.

**Theorem (Gao, Fan, Z. 2019, Spectral Gap)**

For every value of $h \in (0, 2]$, the largest eigenvalue of $T^{(k)}_h$ is $\lambda^{(k)}_k(h)$. In addition, for every value of $h \in (0, \Delta_k]$, the spectral gap $G^{(k)}(h)$ between the largest and the second largest eigenvalue of $T^{(k)}_h$ is

$$G^{(k)}(h) = \frac{2^{k+2} - (2 - h)^{k+1} ((k + 1)h + 2)}{2^{k+1}(k + 2)}.$$
We have the following characterization of the spectral gap for $T_h^{(k)}$ in the regime $0 < h \ll 1$ with $\Delta_k := \frac{1}{k+1}$.

**Theorem (Gao, Fan, Z. 2019, Spectral Gap)**

For every value of $h \in (0, 2]$, the largest eigenvalue of $T_h^{(k)}$ is $\lambda_k^{(k)}(h)$. In addition, for every value of $h \in (0, \Delta_k]$, the spectral gap $G^{(k)}(h)$ between the largest and the second largest eigenvalue of $T_h^{(k)}$ is

$$G^{(k)}(h) = \frac{2^{k+2} - (2 - h)^{k+1} ((k + 1)h + 2)}{2^{k+1}(k + 2)}.$$

- When $h \ll 1$, the top spectral gap is $G^{(k)}(h) \approx \frac{1+k}{4} h^2$, which increases with the angular frequency.

---

Noise Model—Random Rewiring

- The ground truth local parallel transport data is computed by aligning the local frames within the connected neighborhood \( \langle v_i, v_j \rangle > 1 - h \), determined by the entries of the matrix \( R_i^{-1} R_j \).

- The clean graph is then perturbed following the **random rewiring model**:

\[
(i, j) \in E = \begin{cases} 
(i, j) & \text{with probability } p \\
(i, j) \rightarrow (i, l), \alpha_{il} \sim \text{Unif}[0, 2\pi) & \text{with probability } 1 - p
\end{cases}
\]
Numerical Experiments

$p = 1$

$k_c = 1$

$p = 0.4$

$k_c = 4$

$p = 0.2$

$k_c = 7$
Nearest neighbor identification & rotational alignment

- Histograms of nearest neighbor identification accuracy (The histogram with more points close to 0 is better) and rotational alignment errors.
- MFVDM is very robust to noise.
Unsupervised Learning on $G$-Manifold
The $\mathcal{G}$-Manifold and fibre bundles

In geometric terms, on top of a differentiable manifold $\mathcal{M}$ underlying the dataset of interest, the $\mathcal{G}$-manifold admits a smooth *right action* of a Lie group $\mathcal{G}$, in the sense that there is a smooth map $\phi : \mathcal{G} \times \mathcal{M} \to \mathcal{M}$ satisfying $\phi(e, m) = m$ and $\phi(g_2, \phi(g_1, m)) = \phi(g_1g_2, m)$ for all $m \in \mathcal{M}$ and $g_1, g_2 \in \mathcal{G}$. Each of these fibre bundles provide a separate "view" towards unveiling the geometry of the common base manifold on which all the fibre bundles reside.
The $\mathcal{G}$-Manifold and fibre bundles

In geometric terms, on top of a differentiable manifold $\mathcal{M}$ underlying the dataset of interest, the $\mathcal{G}$-manifold admits a smooth right action of a Lie group $\mathcal{G}$, in the sense that there is a smooth map $\phi: \mathcal{G} \times \mathcal{M} \to \mathcal{M}$ satisfying $\phi(e, m) = m$ and $\phi(g_2, \phi(g_1, m)) = \phi(g_1g_2, m)$ for all $m \in \mathcal{M}$ and $g_1, g_2 \in \mathcal{G}$.

A $\mathcal{G}$-manifold admitting a principal bundle structure is naturally associated with as many vector bundles as the number of distinct irreducible representations of the transformation group $\mathcal{G}$. 
The $G$-Manifold and fibre bundles

- In geometric terms, on top of a differentiable manifold $\mathcal{M}$ underlying the dataset of interest, the $G$-manifold admits a smooth *right action* of a Lie group $G$, in the sense that there is a smooth map $\phi : G \times \mathcal{M} \to \mathcal{M}$ satisfying $\phi(e, m) = m$ and $\phi(g_2, \phi(g_1, m)) = \phi(g_1g_2, m)$ for all $m \in \mathcal{M}$ and $g_1, g_2 \in G$.

- A $G$-manifold admitting a principal bundle structure is naturally associated with as many vector bundles as the number of distinct irreducible representations of the transformation group $G$.

- Each of these vector bundles provide a separate “view” towards unveiling the geometry of the common base manifold on which all the fibre bundles reside.
Within each graph of a single irrep the cycle consistency of the group transformation holds $\rho_k(g_{js})\rho_k(g_{si})\rho_k(g_{ij}) \approx I_{d_k \times d_k}$.

The irreps should be consistent algebraically along the orange lines connecting the blue dots representing transformations on the edges.

Our proposed paradigm exploits all such consistencies.
Weight matrix normalization and filtering

- With \( \{\lambda^{(k)}_l, u^{(k)}_l\}_{l=1}^{m_kd_k} \) of \( \tilde{A}_k \), we define a \( \mathcal{G} \)-equivariant embedding,

\[
\psi^{(k)}_t : i \mapsto \left[ \eta_2 t(\lambda_1)^{1/2} u^{(k)}_1(i), \ldots, \eta_2 t(\lambda m_k d_k)^{1/2} u^{(k)}_{m_k d_k}(i) \right].
\]

- Denoise \( \tilde{A}_k \) by spectral filter \( \tilde{W}_{k,t} = \eta_2 t(\tilde{A}_k) \). For example, \( \eta_2 t(\lambda) = \lambda^{2t} \), or \( \eta_2 t(\lambda) = (2 \lambda - \lambda^2)^{2t} \).

- Optimal alignment affinity measure:

\[
S^{OA}_t(i,j) = \max_{g \in \mathcal{G}} \frac{1}{k_{\max}} \left| \sum_{k=1}^{k_{\max}} \text{Tr} \left[ \tilde{W}_{k,t}(i,j) \rho_k(g) \right] \right|,
\]
Unlike normal moments, the lower order moments of the power spectrum are not informative to the higher order image, e.g. $M_4$ and $M_6$. Finding the pairwise optimal alignment is challenging and time consuming.
Invariant moments affinity: power spectrum

- Finding the pairwise optimal alignment is challenging and time consuming.
- Use group invariant features.
Finding the pairwise optimal alignment is challenging and time consuming.

Use group invariant features.

For 1D periodic signal, the power spectrum is translational invariant.
Invariant moments affinity: power spectrum

We can extend this to any compact Lie group according to Peter-Weyl.

\[ S_{t}^{\text{power spec}}(i, j) = \frac{1}{k_{\text{max}}} \left| \sum_{k=1}^{k_{\text{max}}} \text{Tr} [P_{k,t}(i, j)] \right|, \text{ with} \]

\[ P_{k,t}(i, j) = \widetilde{W}_{k,t}(i, j)\widetilde{W}_{k,t}(i, j)^* . \]
Invariant moments affinity: power spectrum

- We can extend this to any compact Lie group according to Peter-Weyl.

\[ S_t^{\text{power spec}}(i,j) = \frac{1}{k_{\text{max}}} \left| \sum_{k=1}^{k_{\text{max}}} \text{Tr} [P_{k,t}(i,j)] \right|, \text{ with} \]

\[ P_{k,t}(i,j) = \tilde{W}_{k,t}(i,j) \tilde{W}_{k,t}(i,j)^*. \]

- Related to the multi-frequency vector diffusion maps: the similarity can be computed from the inner product of MFVDM embedding.
Invariant moments affinity: power spectrum

- We can extend this to any compact Lie group according to Peter-Weyl.

\[ S_{t}^{\text{power spec}}(i,j) = \frac{1}{k_{\max}} \left| \sum_{k=1}^{k_{\max}} \text{Tr} [P_{k,t}(i,j)] \right|, \text{ with} \]

\[ P_{k,t}(i,j) = \tilde{W}_{k,t}(i,j) \tilde{W}_{k,t}(i,j)^{*}. \]

- Related to the multi-frequency vector diffusion maps: the similarity can be computed from the inner product of MFVDM embedding.

- Shortcoming: It does not couple information at different frequency channels and loses the relative phase information.
Translational invariance: Bispectrum

- Bispectrum for 1D periodic signal $f$

$$ b_f(k_1, k_2) = \hat{f}(k_1)\hat{f}(k_2)\hat{f}(-(k_1 + k_2)) $$
Translational invariance: Bispectrum

- Bispectrum for 1D periodic signal $f$

$$b_f(k_1, k_2) = \hat{f}(k_1)\hat{f}(k_2)\hat{f}(-(k_1 + k_2))$$

- Bispectrum is shift invariant, complete, and unbiased.
Translational invariance: Bispectrum

- Bispectrum for 1D periodic signal $f$

$$b_f(k_1, k_2) = \hat{f}(k_1)\hat{f}(k_2)\hat{f}(-(k_1 + k_2))$$

- Bispectrum is shift invariant, complete, and unbiased.

- Phase information is preserved (unlike power spectrum)
Translational invariance: Bispectrum

- Bispectrum for 1D periodic signal $f$

$$b_f(k_1, k_2) = \hat{f}(k_1)\hat{f}(k_2)\hat{f}(-(k_1 + k_2))$$

- Bispectrum is shift invariant, complete, and unbiased.

- Phase information is preserved (unlike power spectrum)

Bispectrum for compact Lie group

Consider two unitary irreducible representations on vector spaces $\mathcal{H}_{k_1}$ and $\mathcal{H}_{k_2}$ of $G$. 

---

Zhizhen Jane Zhao (UIUC)  
Learning on Graphs and Manifolds  
HKUST 2019  
56 / 65
Bispectrum for compact Lie group

- Consider two unitary irreducible representations on vector spaces $\mathcal{H}_{k_1}$ and $\mathcal{H}_{k_2}$ of $G$.
- There exists $G$-equivariant maps from $\mathcal{H}_{k_1} \otimes \mathcal{H}_{k_2} \rightarrow \bigoplus \mathcal{H}_k$, called generalized Clebsch–Gordan coefficients $C_{k_1,k_2}$ for compact Lie group $G$, which satisfies

$$\rho_{k_1}(g) \otimes \rho_{k_2}(g) = C_{k_1,k_2} \left[ \bigoplus_{k \in k_1 \otimes k_2} \rho_k(g) \right] C_{k_1,k_2}^*.$$
Bispectrum for compact Lie group

- Consider two unitary irreducible representations on vector spaces $\mathcal{H}_{k_1}$ and $\mathcal{H}_{k_2}$ of $\mathcal{G}$.
- There exists $\mathcal{G}$-equivariant maps from $\mathcal{H}_{k_1} \otimes \mathcal{H}_{k_2} \rightarrow \bigoplus \mathcal{H}_k$, called generalized Clebsch–Gordan coefficients $C_{k_1,k_2}$ for compact Lie group $\mathcal{G}$, which satisfies

\[
\rho_{k_1}(g) \otimes \rho_{k_2}(g) = C_{k_1,k_2} \left[ \bigoplus_{k \in k_1 \otimes k_2} \rho_k(g) \right] C_{k_1,k_2}^*.
\]

- Using the fact that $C_{k_1,k_2}$ and $\rho_k$'s are unitary matrices, we have

\[
\left[ \rho_{k_1}(g) \otimes \rho_{k_2}(g) \right] C_{k_1,k_2} \left[ \bigoplus_{k \in k_1 \otimes k_2} \rho_k^*(g) \right] C_{k_1,k_2}^* = I_{d_{k_1} \times d_{k_2} \times d_{k_1} d_{k_2}}.
\]
Bispectrum for compact Lie group

- The bispectral $G$-invariant affinity:

$$S_{t}^{\text{bispec}}(i,j) = \frac{1}{(k_{\text{max}})^2} \left| \sum_{k_1=1}^{k_{\text{max}}} \sum_{k_2=1}^{k_{\text{max}}} \text{Tr} [B_{k_1,k_2,t}(i,j)] \right|,$$

with

$$B_{k_1,k_2,t}(i,j) = \left[ \tilde{W}_{k_1,t}(i,j) \otimes \tilde{W}_{k_2,t}(i,j) \right] C_{k_1,k_2} \left[ \bigoplus_{k_1} \otimes k_2 \tilde{W}_{k,t}(i,j) \right] C_{k_1,k_2}^*.$$
The bispectral $G$-invariant affinity:

$$S_{t}^{\text{bispec}}(i,j) = \frac{1}{(k_{\max})^2} \left| \sum_{k_1=1}^{k_{\max}} \sum_{k_2=1}^{k_{\max}} \text{Tr} \left[ B_{k_1,k_2,t}(i,j) \right] \right| , \text{ with}$$

$$B_{k_1,k_2,t}(i,j) = \left[ \tilde{W}_{k_1,t}(i,j) \otimes \tilde{W}_{k_2,t}(i,j) \right] C_{k_1,k_2} \left[ \bigoplus_{k \in k_1} \otimes_k \tilde{W}_{k,t}(i,j) \right] C_{k_1,k_2}^*.$$

If the transformations are consistent across different $k$’s, then the trace of $B_{k_1,k_2,t}$ should be large.
Bispectrum for compact Lie group

- The bispectral $G$-invariant affinity:

\[
S_{t}^{\text{bispec}}(i,j) = \frac{1}{(k_{\text{max}})^2} \left| \sum_{k_1=1}^{k_{\text{max}}} \sum_{k_2=1}^{k_{\text{max}}} \text{Tr} \left[ B_{k_1,k_2,t}(i,j) \right] \right| , \text{ with }
\]

\[
B_{k_1,k_2,t}(i,j) = \left[ \tilde{W}_{k_1,t}(i,j) \otimes \tilde{W}_{k_2,t}(i,j) \right] C_{k_1,k_2} \left[ \bigoplus_{k \in k_1} \bigotimes_{k_2} \tilde{W}_{k,t}(i,j) \right] C_{k_1,k_2}^*.
\]

- If the transformations are consistent across different $k$’s, then the trace of $B_{k_1,k_2,t}$ should be large.

- Take into account the consistency of the transformation at each frequency and also enforces the algebraic consistency across different irreps.
Higher-order moments

- Design higher order invariant features to define pairwise affinity?

- The order-$d + 1$ $G$-invariant features,
  \[
  M_{k_1,\ldots,k_d} = [F_{k_1} \otimes \cdots \otimes F_{k_d}] C_{k_1,\ldots,k_d} \left[ \bigoplus_{k \in k_1 \otimes \cdots \otimes k_d} F_k^* \right] C_{k_1,\ldots,k_d}^*.
  \]
Higher-order moments

- Design higher order invariant features to define pairwise affinity?

- The order-$d + 1$ $G$-invariant features,
  \[ M_{k_1,\ldots,k_d} = [F_{k_1} \otimes \cdots \otimes F_{k_d}] \ C_{k_1,\ldots,k_d} \left[ \bigoplus_{k \in k_1} \otimes \cdots \otimes k_d \ F^*_k \right] \ C^*_{k_1,\ldots,k_d}. \]

- The computational complexity of computing the higher-order moments grows exponentially with the order $d$.

- The bispectrum is sufficient to enforce the consistency of the group transformations between nodes and across all irreps.
Example with $\mathcal{G} = \text{SO}(2)$

- The unitary irreps of the group are $\rho_k(\alpha) = e^{i k \alpha}$, where $i = \sqrt{-1}$.

- The dimensions of the irreps are $d_k = 1$, and $k_1 \otimes k_2 = k_1 + k_2$.

- The generalized Clebsch–Gordan coefficients is 1 for all $(k_1, k_2)$ pairs.

- For the optimal alignment affinity, we can use length $N$ zero-padded FFT to efficiently find approximate solution, therefore the computational complexity for evaluating $S_t^{OA}(i, j)$ is $O(N \log N)$. 
Example with $G = \text{SO}(3)$

- The unitary irreps are the Wigner $D$-matrices $D_k(\omega)$ for $\omega \in \text{SO}(3)$.

- The dimensions of $D_k$ are $d_k = 2k + 1$, and $k_1 \bigotimes k_2 = \{|k_1 - k_2|, \ldots, k_1 + k_2\}$.

- The Clebsch–Gordan coefficients for all $(k_1, k_2)$ pairs can be numerically precomputed.

- The optimal alignment affinity can be efficiently approximated using the FFTs on rotation group.
Numerical results

$p = 0.5$
- Power spec. (ours)
- Bispec. (ours)
- Opt (ours)
- VDM

$p = 0.1$
- Power spec. (ours)
- Bispec. (ours)
- Opt (ours)
- VDM

$p = 0.09$
- Power spec. (ours)
- Bispec. (ours)
- Opt (ours)
- VDM

$p = 0.08$
- Power spec. (ours)
- Bispec. (ours)
- Opt (ours)
- VDM
Why using multiple irreducible representations?

- Gain of incorporating multiple representations over the “best” representation?

In any representation, the observations from real data always contain certain level of noise, even for the “best” representation. Incorporating multiple representations allows us to leverage the inherent consistency across different representations of the same information to better remove noise (e.g. multi-frequency phase synchronization).

Methodologically, incorporating multiple representations creates a “redundant” representation akin to redundant wavelets / frames / dictionaries in applied harmonic analysis, which are known to be more robust to noise due to the additional structural rigidity.

*Gao et al. ICML, 2019*

Zhizhen Jane Zhao  
HKUST 2019 62 / 65
Why using multiple irreducible representations?

- Gain of incorporating multiple representations over the “best” representation?

- In any representation, the observations from real data always contain certain level of noise, even for the “best” representation.

*Gao et al. ICML, 2019*
Why using multiple irreducible representations?

- Gain of incorporating multiple representations over the “best” representation?

- In any representation, the observations from real data always contain certain level of noise, even for the “best” representation.

- Incorporating multiple representations allows us to leverage the inherent consistency across different representations of the same information to better remove noise (e.g. multi-frequency phase synchronization*).

*Gao et al. ICML, 2019
Why using multiple irreducible representations?

- Gain of incorporating multiple representations over the “best” representation?

- In any representation, the observations from real data always contain certain level of noise, even for the “best” representation.

- Incorporating multiple representations allows us to leverage the inherent consistency across different representations of the same information to better remove noise (e.g. multi-frequency phase synchronization*).

- Methodologically, incorporating multiple representations creates a “redundant” representation akin to redundant wavelets / frames / dictionaries in applied harmonic analysis, which are known to be more robust to noise due to the additional structural rigidity.

---

*Gao et al. *ICML*, 2019
Summary

- Incorporate numerical schemes in graph neural networks for efficient multiscale analysis of graph structured data.

- Establish a new unsupervised co-learning paradigm on \( G \)-manifold using both the local cycle consistency of group transformations on the manifold (graph) and the algebraic consistency of the unitary irreducible representations of the transformations.

- Introduce the affinity based on invariant moments in order to bypass the computationally intensive pairwise optimal alignment search and efficiently learn the underlying local neighborhood structure.

- Improve the estimation of the underlying clean data manifold.
References

Thank You!