Geometric deep learning with graphs: Linking the classification performance of Graph Convolutional Networks with the alignment of graphs and features

Yifan Qian [1] Paul Expert [2] Pietro Panzarasa [1] Mauricio Barahona [2]

[1]School of Business and Management,
Queen Mary University of London, London, UK

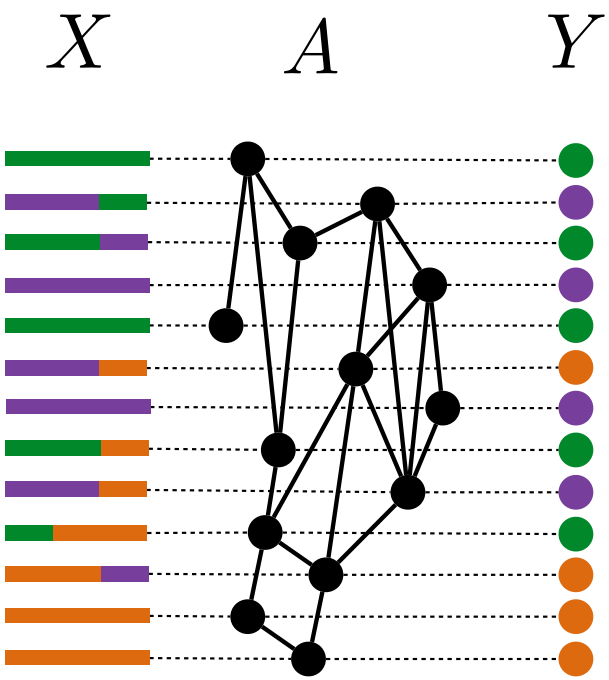
[2]Department of Mathematics,
Imperial College London, London, UK

2nd UK Network Science Workshop
University of Greenwich, London

26th October 2018

Background

Graph Convolutional Networks (GCNs)
(Kipf and Welling, 2017)

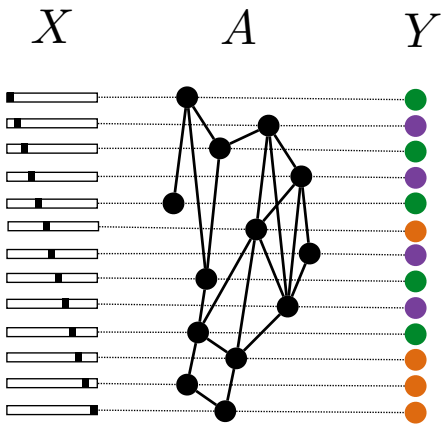


- a feature matrix $X \in R^{N \times D}$.
- an adjacency matrix $A \in R^{N \times N}$.
- a ground truth assignment matrix $Y \in R^{N \times C}$.

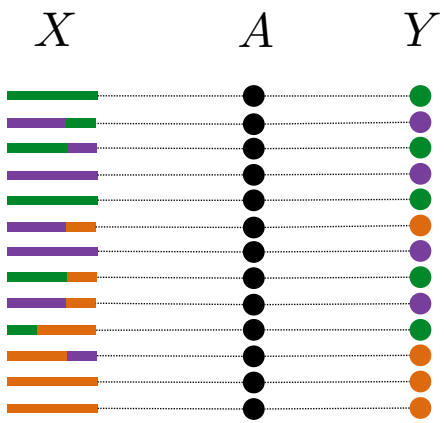
Motivation

Can additional information from the graph always be beneficial to the performance of GCN?

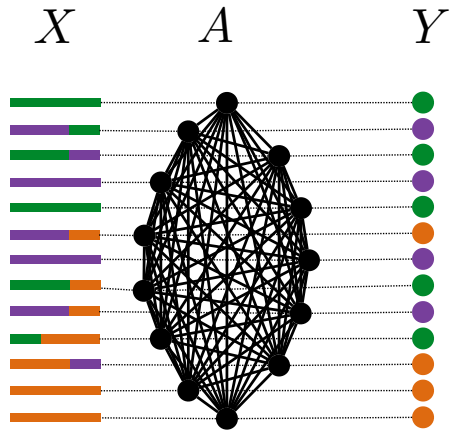
We consider three limit cases of GCN:



No features
(Identity matrix)



No graph



Complete graph

Motivation (continued)

Data sets	Nodes	Edges	Classes	Features
CORA	2,485	5,069	7	1,433
Wikipedia	20,525	215,056	12	100

Cases	**CORA**	**Wikipedia**
GCN	0.811± 0.005	0.392 ± 0.010
No features	0.691 ± 0.006	0.254 ± 0.037
No graph	0.548 ± 0.014	0.450 ± 0.007
Complete graph	0.121 ± 0.066	O.O.M.

Information from the graph can potentially increase the performance of GCN (e.g., CORA), but this is not always the case (e.g., Wikipedia)!

Hypothesis and goal

- **Hypothesis**:

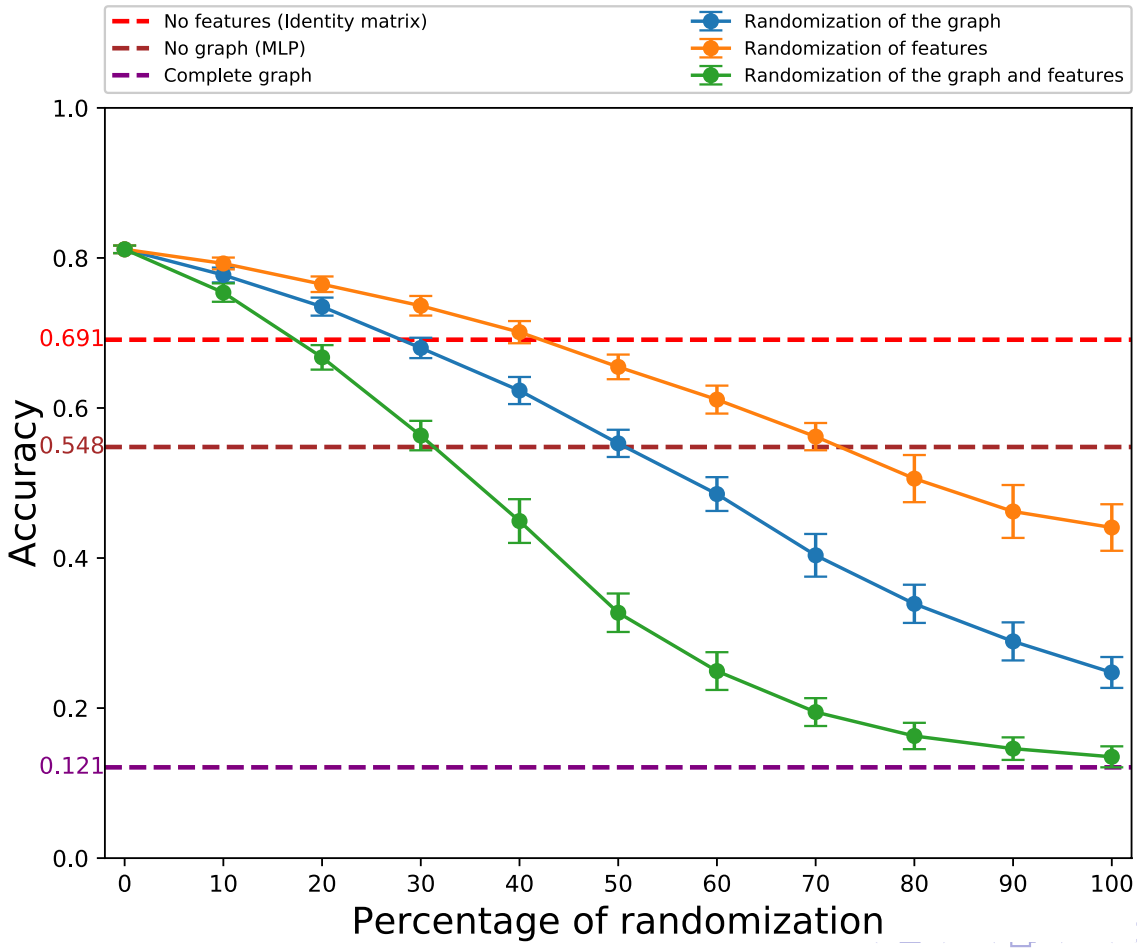
  A certain degree of alignment among X, A and Y is needed to obtain good performance of GCN, and any degradation in the information content leads to worsened performance.

- **Goal**:

  Linking the classification performance of GCN with the alignment of X, A and Y.

Randomization: Testing the hypothesis on CORA

- Randomizing the graph (by rewiring edges while keeping the degree distribution unchanged).
- Randomizing the features (by swapping the feature vectors at random).



Quantifying the alignment among X, A and Y (ongoing work)

Proposing a synthetic measure of alignment based on graph diffusion distance (gdd).

- Building two graphs from X and Y.
- Computing graph diffusion distance: $d_{gdd}(X, A)$,$d_{gdd}(X, Y)$ and $d_{gdd}(A, Y)$.
- A synthetic measure on the three pairwise distances.