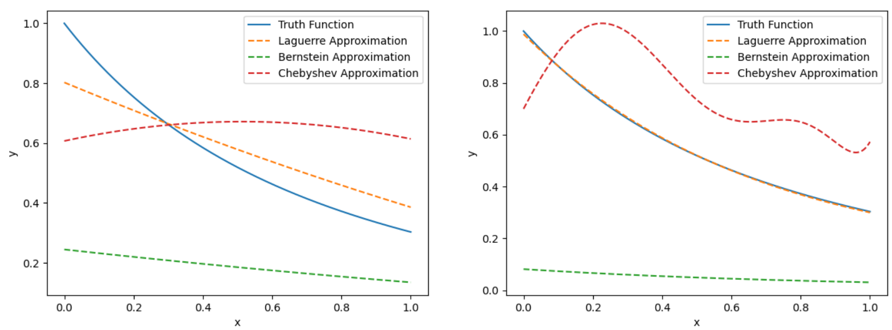
Response to the Comments

1. Approximation Error

To illustrate the approximation performance of Laguerre polynomials, we conducted experiments with an example discussed in [1]. As shown in Fig. 1, the truth function serves as a low-pass graph filter, which is similar to those employed in traditional GNN models like GCN. We conducted Laguerre, Chebyshev, and Bernstein polynomial approximations with varying order K. Fig. 1 clearly demonstrates that Laguerre polynomial approximation outperforms the others, consistent with the findings in Lemma 2.



(a) Order $K = 2$. (b) Order $K = 10$.

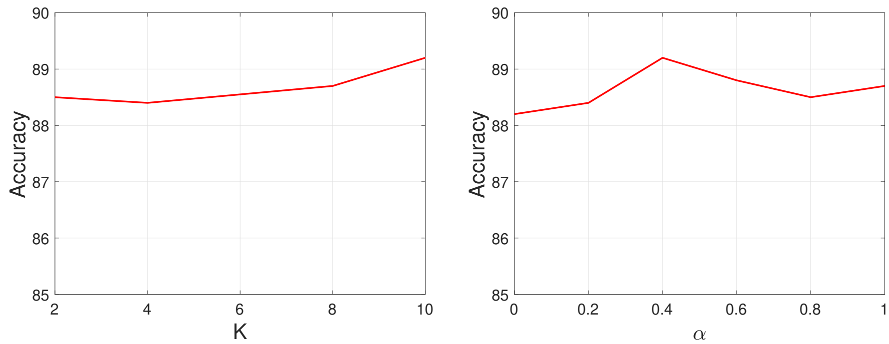
Figure 1: Approximation results with truth function $f(x) = e^{-2x/3}/(4+x^2)$.

2. Hyperparameter Analysis

In practical implementation, we conducted a grid search for the hyperparameters, selecting values for K from $\{2, 4, 8, 10\}$ and for α from $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$. The results of this hyperparameter analysis are summarized in Fig. 2. Notably, we observed that the performance of GLN exhibits robustness to variations in both K and α, indicating that the model's effectiveness is not highly sensitive to specific choices of these hyperparameters.

Additionally, the choice of $K = 10$ is relatively large compared to conventional methods such as GCN with $K = 2$ [2] and GAT with $K = 3$ [3]. With this choice, the approximation error of the Laguerre approximation becomes

$$e^{(K)} \sim C_2 e^{-3.32 C_1},$$



(a) Impact of the order on accuracy. (b) Impact of α on accuracy.

Figure 2: Hyper-parameter analysis.

which could be smaller than the approximation error of Chebyshev polynomials, i.e., $e^{(K)} \sim 0.22C_3$, with large choices of the constants.

3. Model Analysis

By parameterizing the integral with trainable parameters, GLN exhibits the ability to adaptively learn graph filters across diverse datasets. This adaptability is shown in Fig. 3, where we present some of the filters learned by GLN. On the homophilic dataset (Cora), GLN learns a low-pass-like filter, aligning with the observation that nodes with the same label tend to connect in homophilic datasets. In contrast, on the heterophilic dataset (Squirrel), GLN permits a more extensive passage of high-frequency information. This highlights GLN's capability to adaptively learn different filter behaviors based on dataset characteristics.

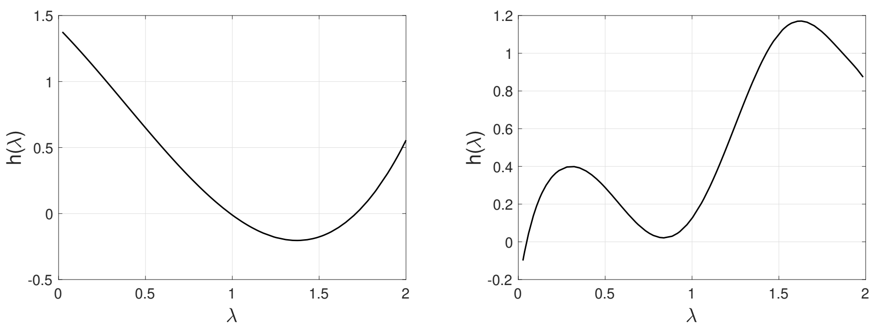
4. Proof of Theorem 1

The proof is given in the following.

Proof. With $\theta_1^{(\alpha)} > 0$, $\theta_0^{(\alpha)} > \alpha - 1$, $\theta_k^{(\alpha)} = 0$ for $k = 2, \cdots, K$, the graph filter of GLN becomes

$$h(\lambda) = \theta_0^{(\alpha)} + \theta_1^{(\alpha)}(1 + \alpha - \lambda) = -\theta_1^{(\alpha)}\lambda + \theta_0^{(\alpha)} + \theta_1^{(\alpha)}(1 + \alpha),$$

which is decreasing in $\lambda \in [0, 2]$ and thus behaves like a low-pass filter. Specifically, when $\theta_0^{(\alpha)} = \theta_1^{(\alpha)} = 1$, $\alpha = \theta_k^{(\alpha)} = 0$, for $k = 2, \cdots, K$, the



(a) Homophilic dataset. (b) Heterophilic dataset.

Figure 3: Filters learned from real-world datasets by GLN.

graph filter of GLN becomes

$$h(\lambda) = 2 - \lambda,$$

which is the same as GCN [2] and SGC [4]. Moreover, with $\theta_0^{(\alpha)} = 2\alpha \in [0, 2]$, $\theta_1^{(\alpha)} = 1$, $\theta_k^{(\alpha)} = 0$, for $k = 2, \cdots, K$, the graph filter of GLN becomes

$$h(\lambda) = 2\alpha + 1 + \alpha - \lambda = 2\underbrace{(\alpha + 1 - \lambda)}_{\text{low-pass}} + \underbrace{(\alpha - 1 + \lambda)}_{\text{low-pass}}.$$

In this way, GLN contains both low- and high-pass filters, and its propagation process reduces to that of FAGCN [5]. $\square$

References

[1] H. Wang, "Optimal convergence analysis of laguerre spectral approximations for analytic functions," *arXiv preprint arXiv:2304.05744*, 2023.

[2] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the 5th International Conference on Learning Representation*, 2017.

[3] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proceedings of the 6th International Conference on Learning Representations*, 2018.

[4] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in *International Conference on Machine Learning*, pp. 6861–6871, PMLR, 2019.

[5] D. Bo, X. Wang, C. Shi, and H. Shen, "Beyond low-frequency information in graph convolutional networks," in *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, 2021.