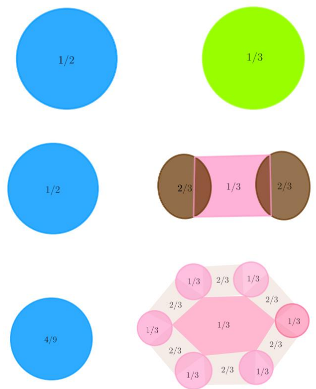
Deep Neural Networks on Random Graphs

Huy Pham - Stanford University

Problem

We are interested in classifying different random graph ensembles. The edges of the graphs are independent Bernoulli variables, with rate determined by some global structure of the graphs. Understanding these global structures are crucial, for example, in analysis of large networks. We consider classifying three pairs of ensembles with varying difficulty.



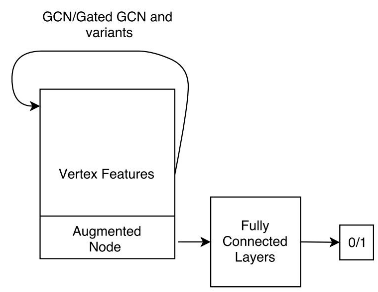
All graphs are encoded via the adjacency matrices A. A crucial property of graphs that distinguishes them from normal 2D-matrices is that graphs are invariant under permutation of vertices.

The graph convolutional models operate on vertex features. To translate to this setting, we initialize the vertex features of each node to be a vector with all entries being 1. We call the feature matrix X.

Augmented node

To extract information from the iterative graph convolutions, we augment the graph from the beginning with a node that is connected to all other nodes. We then use the feature vector from this node after propagating through the convolutions to classify the graph based on a Fully Connected Network.

Models



Graph convolution

The graph convolution is

$$X_{i+1} = \sigma(\tilde{A} X_i H)$$
$$X_1 = X$$

Inspired by LSTM networks, we consider the gated graph convolution,

$$X_{i+1} = c_i \circ \sigma(\tilde{A} X_i H) + (1 - c_i) \circ X_i$$
$$c_i = \sigma'(\tilde{A} X_i G)$$
$$X_1 = X$$

We also consider another variant of the convolutional network performed on graphs, given by

$$X_{i+1} = \sigma((H \circ A) X_i)$$
$$X_1 = X$$

A filtered version of this is

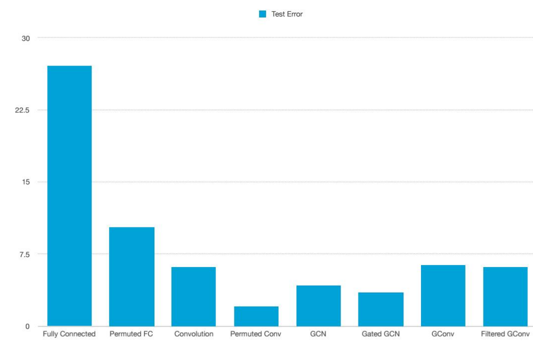
$$X_{i+1} = \sigma((H \circ G_i \circ A) X_i)$$
$$G_i = \sigma'(A Q_A + X_i Q_X)$$

We use the same parameters for the convolutions (as in recurrent neural networks). We use ReLU for σ and sigmoid for σ'.

Permutations

We also train Fully Connected Networks and Convolutional Networks on the same tasks. As an analogue of dropout, we perform a permutation on the graph adjacency matrices before each training epoch. This has a tremendous effect on the ability of the networks to generalize.

Results



Learning across ensembles: We train a GCN on the second ensemble, then use the virtual node feature as an embedding of the graph and retrain the Fully Connected layers on top of this for the third ensemble.

Algorithms	GCN	Gated GCN	GConv	Filtered GConv
Test Error	5.25	6.30	29.30	28.65

References

1. *Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).*

2. *Li, Yujia, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. "Gated graph sequence neural networks." arXiv preprint arXiv:1511.05493 (2015).*

3. *Pham, Trang, Truyen Tran, Hoa Dam, and Svetha Venkatesh. "Graph Classification via Deep Learning with Virtual Nodes." arXiv preprint arXiv:1708.04357 (2017).*