What functions can GNNs compute on random graphs? The role of Positional Encodings

Nicolas Keriven — CNRS, IRISA, Rennes
Samuel Vaiter — CNRS, LJAD, Nice

I - Summary

On large random graphs, GNNs *converge* to prediction functions on some latent space to label nodes. But:

1. the approximation power of these function spaces, and
2. the role of input node features,

are not well-understood.

In this paper, we fully characterize the function space that GNNs can approximate. We study the role of input node features, and in particular of augmenting them with Positional Encodings.

- **Approximation theorem** is natural but not trivial: novel result for ReLU-MLP universality in L^2 norm.
- Novel **concentration result** for ReLU-filtered random graphs of independent interest.
- The theory yields **normalization strategies** for PEs to be consistent across graph sizes, which improves results in practice.

II - Settings and Notations

- **Latent Space Random Graphs**

$$x_i \overset{iid}{\sim} P \in \mathcal{P}(\mathbb{R}^p), \quad a_{ij} \sim \text{Bernoulli}(\alpha_n w(x_i, x_j)) \quad 1 \leqslant i < j \leqslant n$$

  - distribution P with compact support, continuous connectivity kernel w, sparsity factor $\alpha_n \gtrsim \log n/n$
- **Graph matrix** $S \in \mathbb{R}^{n\times n}$, assumed to converge to **graph operator S**.
  - **Ex 1**: normalized adjacency matrix $S = A/(n\alpha_n)$, operator $\mathbf{S}f(x) = \int w(x,z)f(z)dP(z)$
  - **Ex 2**: normalized Laplacian $S = D_A^{-\frac{1}{2}} A D_A^{-\frac{1}{2}}$, operator $\mathbf{S}f(x) = \int \frac{w(x,z)}{\sqrt{d(x)d(z)}} f(z)dP(z)$

Assumption: denoting $\iota_X f = [f(x_i)]_{i=1}^n$,

$$n^{-1}\|S\iota_X f - \iota_X \mathbf{S} f\|_F^2 \xrightarrow[n\to\infty]{\mathbb{P}} 0$$

- **Prop**: True for the two examples above.

- **Graph Neural Networks** for node prediction, using S and ReLU:

$$Z^{(\ell+1)} = \text{ReLU}\left(Z^{(\ell)}\theta_0^{(\ell)} + SZ^{(\ell)}\theta_1^{(\ell)} + 1_n (b^{(\ell)})^\top\right), \quad \Phi_\theta(S, Z^{(0)}) = Z^{(L)}\theta^{(L)} + 1_n (b^{(L)})^\top$$

III - Function approximation

- It has been shown [1] that the output of GNNs on large random graphs is close to a **prediction function on the latent variables**. **How can we characterize precisely the functions well-approximated by GNNs?**

Definition: Given a **base space** $\mathcal{B}$ of input functions, the set of functions **well-approximated** by GNNs is:

$$\mathcal{F}_{\text{GNN}}(\mathcal{B}) = \left\{ f \mid \forall \epsilon, \exists \theta, \exists f^{(0)} \in \mathcal{B}, \ \mathbb{P}\left(n^{-1}\|\Phi_\theta(S, \iota_X f^{(0)}) - \iota_X f\|_F^2 \geqslant \epsilon \right) \to 0 \right\}$$

  - θ depends on ϵ! This is an **approximation** notion (**not** simply convergence)

- What can a GNN do? Structurally, two things: apply S (which converges to $\mathbf{S}$), and compute MLPs (which can approximate any function).

Definition: the $\mathbf{S}$-extension $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$ of $\mathcal{B}$ is defined by the following rules:

1. $\mathcal{B} \subset \mathcal{F}_{\mathbf{S}}(\mathcal{B})$
2. $\forall f \in \mathcal{F}_{\mathbf{S}}(\mathcal{B})$, $\mathbf{S}f \in \mathcal{F}_{\mathbf{S}}(\mathcal{B})$
3. $\forall f \in \mathcal{F}_{\mathbf{S}}(\mathcal{B})$, g Lip., $g \circ f \in \mathcal{F}_{\mathbf{S}}(\mathcal{B})$
4. $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$ is a vector space
5. $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$ is closed
6. $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$ is minimal

Thm.

$$\mathcal{F}_{\text{GNN}}(\mathcal{B}) = \mathcal{F}_{\mathbf{S}}(\mathcal{B})$$

The result is natural, the proof not so trivial!

- Need to prove *both* sides of the inclusion
- L^2 norm (convergence of GNNs) and L^∞ norm (universality of MLPs) do not mix well: need **new, specialized universality theorem for ReLU MLPs in L^2**

IV - Consequences and input $\mathcal{B}$

- We need only focus on $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$ to characterize the functions computed by GNNs. The inputs to the GNNs $\mathcal{B}$ is key.

Node features

1. if $Z^{(0)} = \iota_X f^{(0)}$, then $\mathcal{B} = \{f^{(0)}\}$

2a. if $Z^{(0)} = \iota_X f^{(0)} + \nu$ with ν_i centered i.i.d., the noise doesn't vanish (imperfect approximation).

2b. but (e.g.) $Z^{(0)} = S(\iota_X f^{(0)} + \xi)$ restores $\mathcal{B} = \{\mathbf{S}f^{(0)}\}$

No node features?

1. Constant $Z^{(0)} = 1_n$ yields $\mathcal{B} = \{1\}$
2. Degrees $Z^{(0)} = S1_n$ yields $\mathcal{B} = \{\mathbf{S}1\}$, but same $\mathcal{F}_{\mathbf{S}}(\mathcal{B})$
3. More recently, **"Positional Encodings"** (PE) [2]:

$$Z^{(0)} = \text{PE}_\gamma(S, Z)$$

   - potentially considering existing node features Z (often simple concatenation)
   - Ex: Spectral, random walk... need to be **permutation equivariant** $\text{PE}_\gamma(\pi S \pi^\top, \pi Z) = \pi \text{PE}_\gamma(S, Z)$

V - Spectral PEs and SignNet

Take $u_1^S, u_2^S \ldots$ the eigenvectors of S by decreasing eigenvalues. SignNet [2] is a type of spectral-PEs insensitive to sign ambiguity:

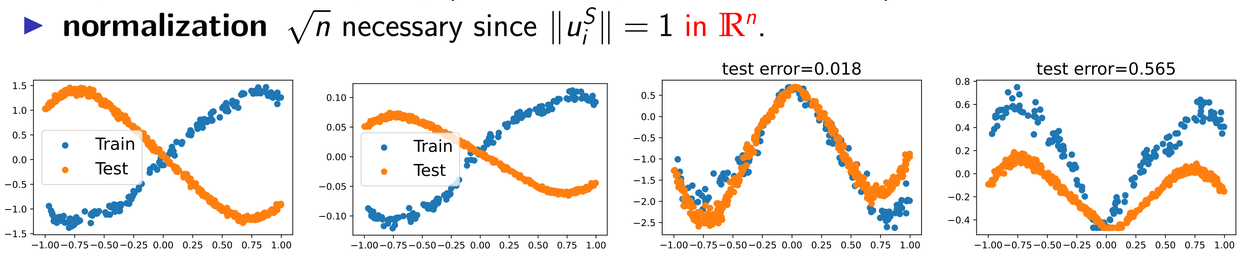
$$\text{PE}_\gamma(S) = \left[\text{MLP}_{\gamma_i}(\sqrt{n}u_i^S) + \text{MLP}_{\gamma_i}(-\sqrt{n}u_i^S)\right]_{i=1}^q \in \mathbb{R}^{n\times q}$$

It is known that eigenvectors often converge to the **eigenfunctions** $u_i^{\mathbf{S}}$ of $\mathbf{S}$:

Thm: for a p.s.d. kernel w or SBM random graphs, SignNet yields

$$\mathcal{B}_{\text{PE}} = \left\{ [f_i \circ u_i^{\mathbf{S}} + f_i \circ (-u_i^{\mathbf{S}})]_{i=1}^q \mid f_i \text{ continuous} \right\}$$

- w psd or SBM for simplicity (not necessary, case-by-case basis)
- **normalization** $\sqrt{n}$ necessary since $\|u_i^S\| = 1$ in $\mathbb{R}^n$.



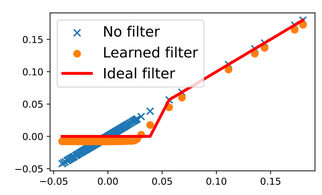
Left to right: before SignNet with normalization, without normalization, after SignNet idem.

VI - Distance-encoding PEs

Distance-encoding PEs [2] aggregate distances (random walk, shortest path, etc.). Many choices, here we take the **columns of S^k aggregated with DeepSet** (aka MLP + average):

$$\text{PE}_\gamma(S) = n^{-1} \sum_j \text{MLP}_\gamma(n\,[Se_j, \ldots, S^q e_j]) \in \mathbb{R}^{n\times q}$$

This does **not** converge! We need to use a **ReLU-MLP filter** on the eigenvalues of $S \leftarrow S_\xi = h_{\text{MLP}_\xi}(S)$ and **a novel concentration result for ReLU-filtered random graphs** (of independent interest).



Thm: for a p.s.d. kernel w or SBM random graphs, with $S = S_\xi$ above, distance-PE yields (with δ_x "Dirac" at x)

$$\mathcal{B}_{\text{PE}} = \left\{ \int f\left([\mathbf{S}\delta_x, \ldots, \mathbf{S}^q\delta_x]\right) dP(x) \mid f : \mathbb{R}^q \to \mathbb{R}^p \right\}$$

- $\mathcal{F}_{\mathbf{S}}(\mathcal{B}_{\text{PE}})$ is then sometimes universal! Generalization of [4]

VII - Other properties

- **Prop. GNNs are useful**: for all examples, there are cases where $\mathcal{B}_{\text{PE}} \subset \mathcal{F}_{\mathbf{S}}(\mathcal{B}_{\text{PE}})$ *strictly*
- **Prop. PEs are powerful**: for all examples, there are cases where $\mathcal{F}_{\mathbf{S}}(\{1\}) \subset \mathcal{F}_{\mathbf{S}}(\mathcal{B}_{\text{PE}})$ *strictly*

Normalization to be consistent across graph sizes is useful even on real data.

Dataset	Eigenvectors		Distance-encoding	
	w/ norm.	w/o norm.	w/ norm.	w/o norm.
`Synthetic`	68.61	65.59	67.31	62.49
`Synthetic (out-of-dist)`	67.87	62.51	66.80	63.33
`CiteSeer-subgraphs`	49.45	49.43	48.99	37.09
`IMDB-BINARY` (graph-classif.)	67.80	66.10	71.10	63.95
`COLLAB` (graph-classif.)	73.74	74.77	75.65	75.02

[1] Keriven, Bietti, Vaiter. **Convergence and Stability of Graph Convolutional Networks on Large Random Graphs**, *NeurIPS* 2020.

[2] Dwivedi et al. **Graph Neural Networks with Learnable Structural and Positional Representations**, *ICLR* 2022.

[3] Lim et al. **Sign and Basis Invariant Networks for Spectral Graph Representation Learning**, *ICLR* 2022.

[4] Keriven, Bietti, Vaiter. **On the Universality of Graph Neural Networks on Large Random Graphs**. *NeurIPS* 2021.

firstname.lastname@cnrs.fr

This work was supported by ANR GRandMa ANR-21-CE23-0006 and ANR GraVa ANR-18-CE40-0005.