Deep learning on graphs: Convolutional Neural Networks on Irregular Domains

Sarah Parisot

14/09/2017

Convolutional Neural Networks

- Exploit the underlying structure of the data by learning filters applied to local patches of the input image
  - Translational invariance (weight sharing)
  - Hierarchical representation
- Substantially reduces the complexity of the model with respect to MLPs
- Highly successful for many analysis tasks on euclidean data (image, speech , natural language processing…)

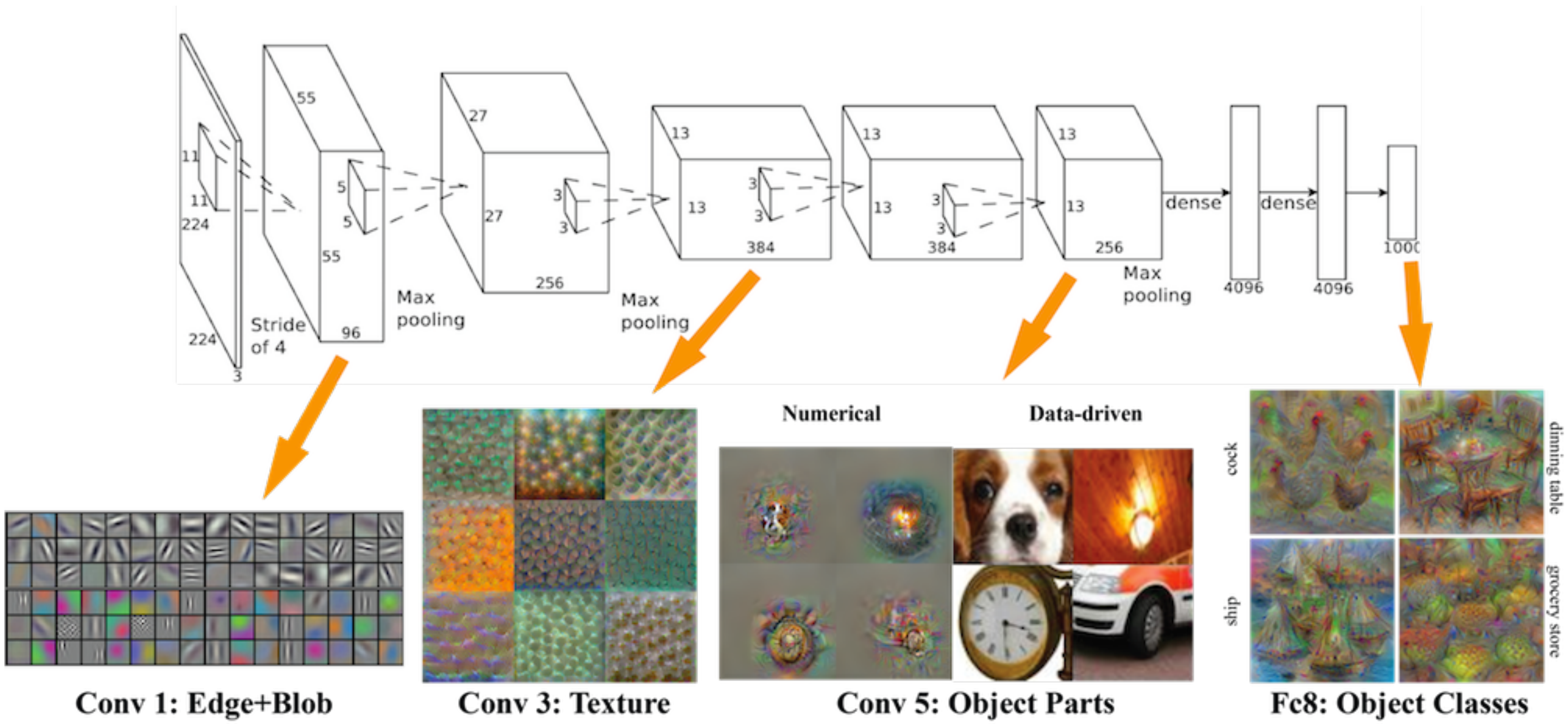
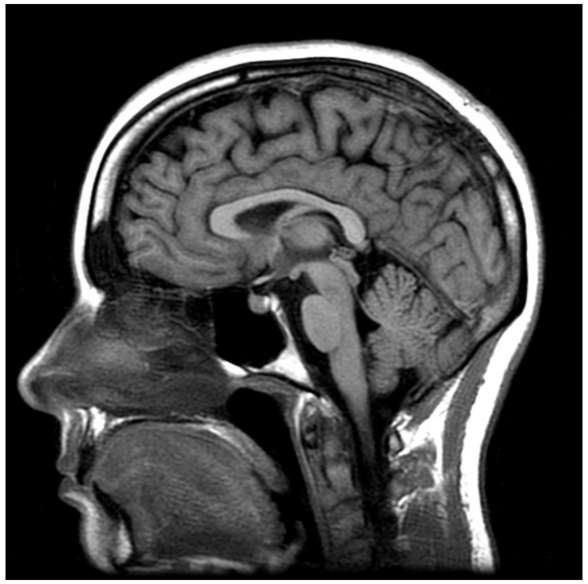
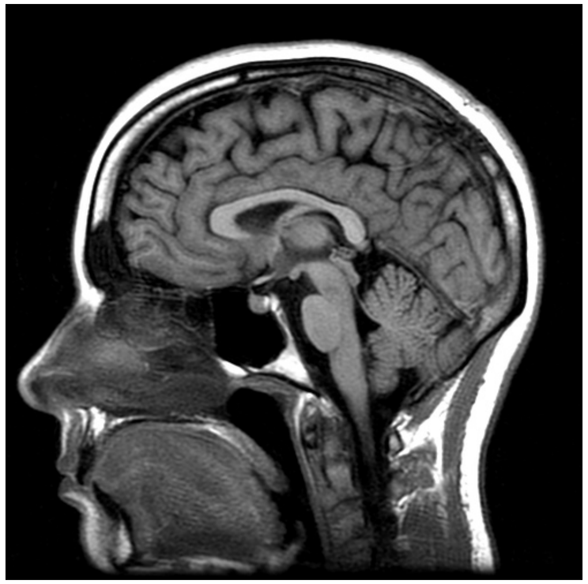


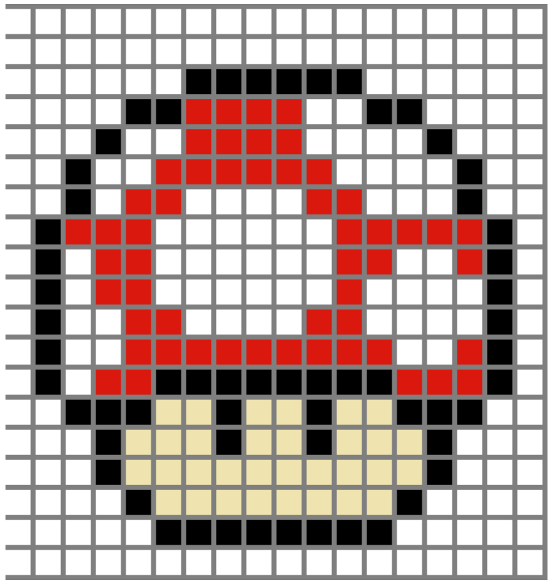
Image source: http://www.computervisionblog.com/

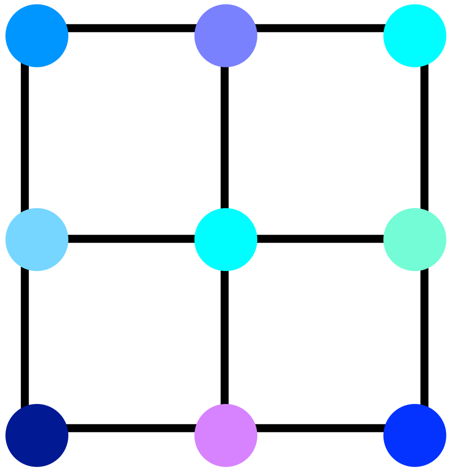
What about irregular domains?



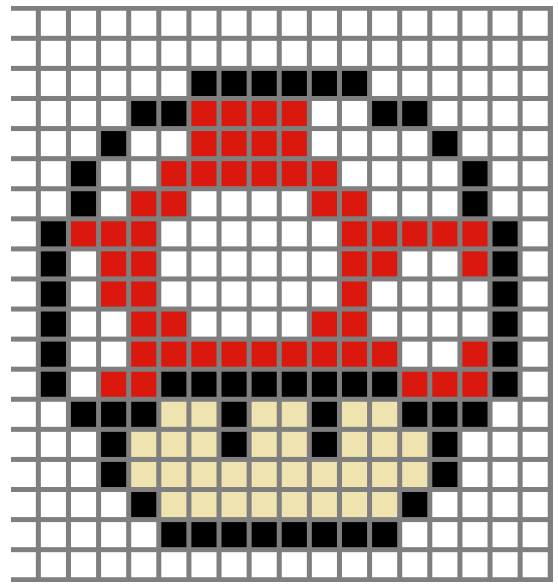
What about irregular domains?

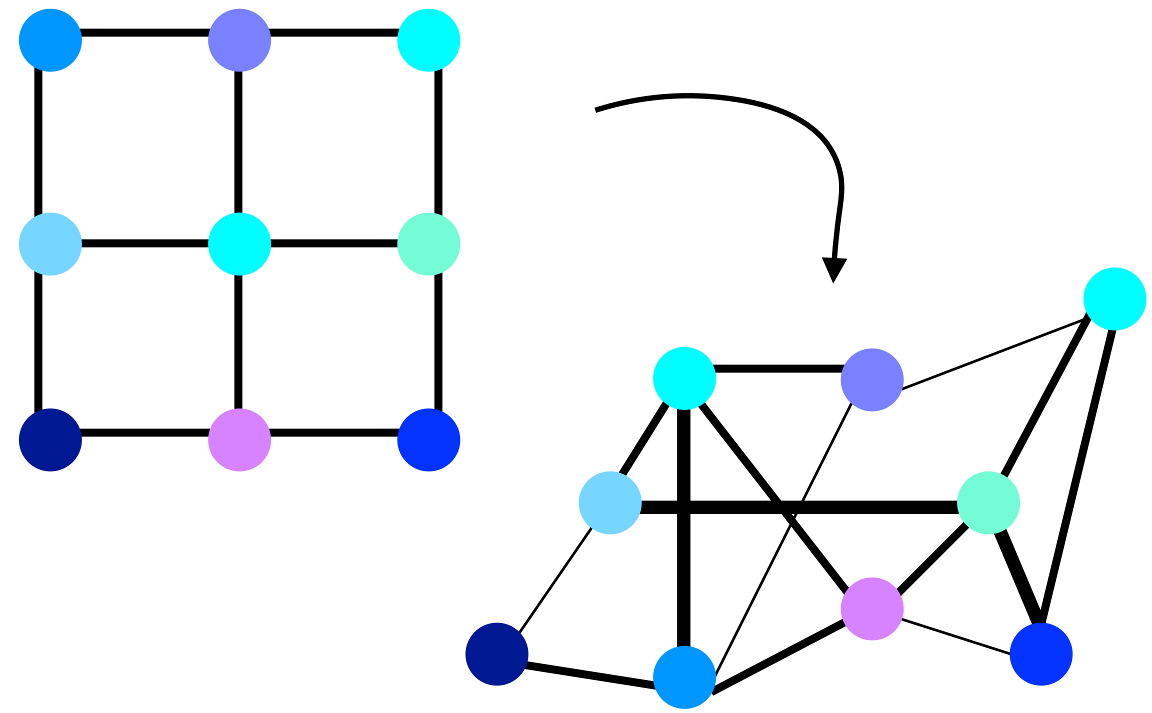




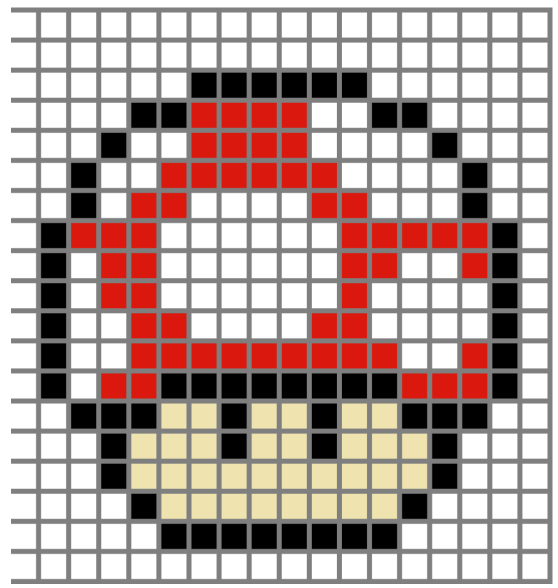


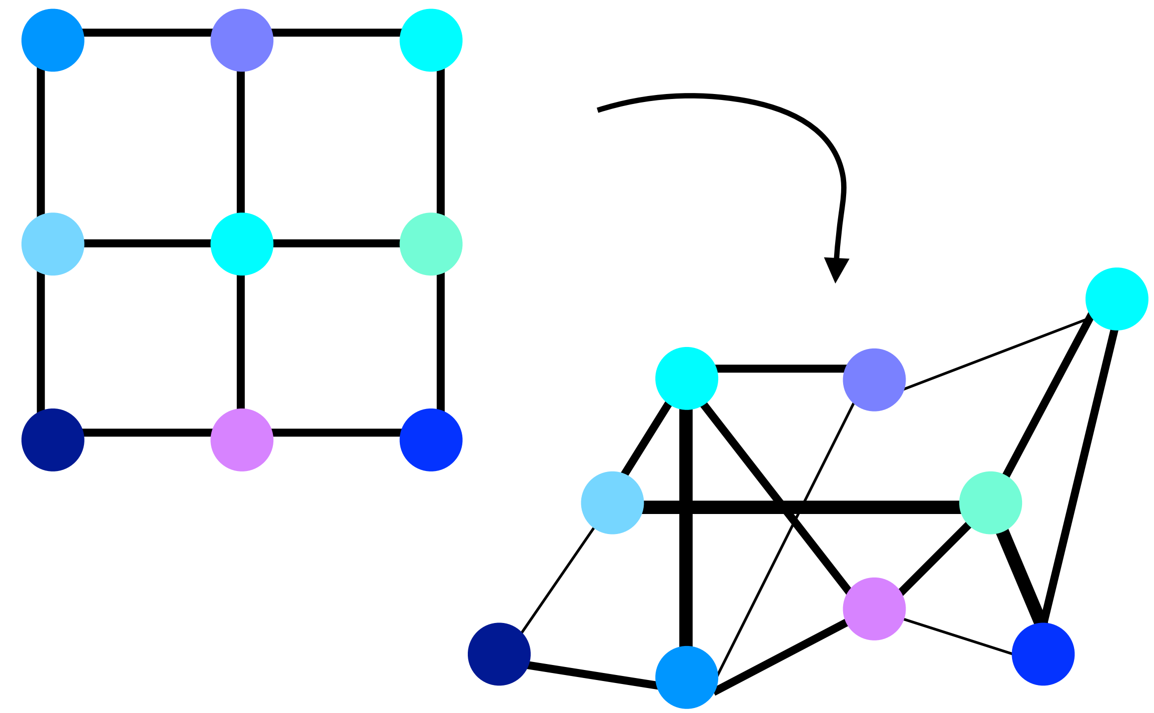
What about irregular domains?



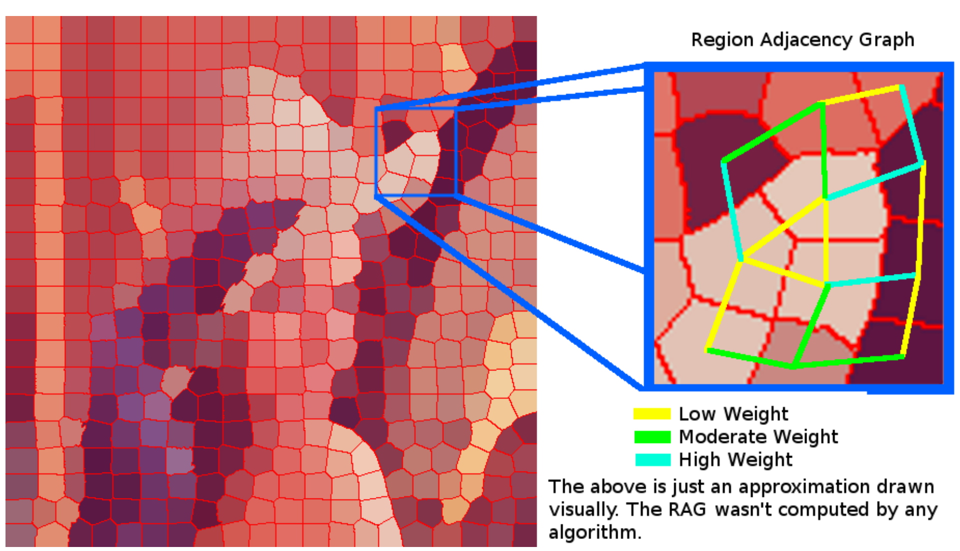


What about irregular domains?

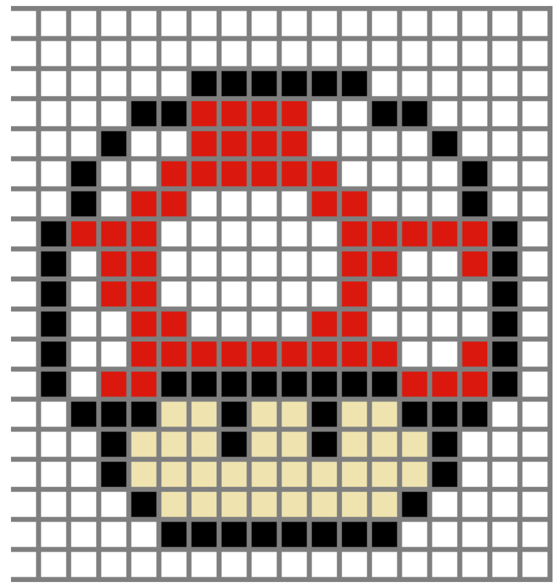


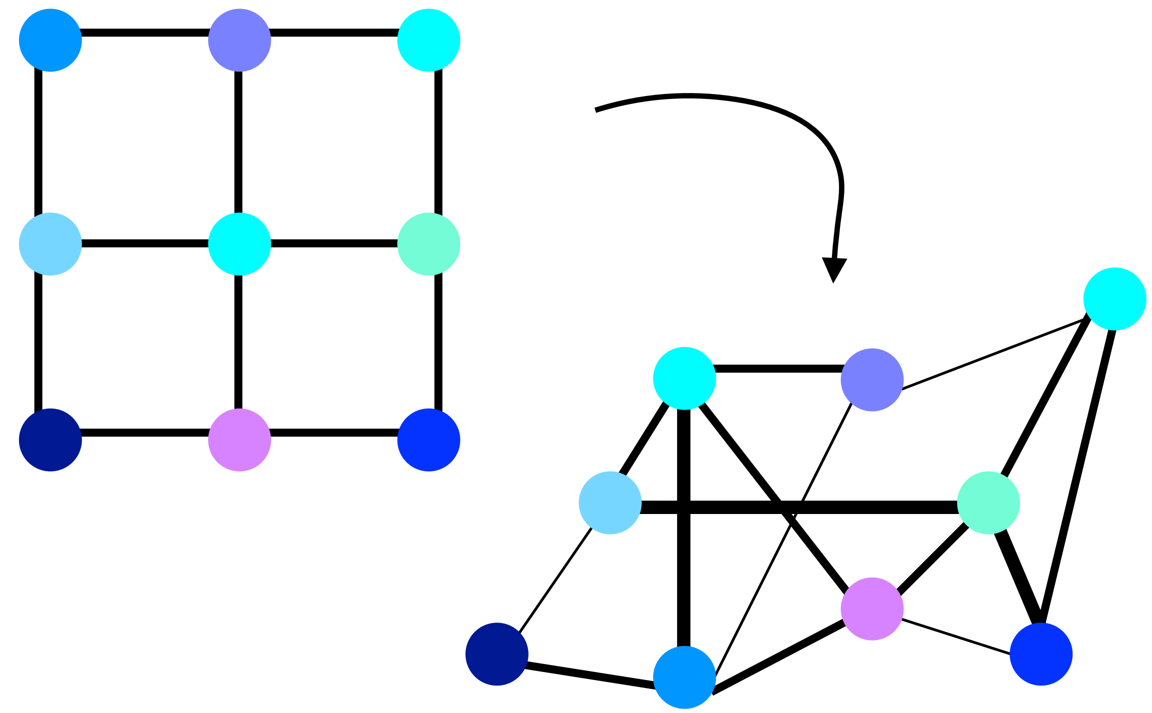


Superpixels

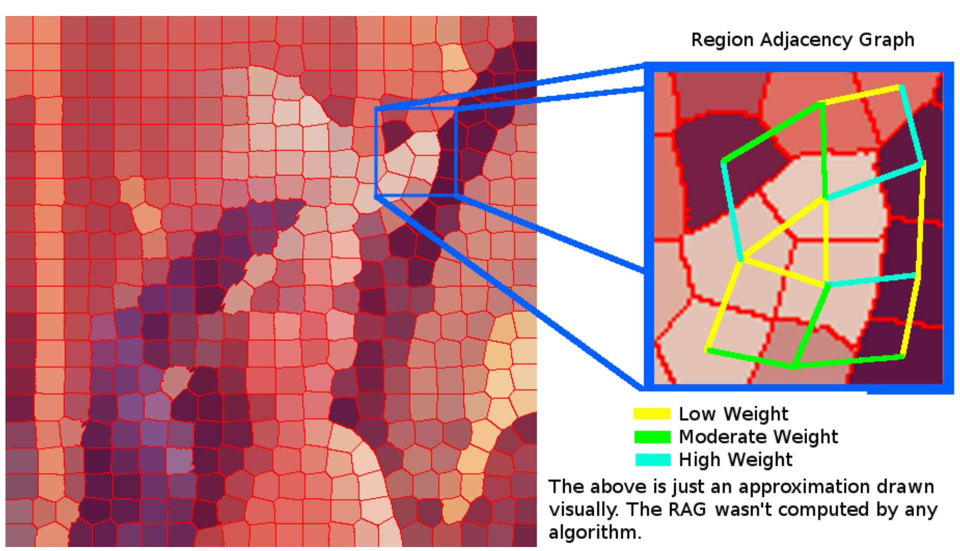


What about irregular domains?

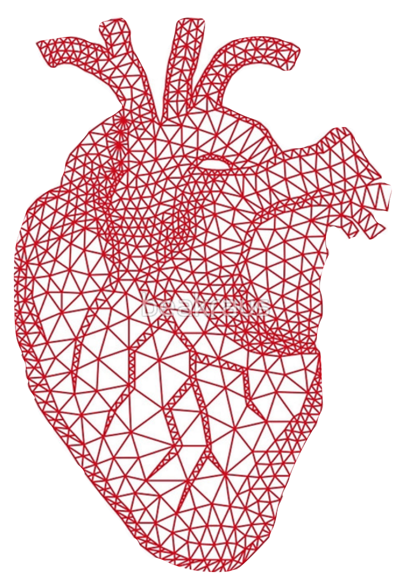




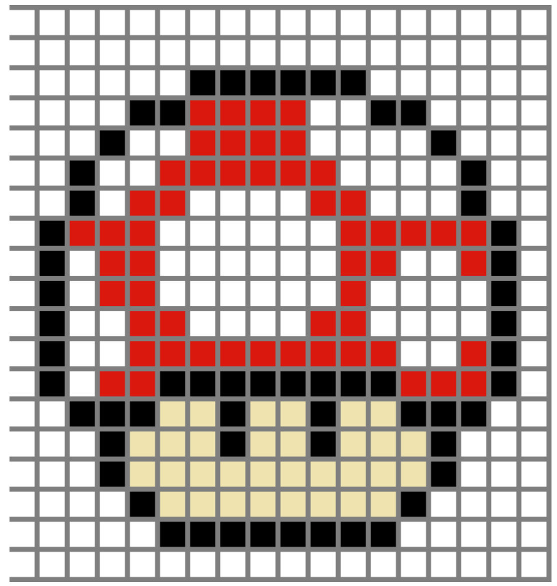
Superpixels

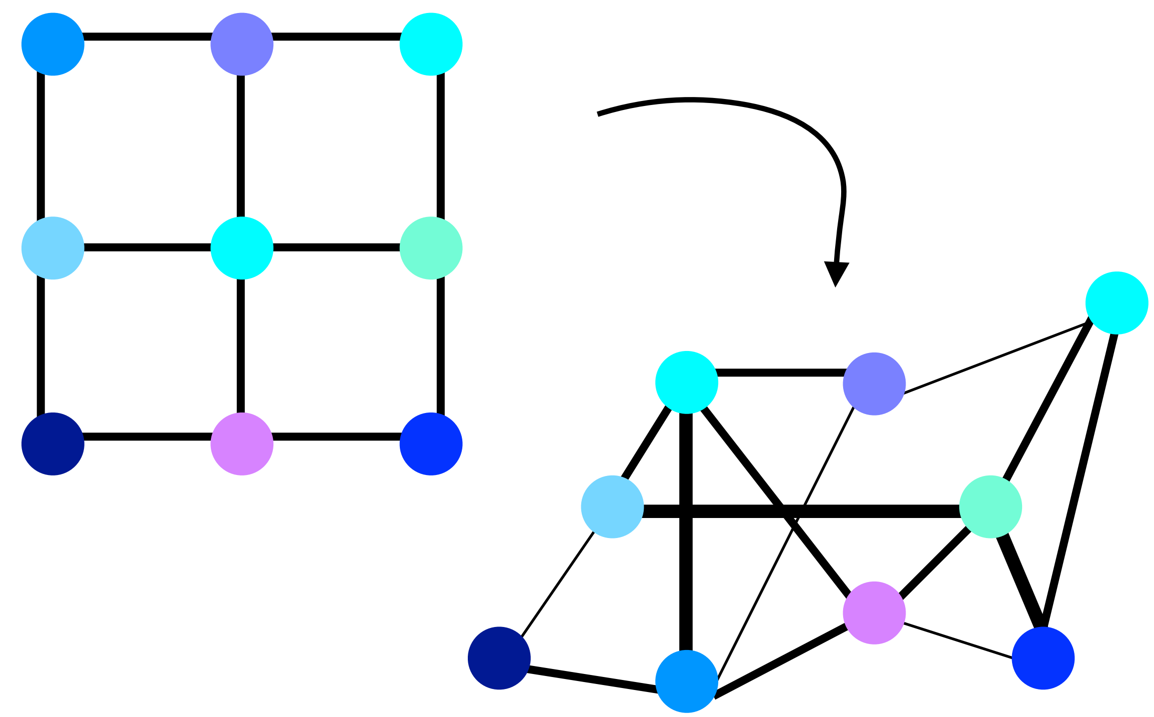


Meshes

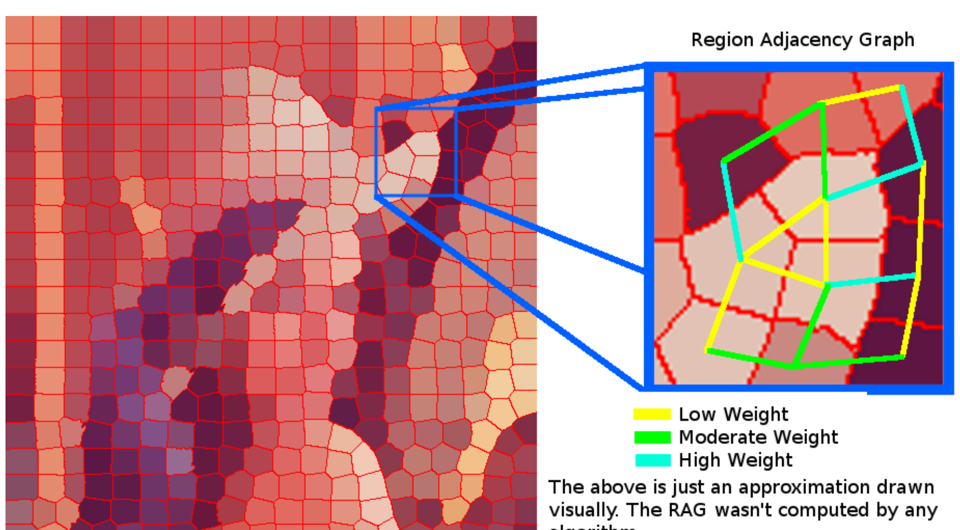


What about irregular domains?

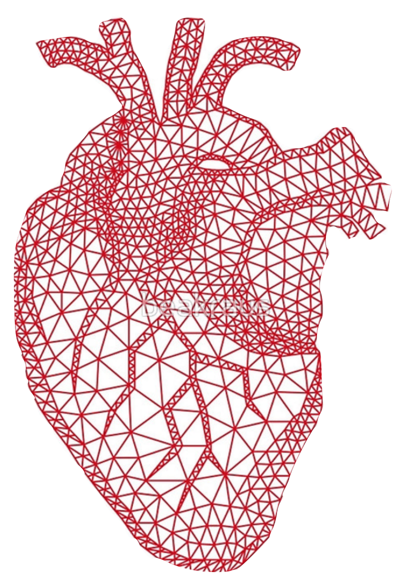




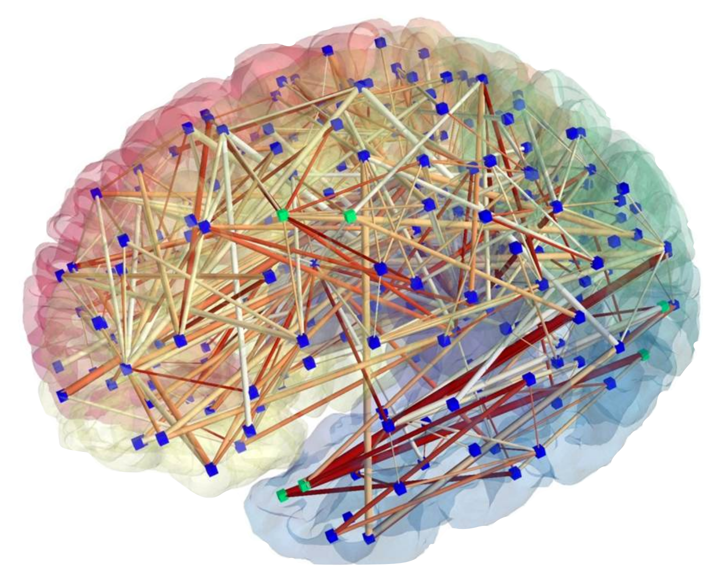
Superpixels



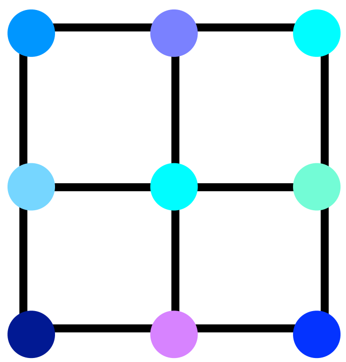
Meshes



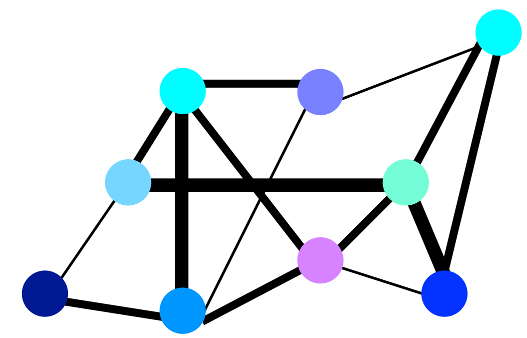
Brain Networks



Data defined on irregular graphs: Analogy



Euclidean data



Irregular data

Domain Structure

- Regular pixel grid
- Same number of neighbours per node/ pixel
- Intrinsic node ordering

- Any graph structure
- Variable number of neighbours per node
- No node ordering

Node Signal/Features

- Image intensities

- Feature vector assigned to each node

Tasks

- Image classification
- Image segmentation

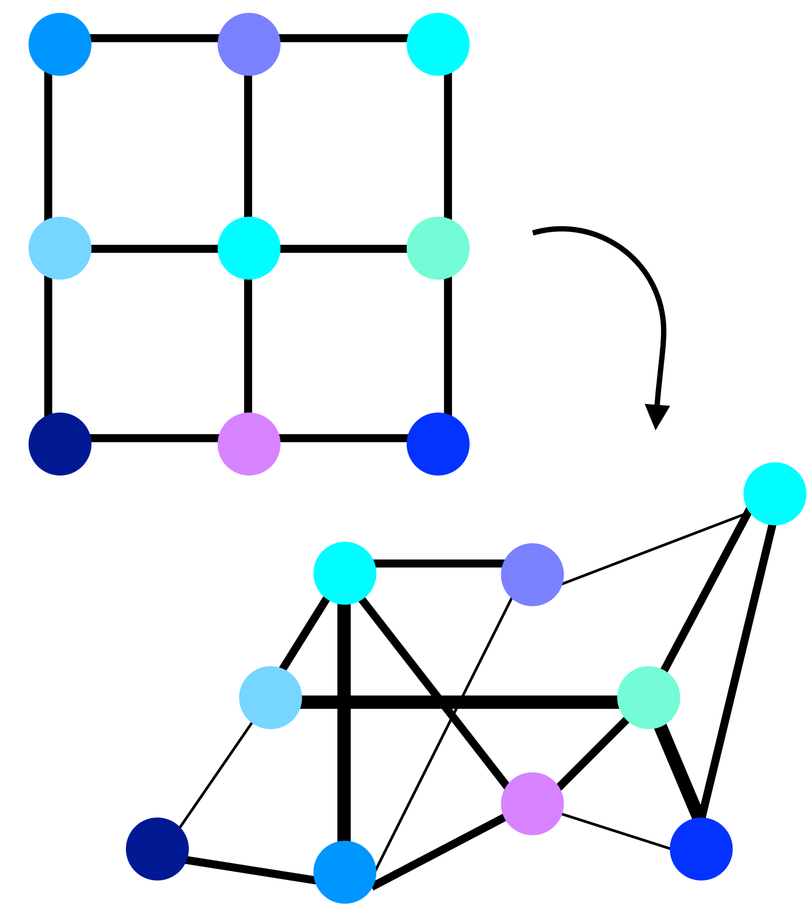
- Whole graph classification
- Node classification

Graph Convolutional Neural Networks

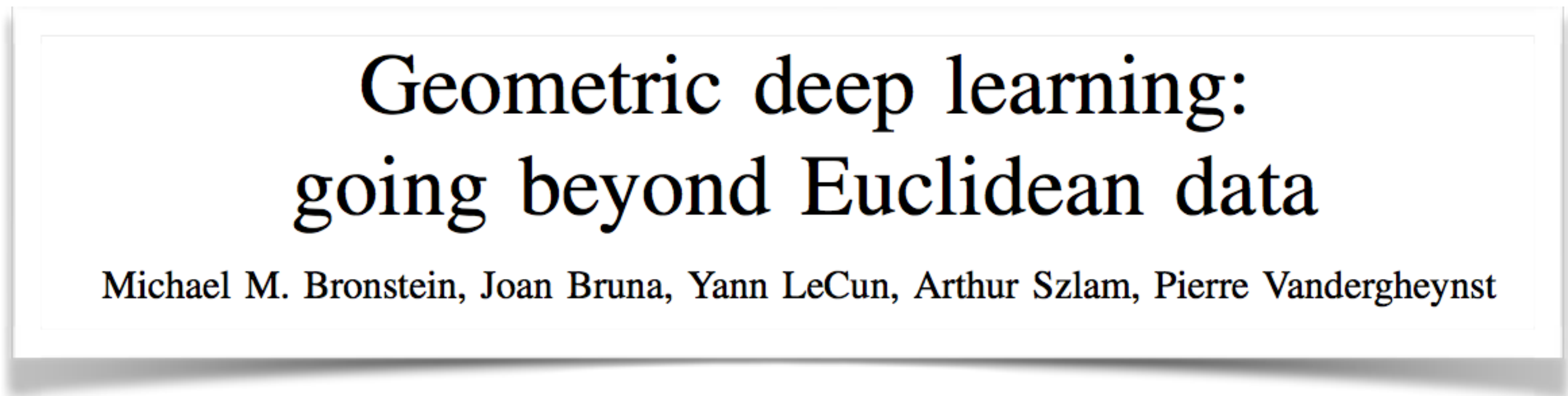
Geometric deep learning: going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst

- Extending the concept of CNNs to irregular domains

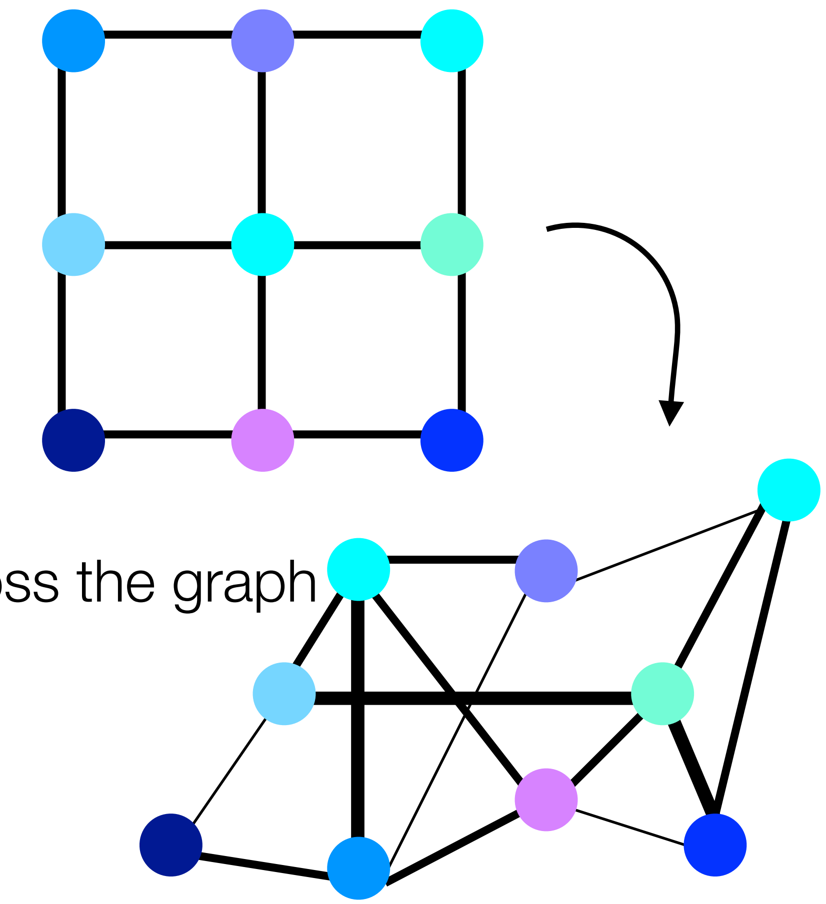


Graph Convolutional Neural Networks


Geometric deep learning:
going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst

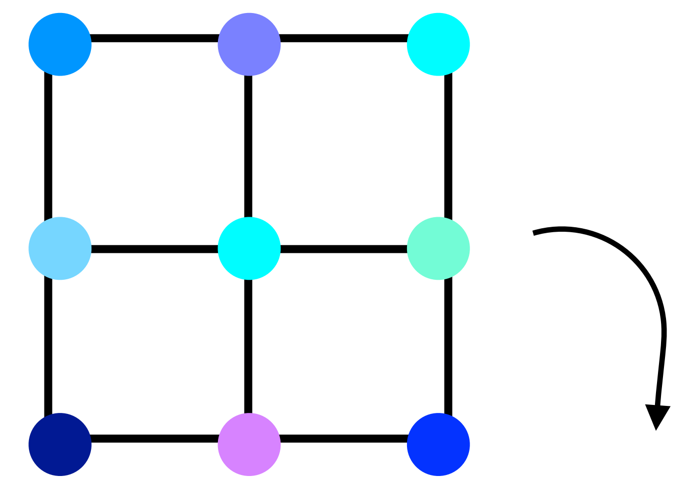
- Extending the concept of CNNs to irregular domains
    - **Spatial filtering** : sliding a filter of defined receptive field across the graph



Graph Convolutional Neural Networks

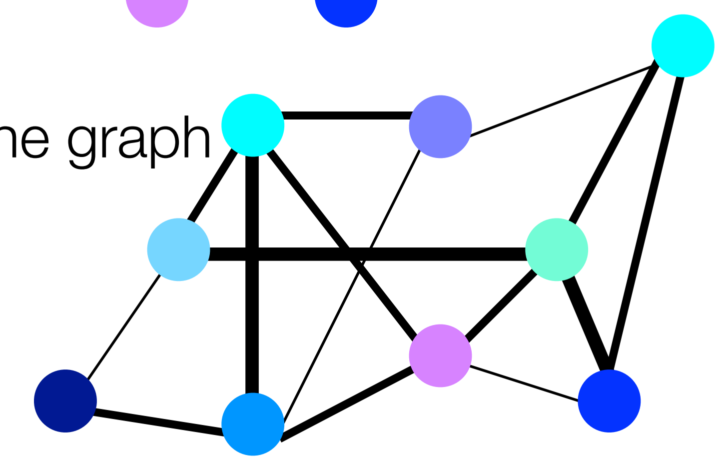
Geometric deep learning: going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst



- Extending the concept of CNNs to irregular domains
  - **Spatial filtering** : sliding a filter of defined receptive field across the graph

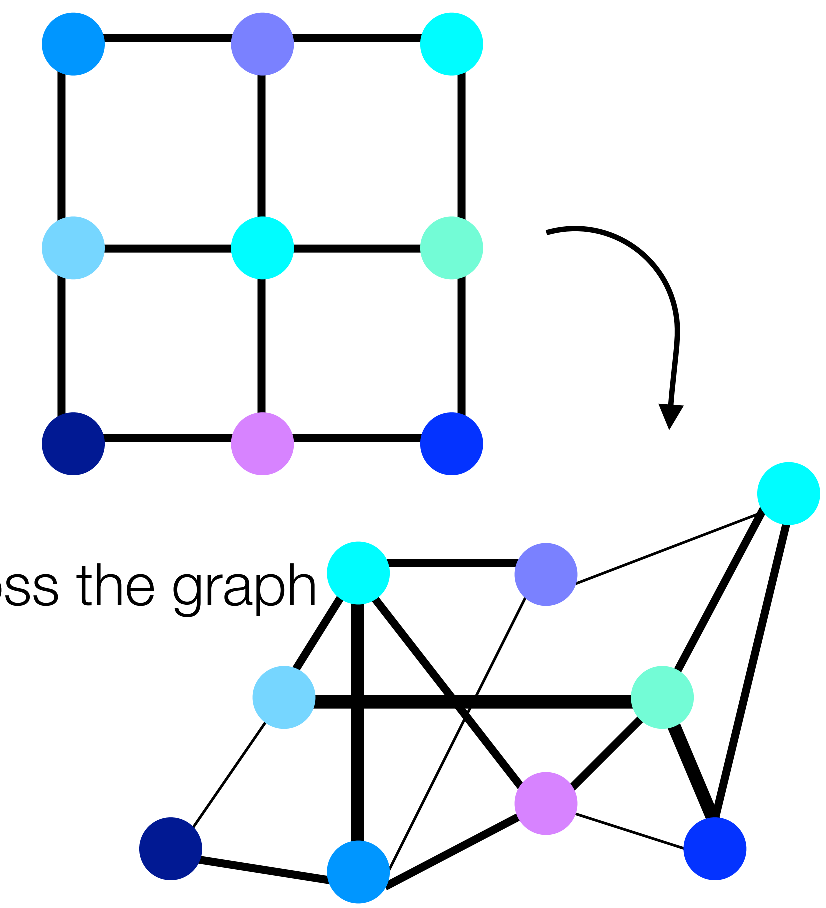
    Most natural analogy with regular structures
    Intuitive interpretation



Graph Convolutional Neural Networks

Geometric deep learning: going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst

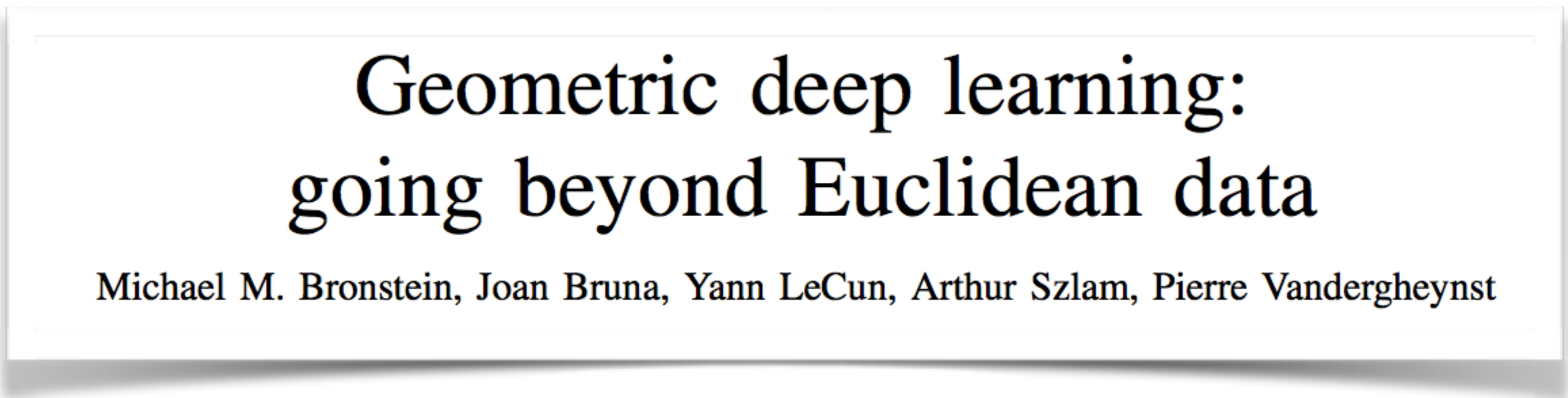


- Extending the concept of CNNs to irregular domains
  - **Spatial filtering** : sliding a filter of defined receptive field across the graph

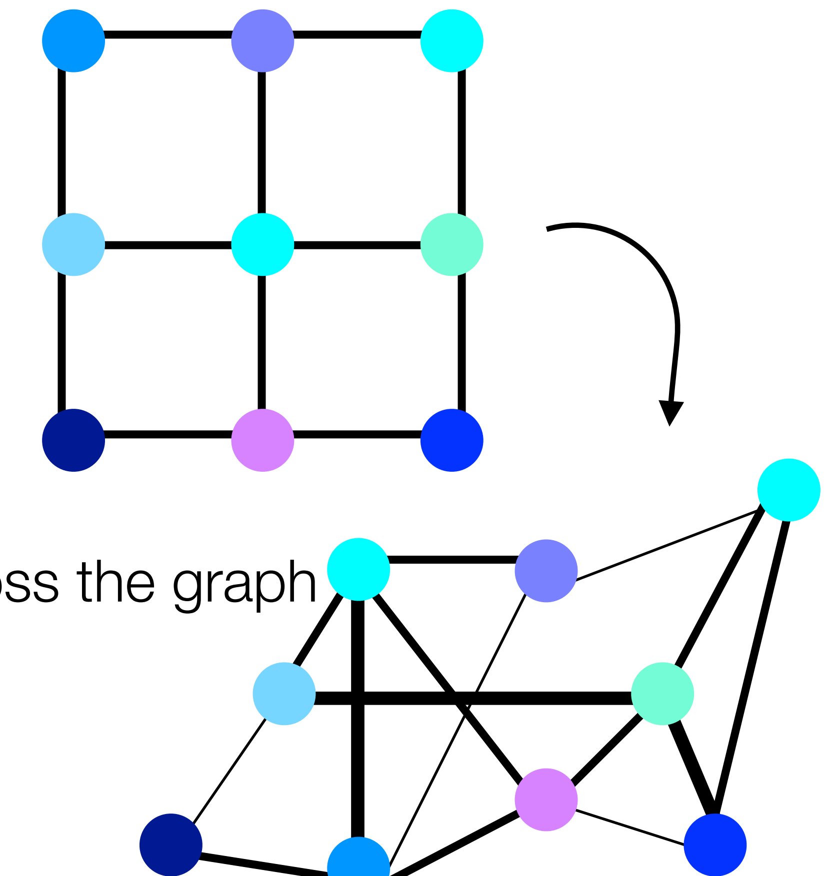
    😁 Most natural analogy with regular structures
    Intuitive interpretation

    😡 Requires defining a neighbourhood system and a node ordering -> not straightforward

Graph Convolutional Neural Networks


Geometric deep learning:
going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst



- Extending the concept of CNNs to irregular domains
  - **Spatial filtering** : sliding a filter of defined receptive field across the graph

    😁 Most natural analogy with regular structures
    Intuitive interpretation

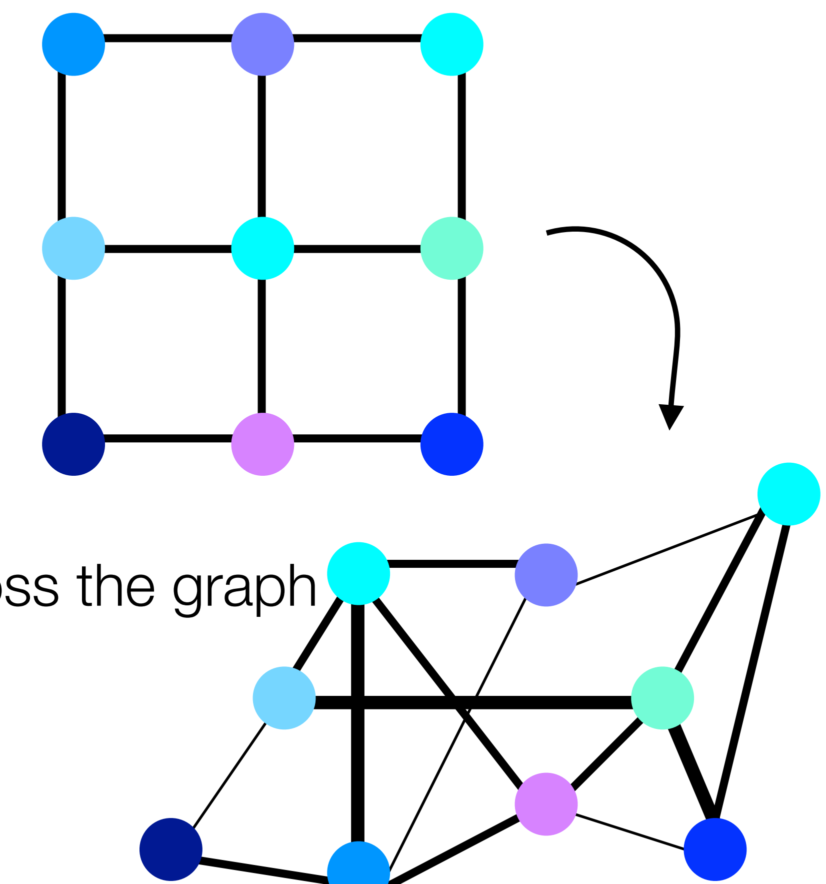
    😡 Requires defining a neighbourhood system and a node ordering -> not straightforward

  - **Spectral filtering**: Exploiting the concept that convolutions in the spatial domain corresponds to multiplications in the Fourier domain

Graph Convolutional Neural Networks

Geometric deep learning: going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst



- Extending the concept of CNNs to irregular domains

  - **Spatial filtering** : sliding a filter of defined receptive field across the graph

    😁 Most natural analogy with regular structures
    Intuitive interpretation

    😡 Requires defining a neighbourhood system and a node ordering -> not straightforward

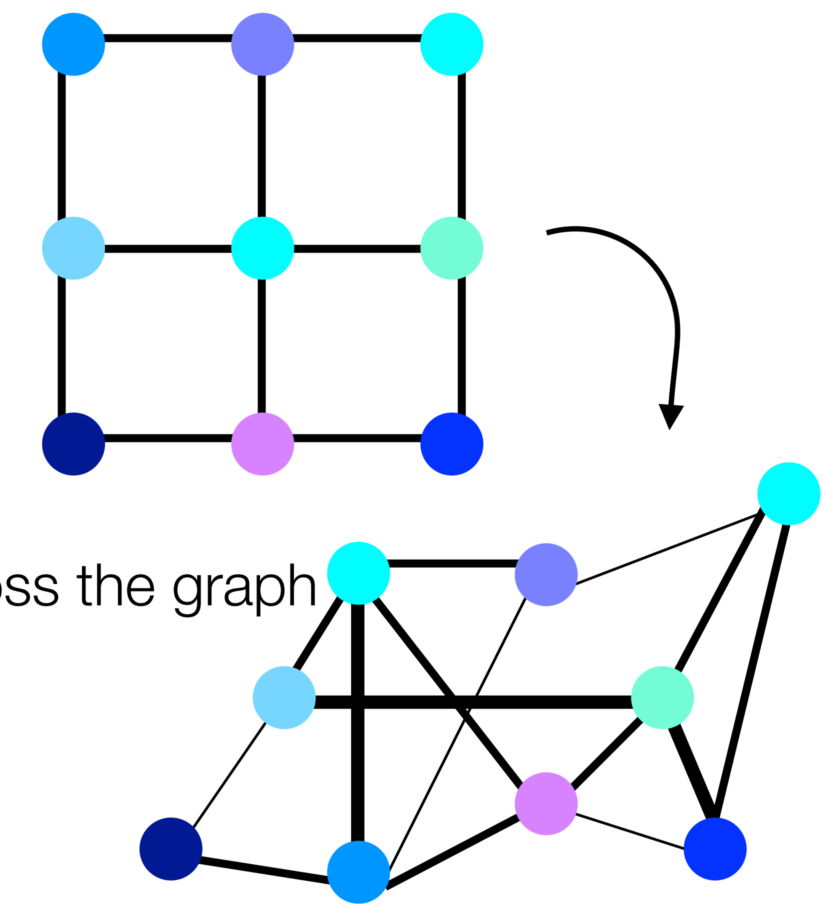
  - **Spectral filtering**: Exploiting the concept that convolutions in the spatial domain corresponds to multiplications in the Fourier domain

    😁 No neighbourhood, node ordering definition
    Principled definition of the convolution operator
    Can obtain strictly localised filters

Graph Convolutional Neural Networks

Geometric deep learning: going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst



- Extending the concept of CNNs to irregular domains
  - **Spatial filtering** : sliding a filter of defined receptive field across the graph

    😬 Most natural analogy with regular structures
    Intuitive interpretation

    😡 Requires defining a neighbourhood system and a node ordering -> not straightforward

  - **Spectral filtering**: Exploiting the concept that convolutions in the spatial domain corresponds to multiplications in the Fourier domain

    😬 No neighbourhood, node ordering definition
    Principled definition of the convolution operator
    Can obtain strictly localised filters

    😡 Non transferrable between different graph structures

Spatial GCNs

- Tailored to a specific type of data [Kawahara et al. NeuroImage 2017]
- Learning a patch operator which represents the neighbourhood system
  - e.g. using geodesic distance, diffusion kernels or mixture models [Monti et al. 2017, Bronstein et al. 2017]
- Filters conditioned on graph edges [Simonovsky et al. 2017]
- Main advantage: Trained spatial GCNs can be applied to data defined on variable domain structures

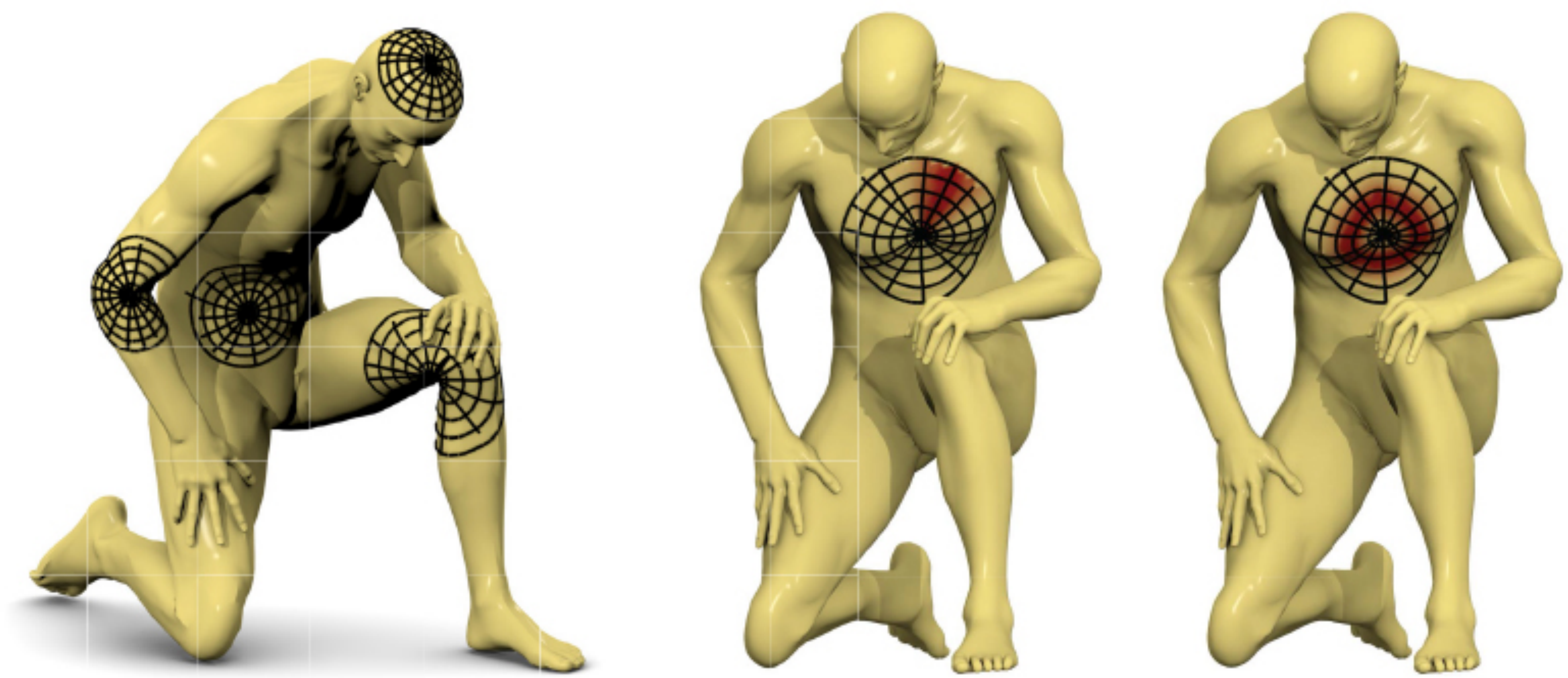


Image source: http://vgl.ict.usc.edu/Research/GeometricDeepLearning/

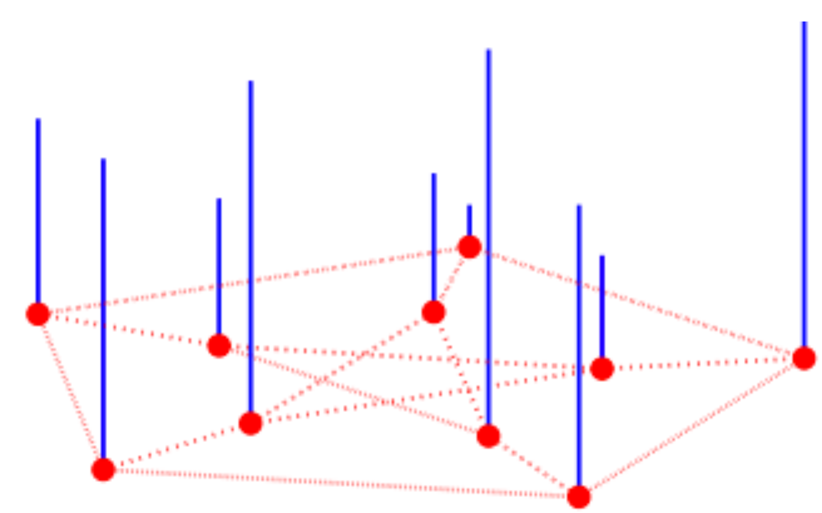
Graph Signal Processing

The Emerging Field of Signal Processing on Graphs

Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains

David I Shuman†, Sunil K. Narang ‡, Pascal Frossard†, Antonio Ortega‡ and Pierre Vandergheynst†
†Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Laboratory (LTS2 and LTS4)
‡University of Southern California (USC), Signal and Image Processing Institute

- Analysing signals $x \in \mathcal{V} \to \mathbb{R}$ defined on an undirected weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$



Graph Signal Processing

The Emerging Field of Signal Processing on Graphs

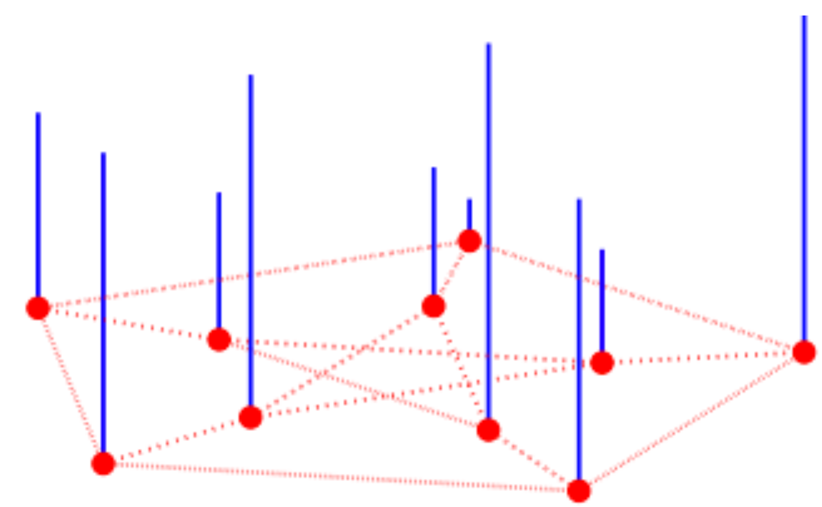
Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains

David I Shuman[†], Sunil K. Narang [‡], Pascal Frossard[†], Antonio Ortega[‡] and Pierre Vandergheynst[†]
†Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Laboratory (LTS2 and LTS4)
‡University of Southern California (USC), Signal and Image Processing Institute

- Analysing signals $x \in \mathcal{V} \rightarrow \mathbb{R}$ defined on an undirected weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$
- Graph laplacian: $L = D - W, D_{ii} = \sum_j Wij$

  acts as a **difference operator** on the signal x

$$(Lx)(i) = \sum_{j \in \mathcal{N}_i} W_{ij}[x(i) - x(j)]$$



Graph Signal Processing

The Emerging Field of Signal Processing on Graphs

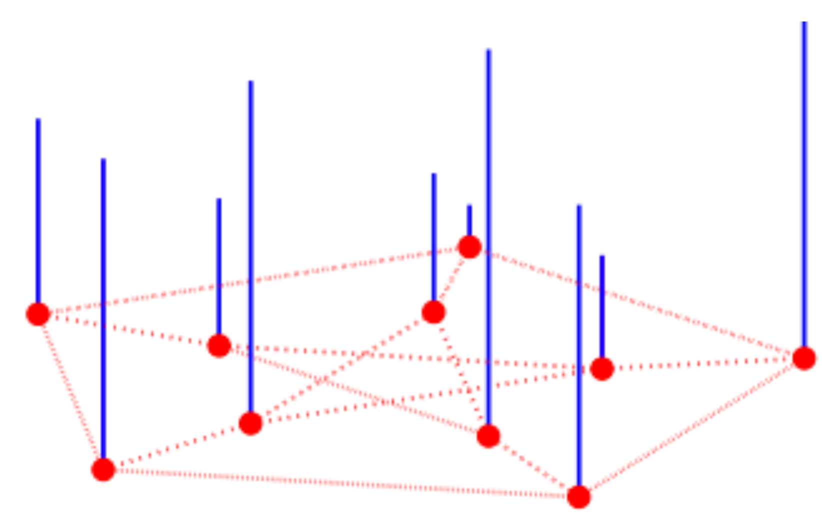
Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains

David I Shuman†, Sunil K. Narang ‡, Pascal Frossard†, Antonio Ortega‡ and Pierre Vandergheynst†
†Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Laboratory (LTS2 and LTS4)
‡University of Southern California (USC), Signal and Image Processing Institute

- Analysing signals $x \in \mathcal{V} \to \mathbb{R}$ defined on an undirected weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$
- Graph laplacian: $L = D - W, D_{ii} = \sum_j Wij$

  acts as a **difference operator** on the signal x

$$(Lx)(i) = \sum_{j \in \mathcal{N}_i} \boxed{W_{ij}} [x(i) - x(j)]$$



Graph Signal Processing

The Emerging Field of Signal Processing on Graphs

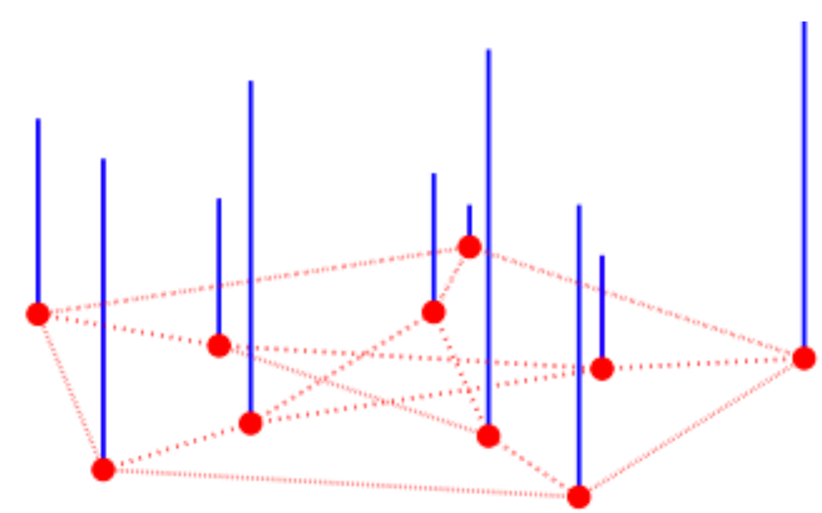
Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains

David I Shuman†, Sunil K. Narang ‡, Pascal Frossard†, Antonio Ortega‡ and Pierre Vandergheynst†
†Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Laboratory (LTS2 and LTS4)
‡University of Southern California (USC), Signal and Image Processing Institute

- Analysing signals $x \in \mathcal{V} \rightarrow \mathbb{R}$ defined on an undirected weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$
- Graph laplacian: $L = D - W, D_{ii} = \sum_j Wij$

  acts as a **difference operator** on the signal x

$$(Lx)(i) = \sum_{j \in \mathcal{N}_i} \boxed{W_{ij}} [x(i) - x(j)]$$



- L is symmetric and positive semi-definite: $L = U \Lambda U^T$

Graph Fourier Transform

- Analogy with the euclidean domain: expansion of a signal in terms of eigenfunctions of the Laplace operator

D. I Shuman et al. , "The emerging field of signal processing on graphs" , 2013

Graph Fourier Transform

- Analogy with the euclidean domain: expansion of a signal in terms of eigenfunctions of the Laplace operator
- Graph signal: $x \in \mathcal{V} \to \mathbb{R}$
- Laplacian eigendecomposition: $L = U \Lambda U^T$

D. I Shuman et al. , "The emerging field of signal processing on graphs" , 2013

Graph Fourier Transform

- Analogy with the euclidean domain: expansion of a signal in terms of eigenfunctions of the Laplace operator
- Graph signal: $x \in \mathcal{V} \to \mathbb{R}$
- Laplacian eigendecomposition: $L = U \Lambda U^T$
- Graph Fourier transform: $\hat{x} = U^T x$
- Graph inverse Fourier transform: $x = U \hat{x}$

D. I Shuman et al. , "The emerging field of signal processing on graphs" , 2013

Graph Fourier Transform

- Analogy with the euclidean domain: expansion of a signal in terms of eigenfunctions of the Laplace operator
- Graph signal: $x \in \mathcal{V} \to \mathbb{R}$
- Laplacian eigendecomposition: $L = U \Lambda U^T$
- Graph Fourier transform: $\hat{x} = U^T x$
- Graph inverse Fourier transform: $x = U \hat{x}$
- Graph Fourier orthonormal basis: $U \in \mathbb{R}^{N \times N}$
- Graph “frequencies”: $\Lambda = diag(\lambda_i) \in \mathbb{R}^{N \times N}$

D. I Shuman et al. , "The emerging field of signal processing on graphs” , 2013

Graph Fourier Transform

- Analogy with the euclidean domain: expansion of a signal in terms of eigenfunctions of the Laplace operator
- Graph signal: $x \in \mathcal{V} \rightarrow \mathbb{R}$
- Laplacian eigendecomposition: $L = U \Lambda U^T$
- Graph Fourier transform: $\hat{x} = U^T x$
- Graph inverse Fourier transform: $x = U \hat{x}$
- Graph Fourier orthonormal basis: $U \in \mathbb{R}^{N \times N}$
- Graph “frequencies”: $\Lambda = diag(\lambda_i) \in \mathbb{R}^{N \times N}$

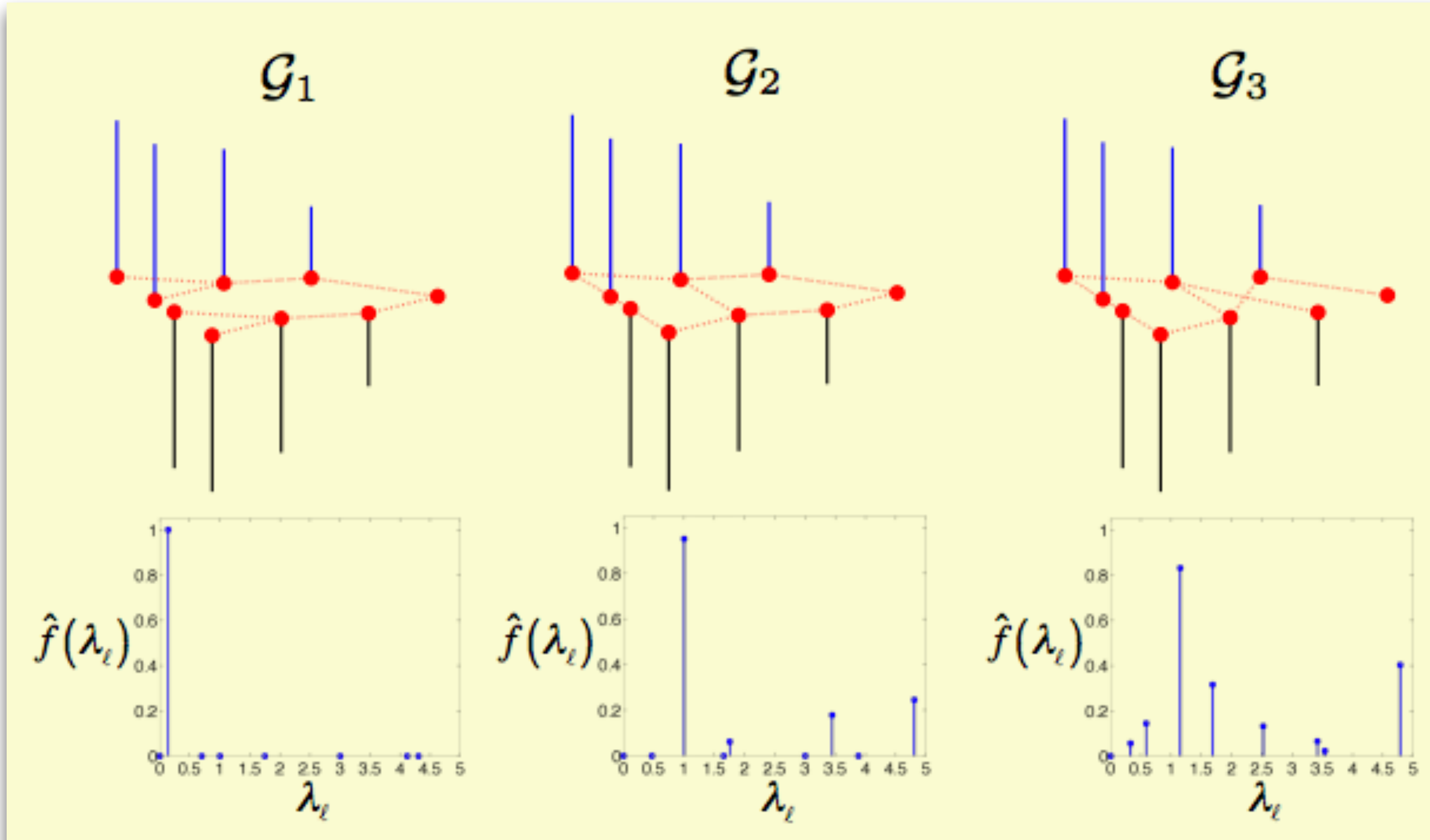
Trained spectral graphs CNNs are non transferrable across graphs

D. I Shuman et al. , "The emerging field of signal processing on graphs” , 2013

Graph Fourier Transform: smoothness of the signal

Spatial domain

Spectral domain



D. I Shuman et al. , "The emerging field of signal processing on graphs" , 2013

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = U g_\theta(\Lambda) U^T \mathbf{x}$$

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = U g_\theta(\Lambda) \boxed{U^T \mathbf{x}}$$

Fourier transform of x

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = U \boxed{g_\theta(\Lambda) U^T \mathbf{x}}$$ Filtering in the Fourier domain

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

😡 Computationally expensive

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

Computationally expensive

Filters are not localised

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

😡 Computationally expensive

😡 Filters are not localised

- Polynomial parametrisation of the filters [Defferrard et al. 2016]

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$$

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

😡 Computationally expensive

😡 Filters are not localised

- Polynomial parametrisation of the filters [Defferrard et al. 2016]

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$$

Learned parameters

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

Computationally expensive

Filters are not localised

- Polynomial parametrisation of the filters [Defferrard et al. 2016]

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$$

Learned parameters

Can be computed **recursively** directly **from the laplacian**

Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Graph convolution

- Multiplication in the Fourier domain

$$g_\theta * \mathbf{x} = \boxed{U g_\theta(\Lambda) U^T \mathbf{x}}$$

Inverse transform of the filtered signal

- Filter g_θ parametrised in the Fourier domain on the laplacian eigenvalues

😡 Computationally expensive

😡 Filters are not localised

- Polynomial parametrisation of the filters [Defferrard et al. 2016]

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$$

Learned parameters

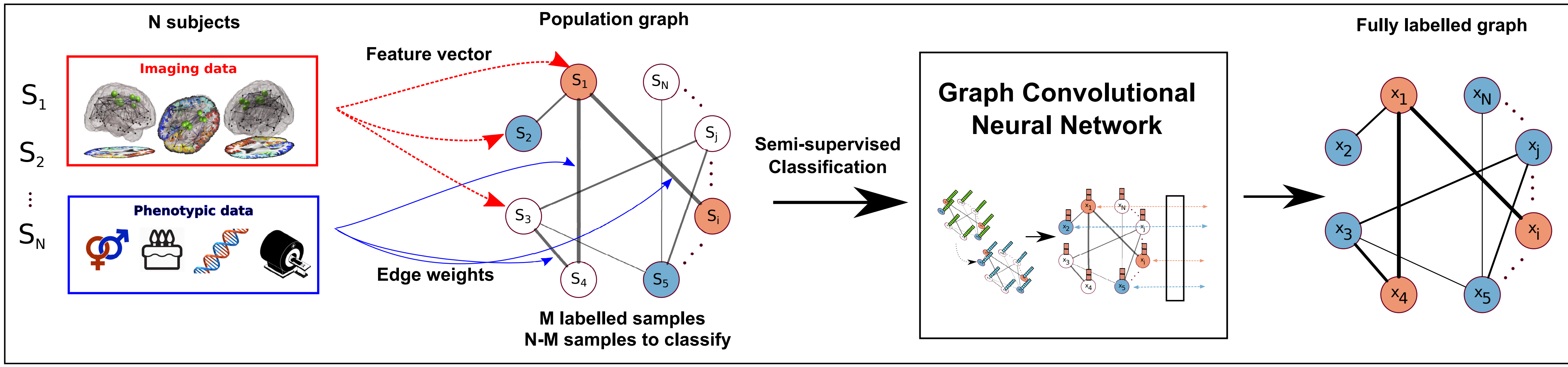
😁 Can be computed **recursively** directly **from the laplacian**

😁 Strictly K-localised (K = order of the polynomials)

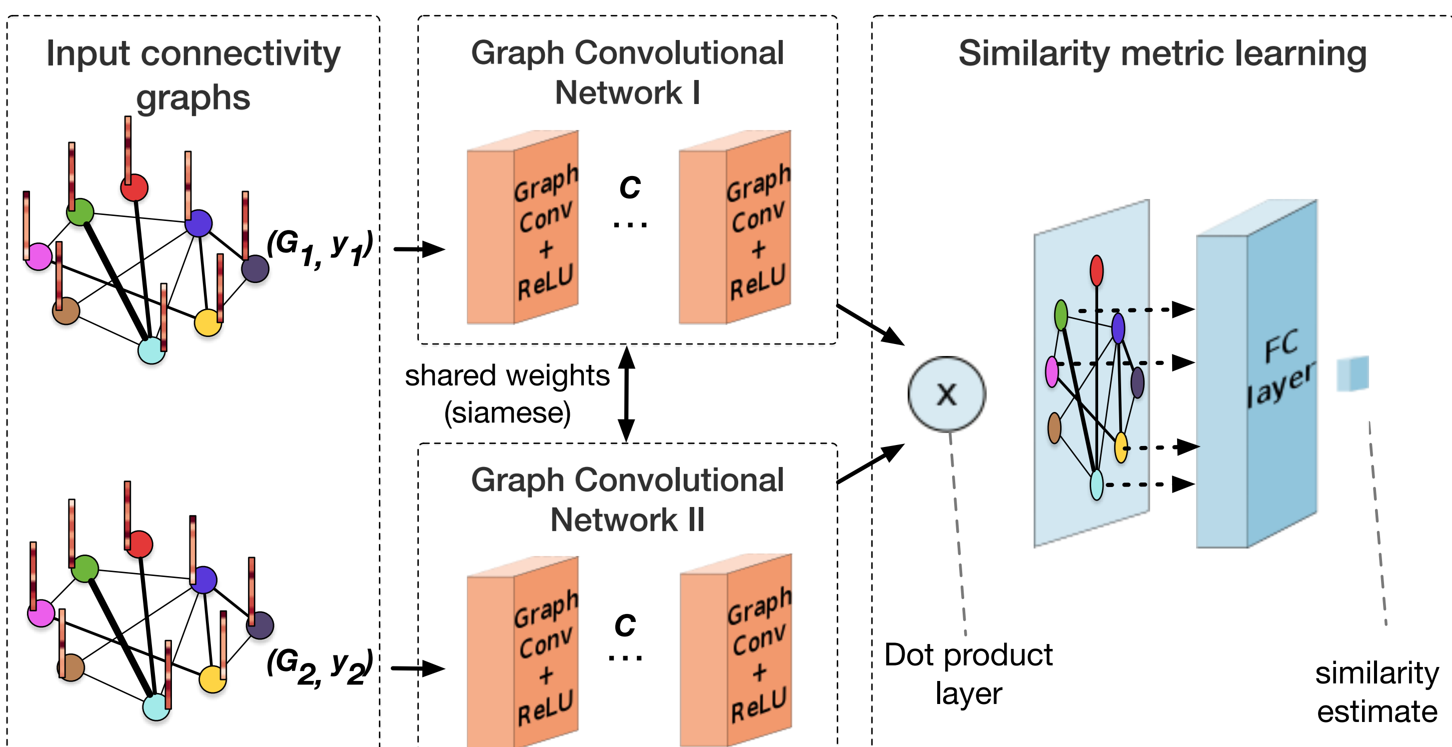
Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering 2016

Applications

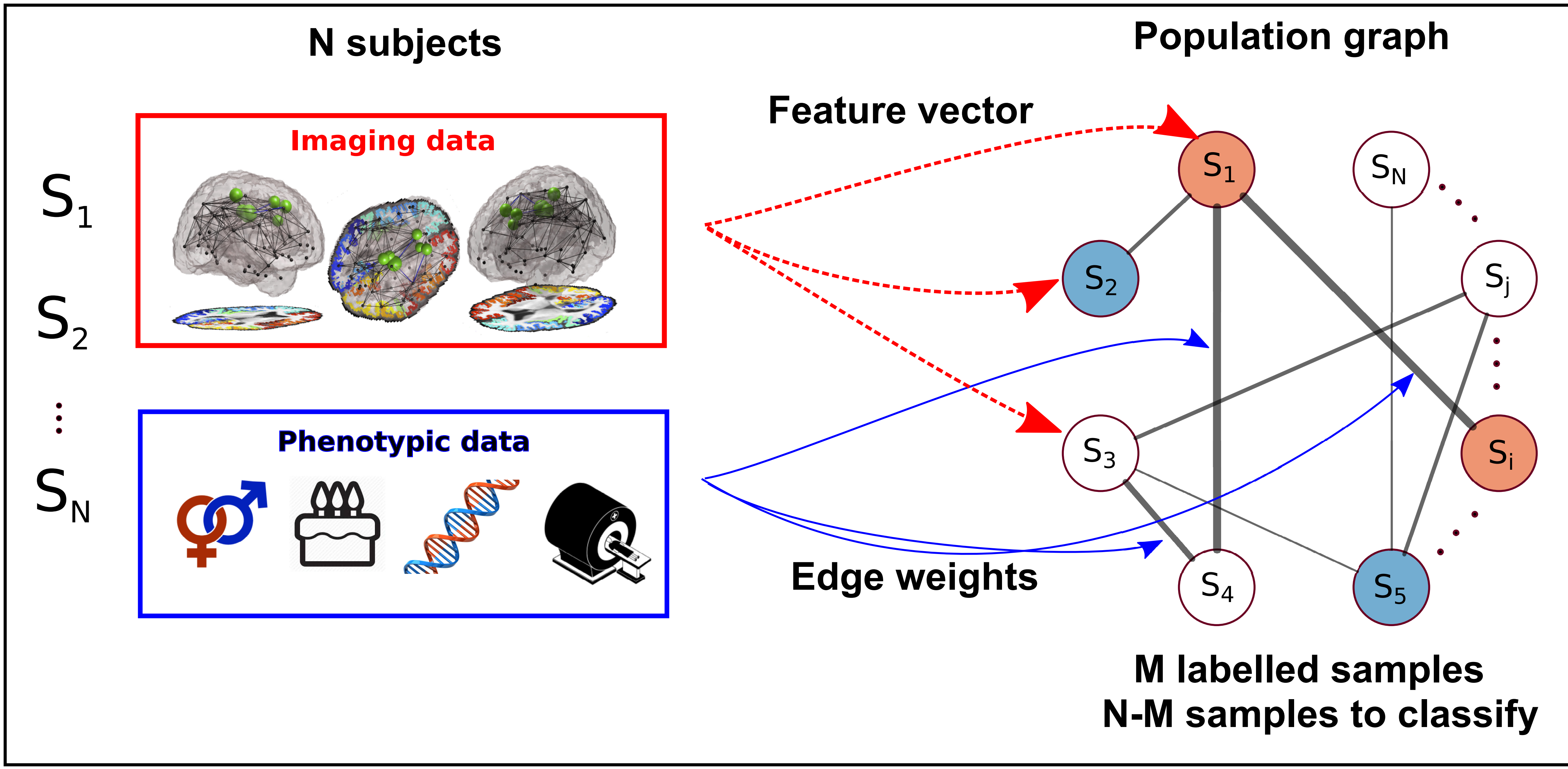
- Semi-supervised classification [Kipf et al. 2016, Parisot et al. 2017]



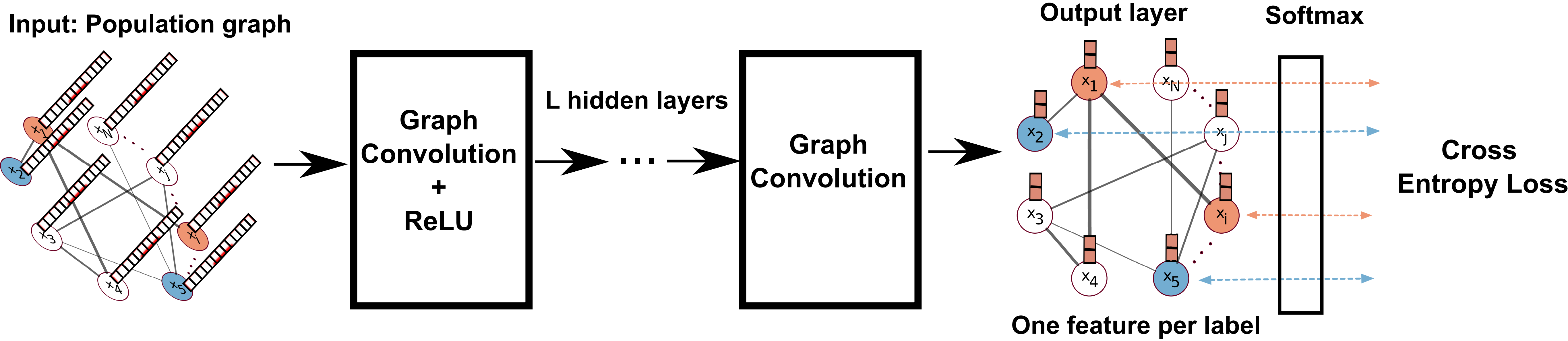
- Distance Metric learning [Ktena et al. 2017]



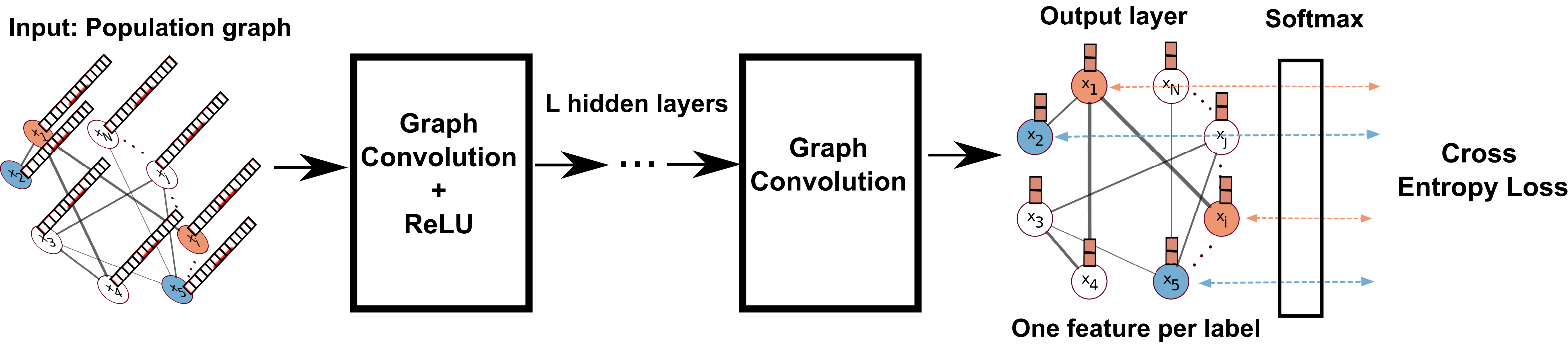
Semi-supervised node classification

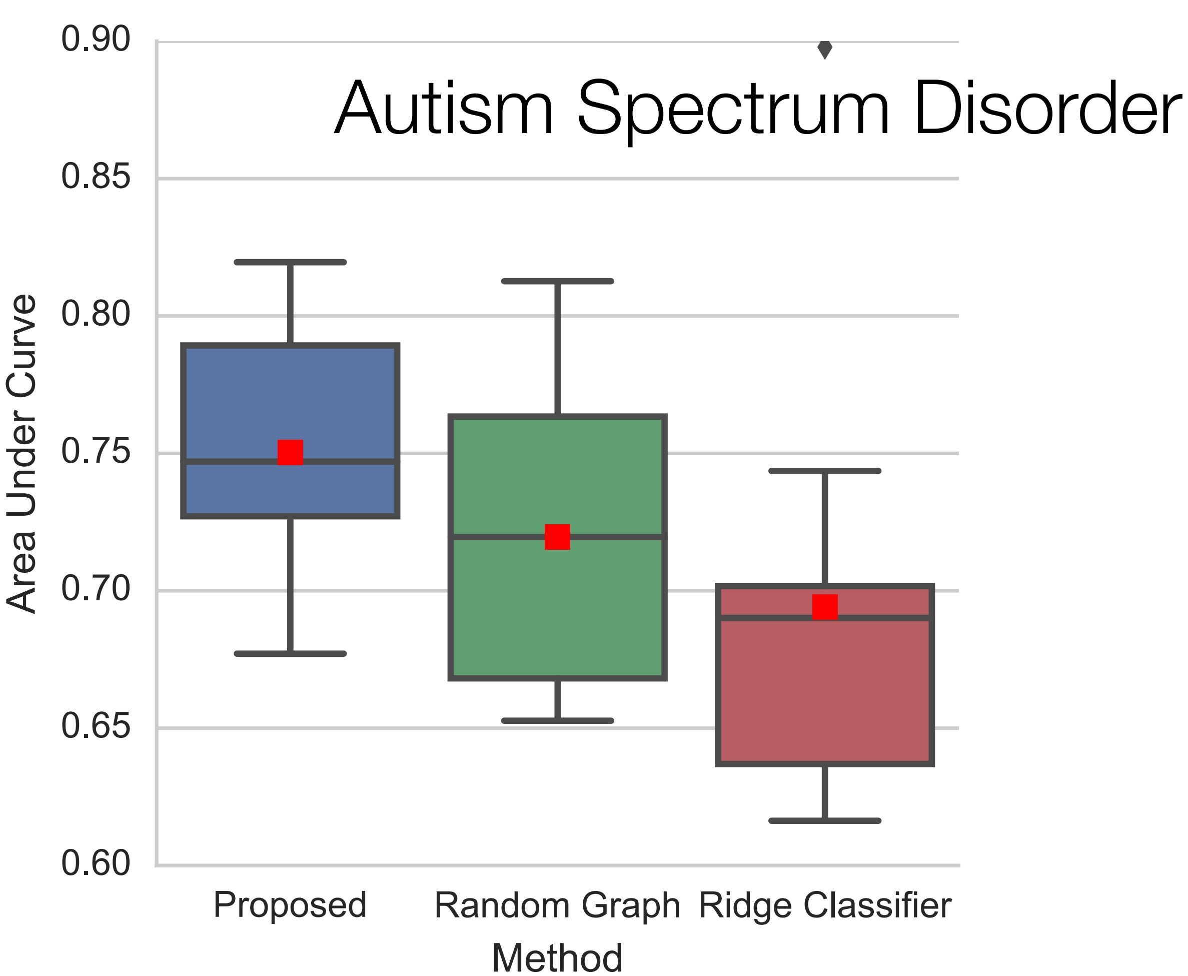


Semi-supervised node classification

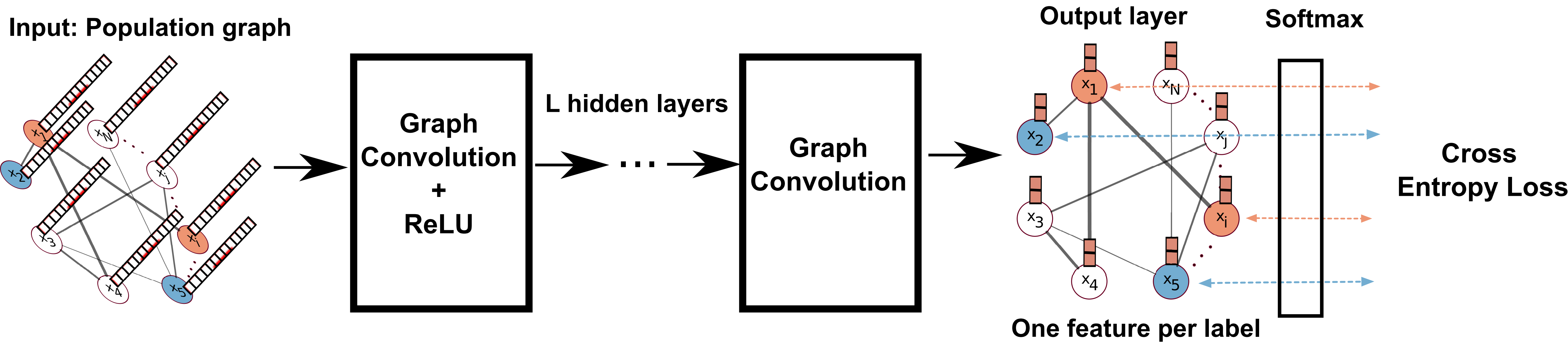


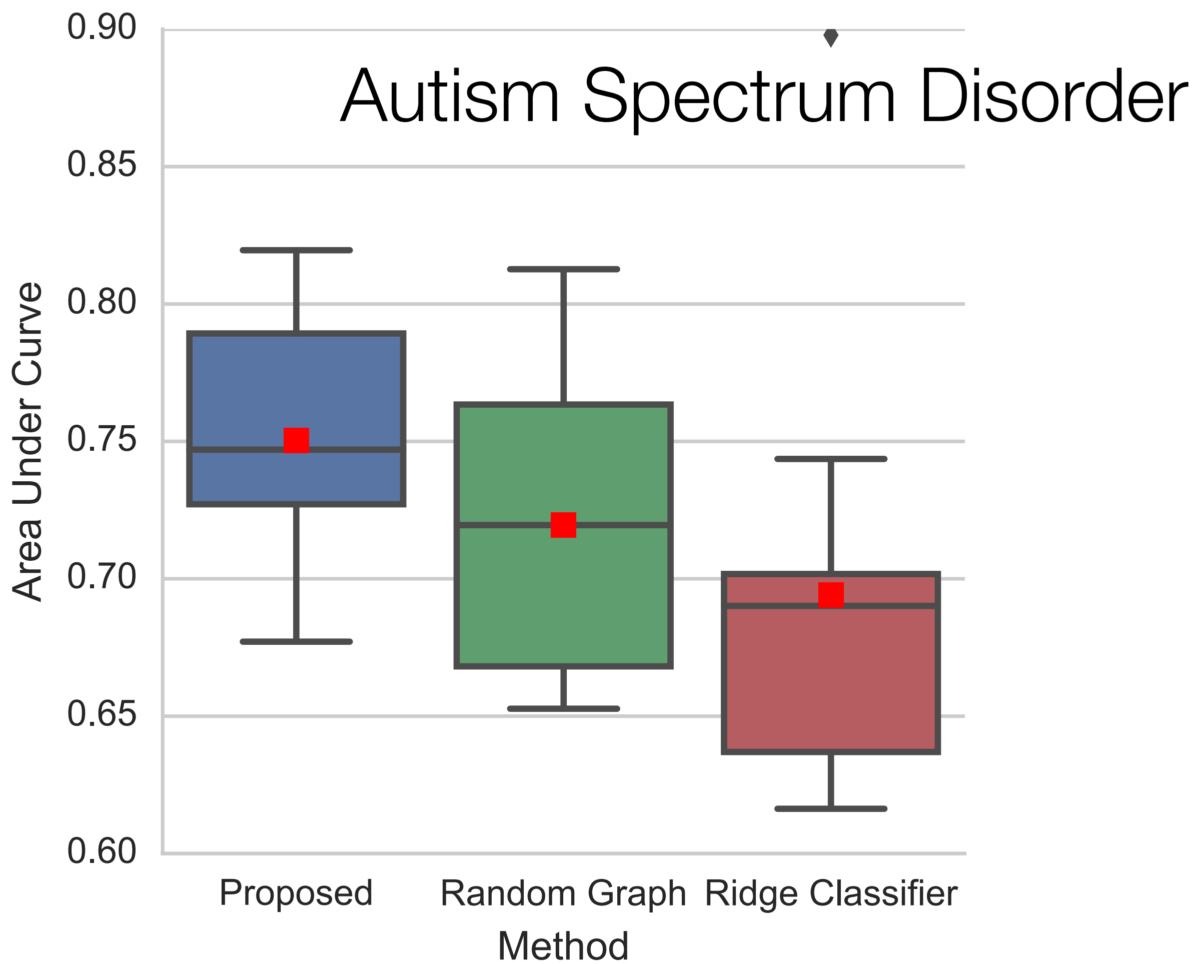
Semi-supervised node classification

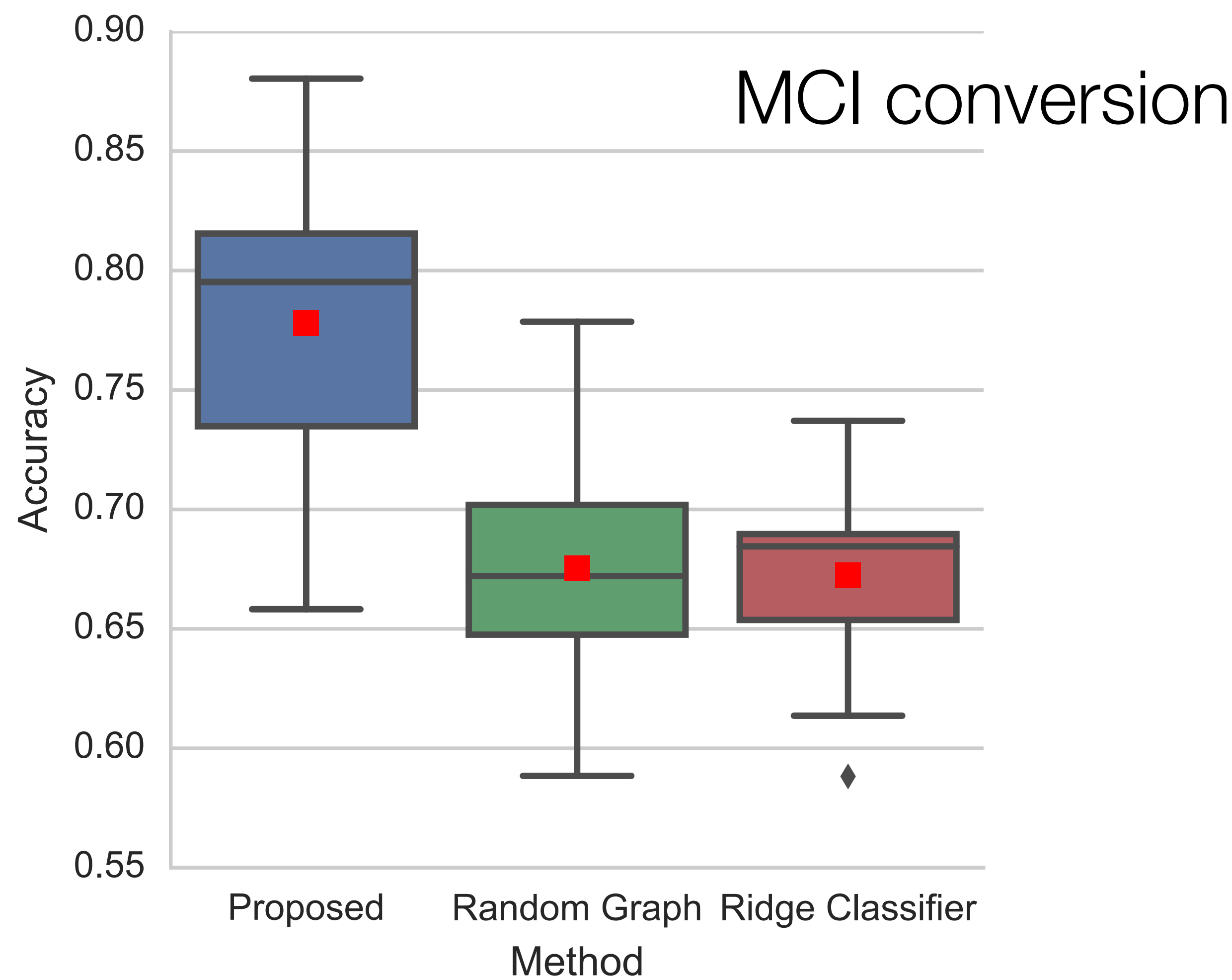




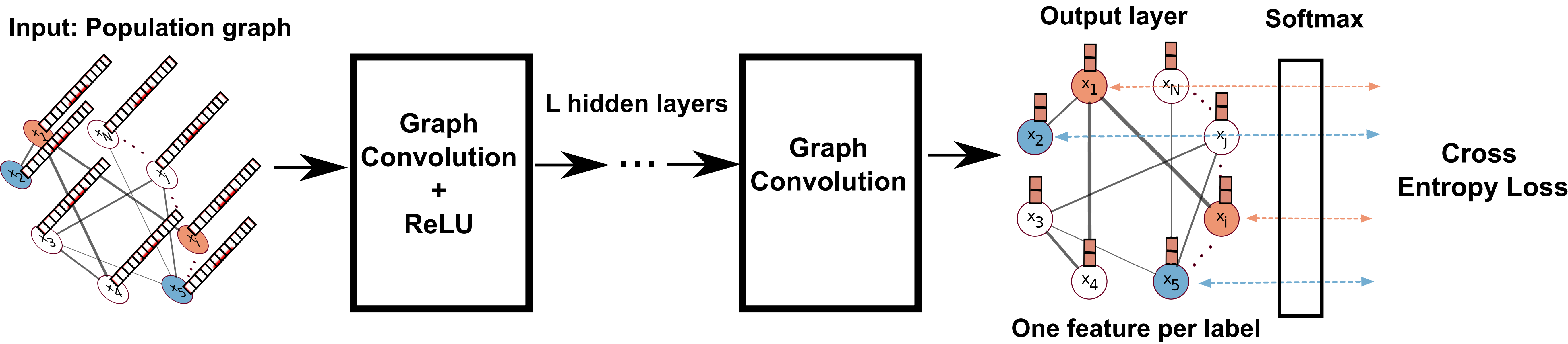
Semi-supervised node classification

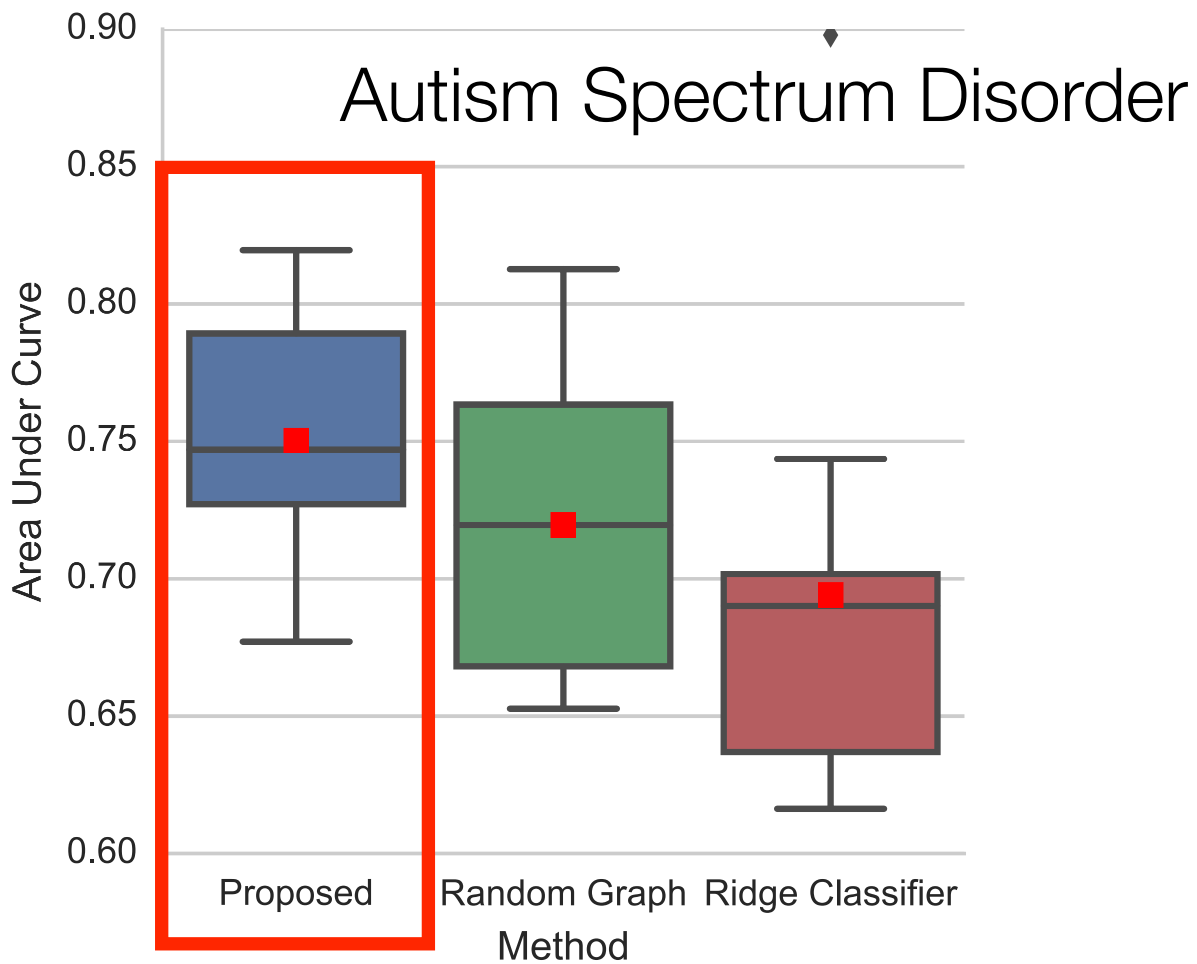


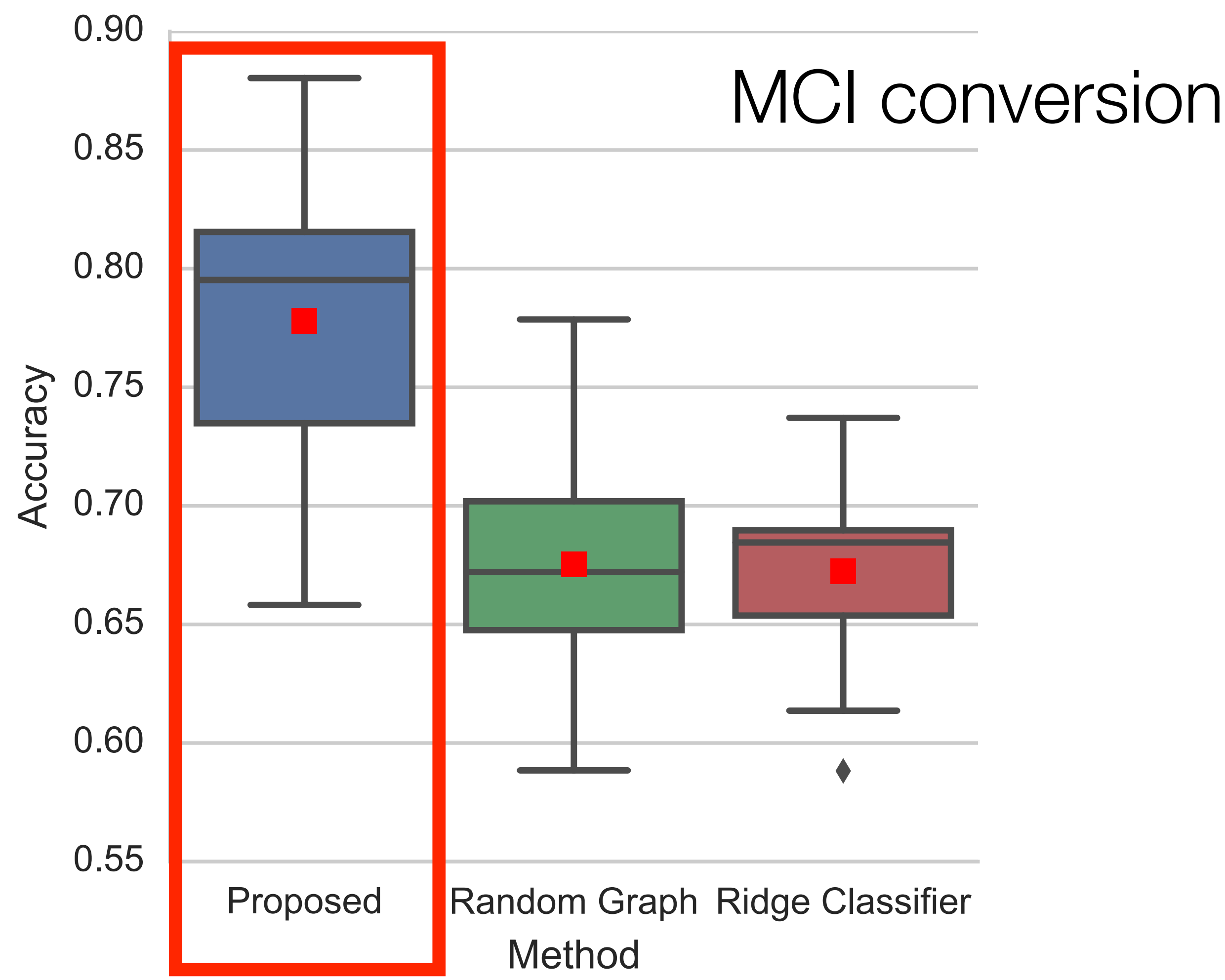




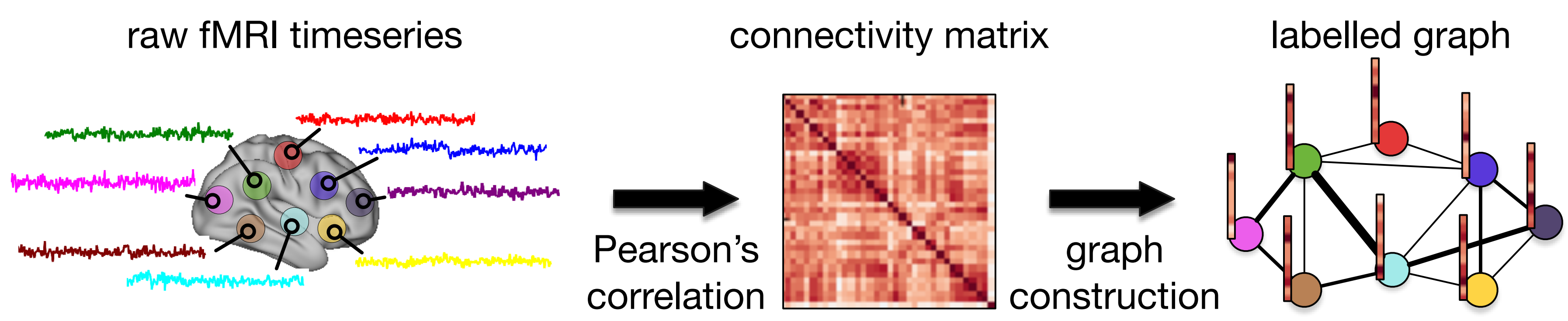
Semi-supervised node classification

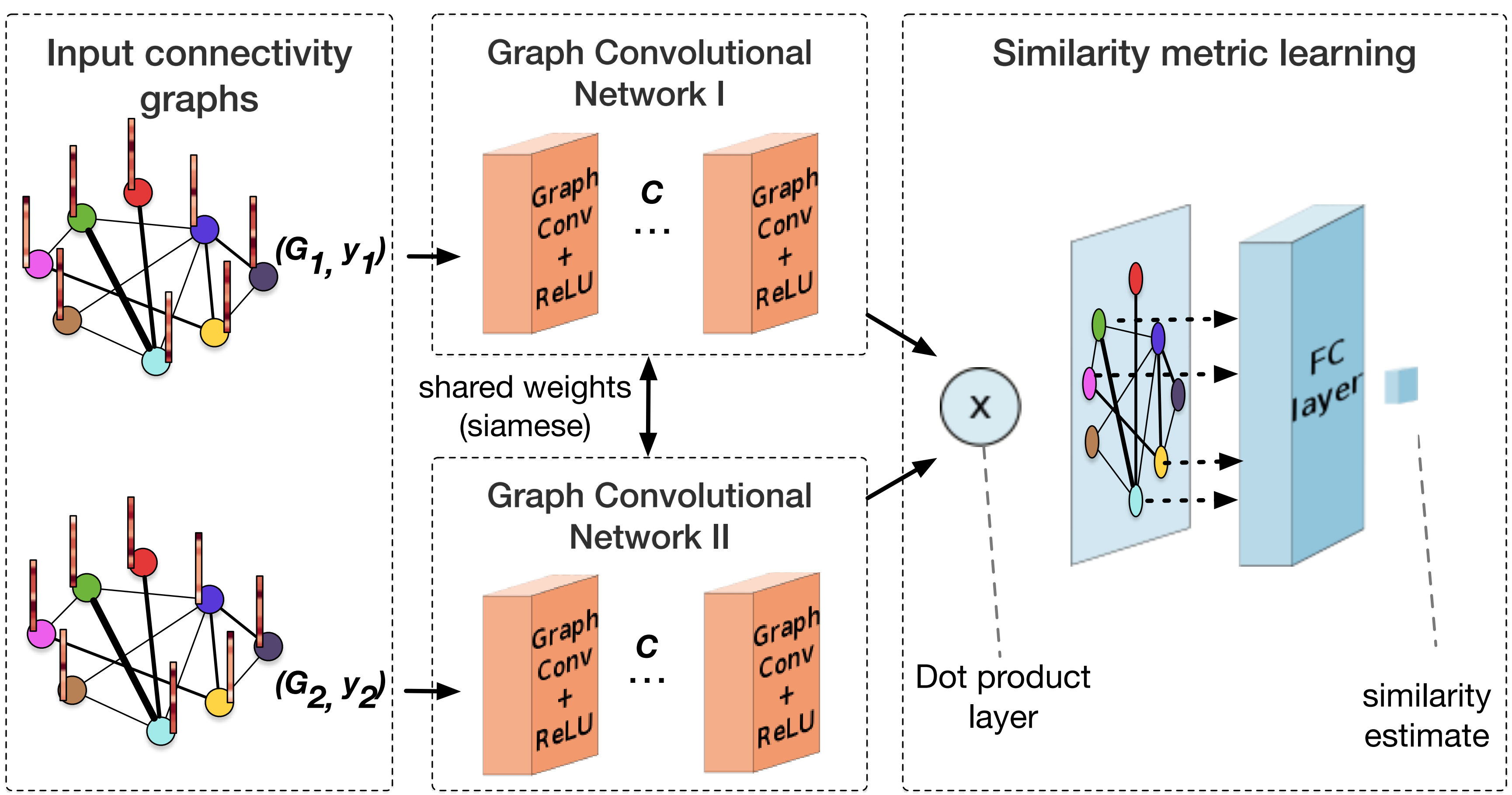




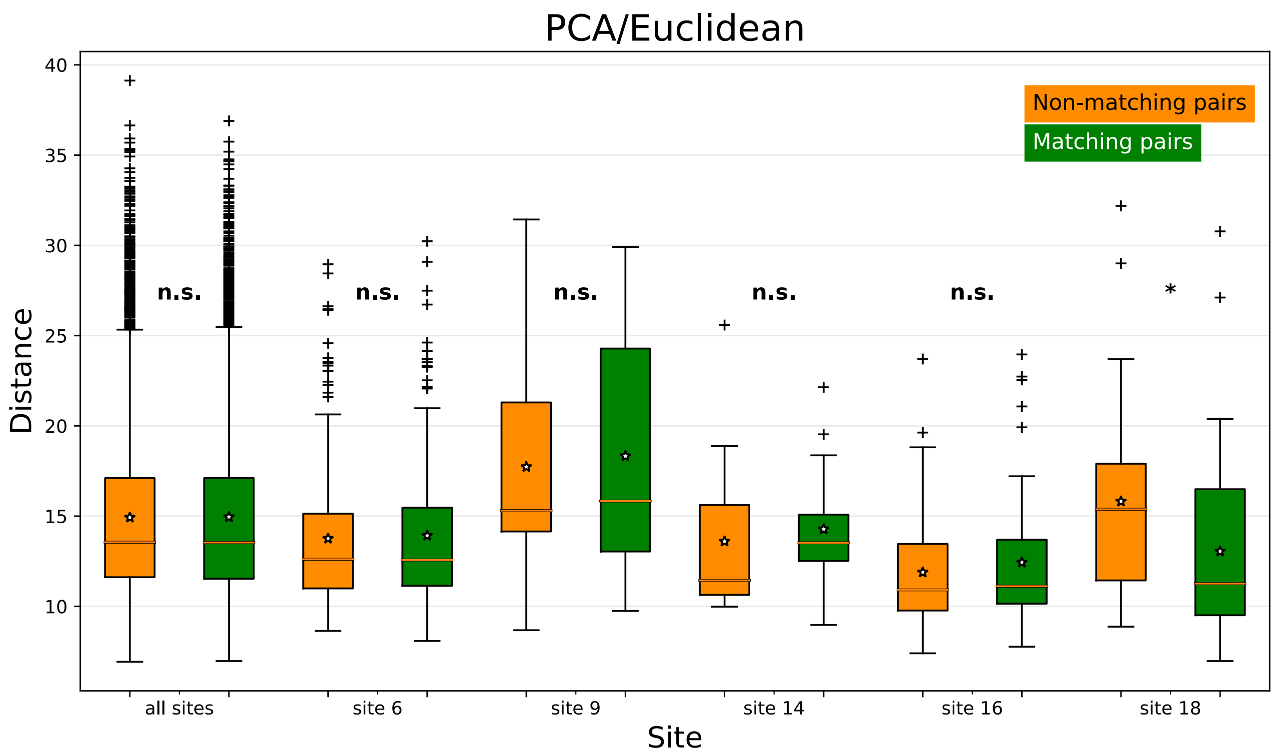


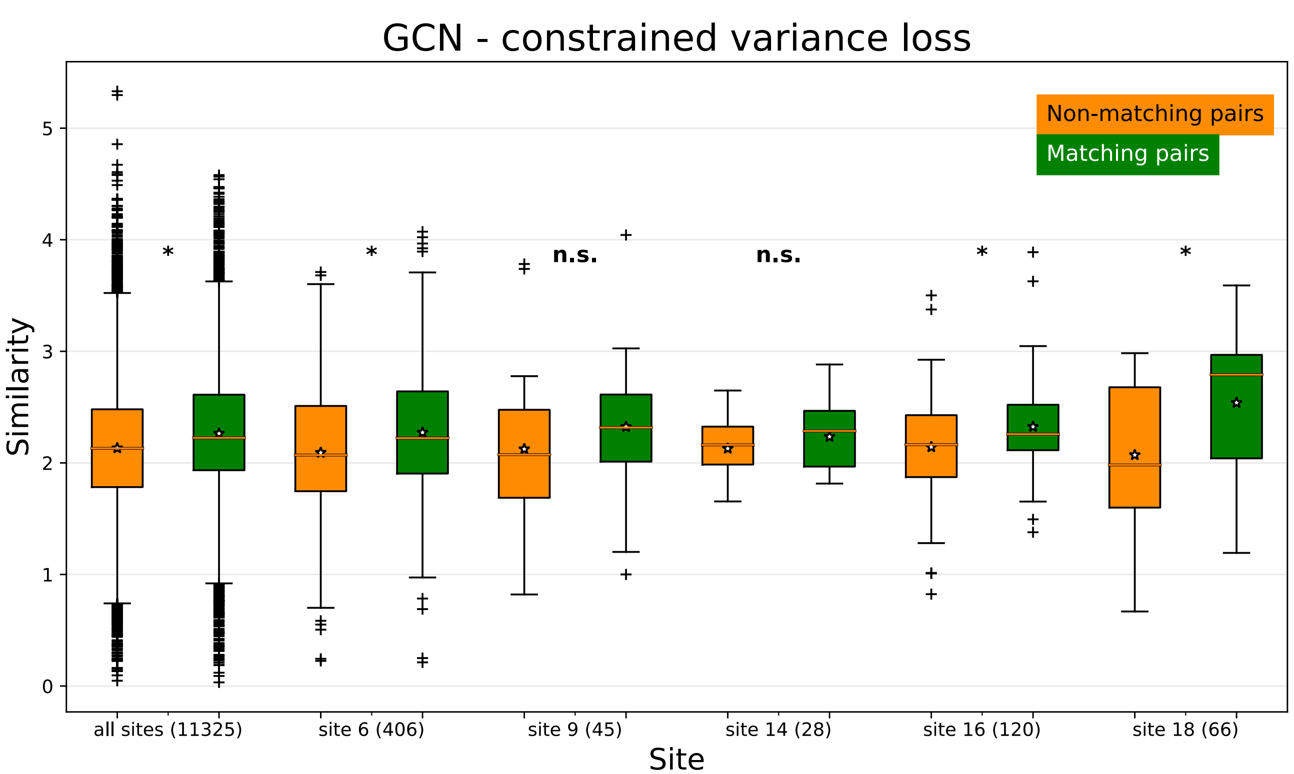
Distance metric learning

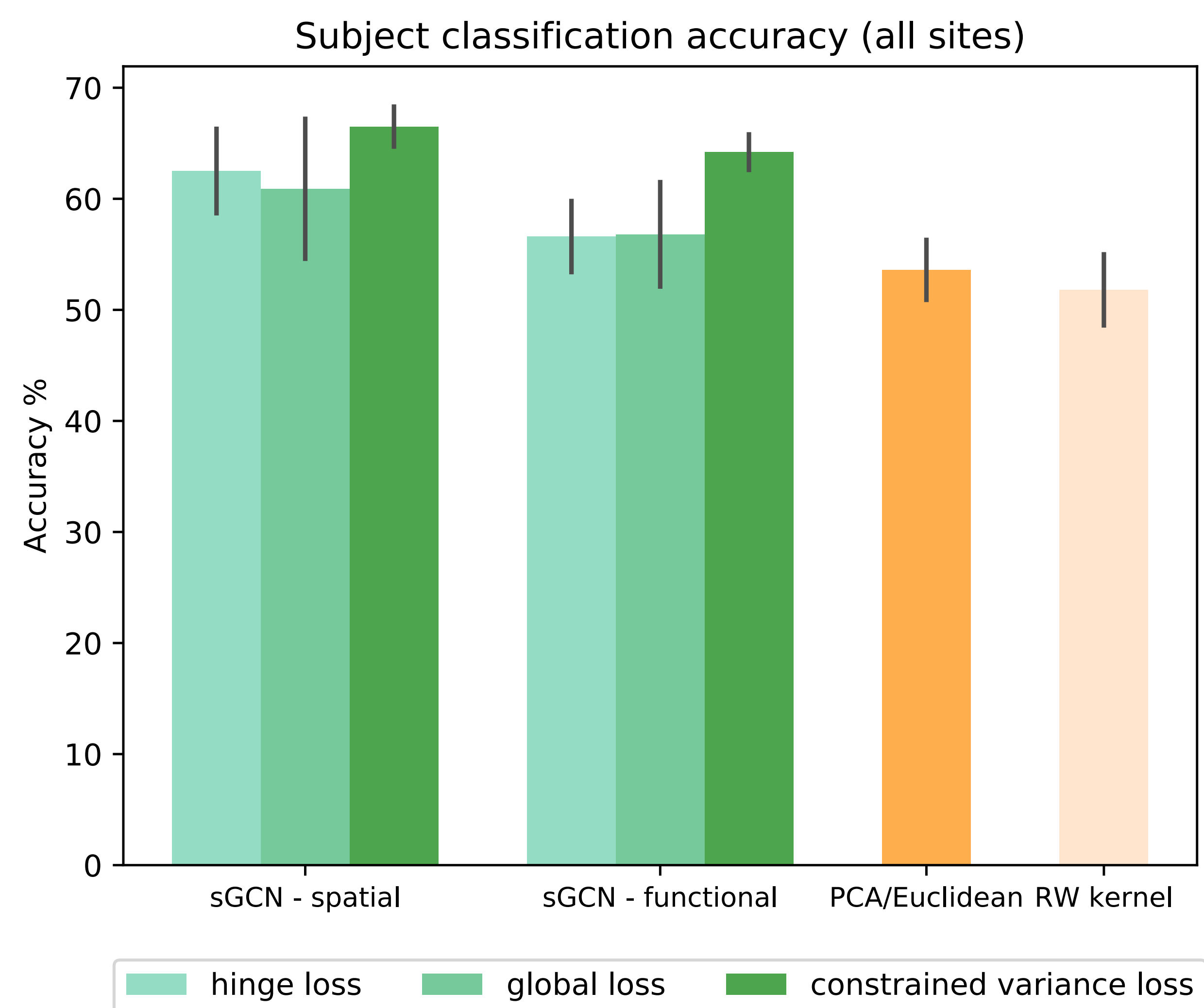




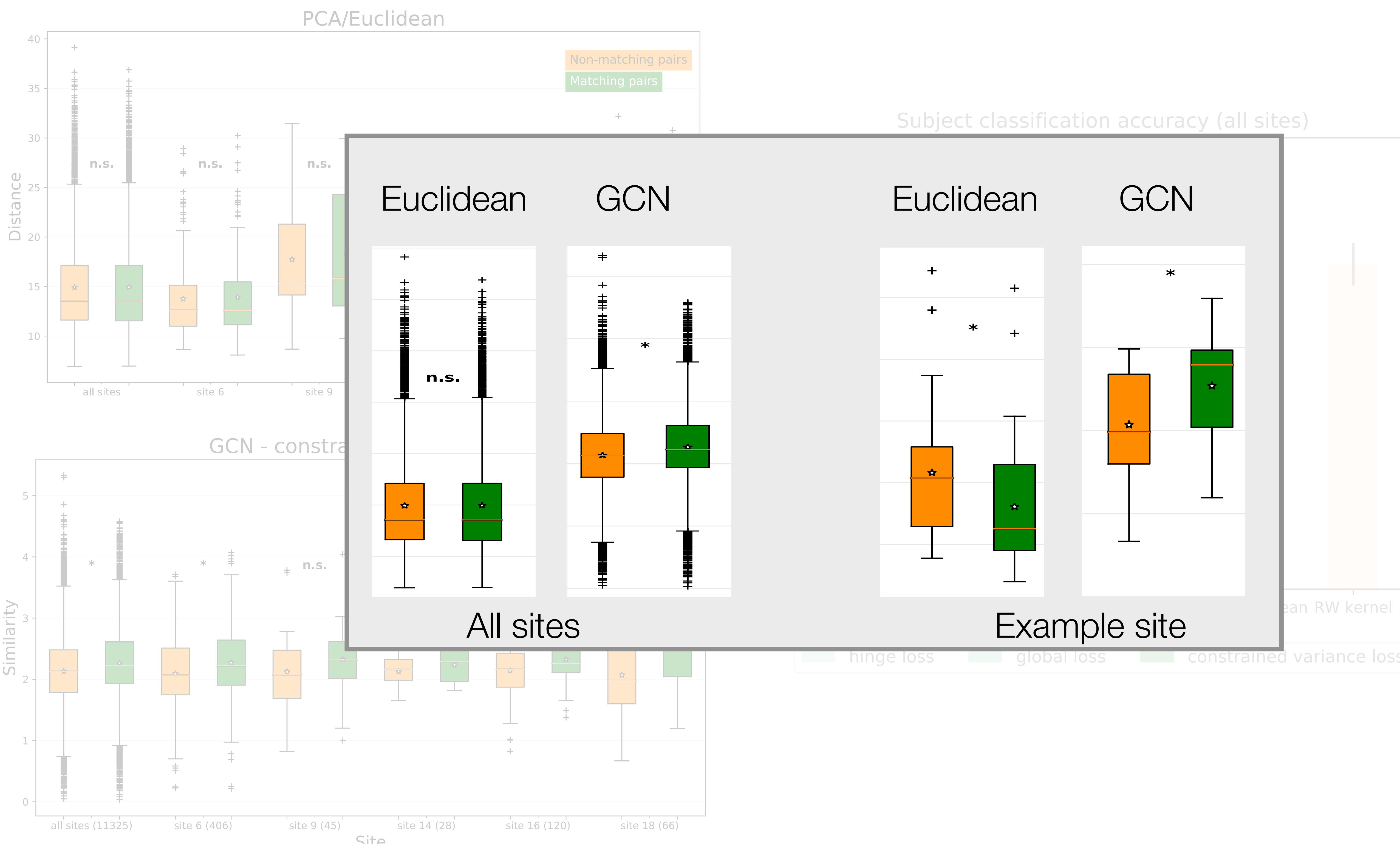
Distance metric learning: Results on ABIDE database







Distance metric learning: Results on ABIDE database



Conclusion

Conclusion

- Graph Convolutional Neural Networks

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains
  - Increasing popularity in the computer vision/machine learning fields (social, medical, graphics applications etc.)

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains
  - Increasing popularity in the computer vision/machine learning fields (social, medical, graphics applications etc.)
  - Very recent concept, many research opportunities

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains
  - Increasing popularity in the computer vision/machine learning fields (social, medical, graphics applications etc.)
  - Very recent concept, many research opportunities
- Which model to use?

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains
  - Increasing popularity in the computer vision/machine learning fields (social, medical, graphics applications etc.)
  - Very recent concept, many research opportunities
- Which model to use?
  - Fixed graph: spectral CNNs

Conclusion

- Graph Convolutional Neural Networks
  - Adapt the concept of CNNs to signals defined on non-Euclidean domains
  - Increasing popularity in the computer vision/machine learning fields (social, medical, graphics applications etc.)
  - Very recent concept, many research opportunities
- Which model to use?
  - Fixed graph: spectral CNNs
  - Variable graph, meshes: spatial CNNs

Acknowledgements



Questions?