

Supplementary Material for: Graph Neural Networks (with Proper Weights) Can Escape Oversmoothing

Appendix A. Proofs of Main Results

In this section, we provide omitted proofs in the main paper.

A.1. Proof of Propostion 2

We first give some useful lemmas here, which would be used in the later proof.

Definition 9 *The square matrix A is irreducible if there is no permutation matrix P s.t.,*

$$PAP^{-1} \neq \begin{pmatrix} E & F \\ 0 & G \end{pmatrix}, \quad (16)$$

where E and G are non-trivial square matrices. Otherwise, we say A is reducible.

By combining Theorem I and Theorem II in [Herstein \(1954\)](#), we have the following lemma.

Lemma 10 *If $A \geq 0$ and $A^m > 0$ for some $m \in \mathbb{N}$, then A has a positive eigenvalue r such that*

- *if λ is any other eigenvalue of A , then $|\lambda| < r$ (thus $\rho(A) = r$),*
- *there exists a eigenvector x associated eigenvalue r such that $x > 0$.*

Lemma 11 (Lemma 3 in [Herstein \(1954\)](#)) *If an $N \times N$ matrix $A \geq 0$ and is irreducible, and $A_{ii} > 0$ for all $i \in [N]$, then $A^{N-1} > 0$.*

The following proposition is an extension of Theorem 1 in [Li et al. \(2018\)](#).

Proposition 12 *Under Assumption 1, for any $P \in \mathcal{P} = \{P \in \mathbb{R}^{N \times N} | \exists a > 0 \text{ s.t., } aP \geq A \text{ and } \rho(P) = 1\}$, P has eigenpair $(1, v)$ where all components of v are positive and any other eigenvalue λ of P satisfies $|\lambda| < 1$. Let λ_2 be the eigenvalue of P with the second maximal module. Then for any $x \in \mathbb{R}^N$, we have*

$$\lim_{k \rightarrow +\infty} P^k x = \theta v, \quad \text{for some } \theta \in \mathbb{R}, \quad (17)$$

$$\|P^k x - \theta v\|_2 = O(|\lambda_2|^k). \quad (18)$$

Proof [Proof of Proposition 12] Since there exists $a > 0$, $P \geq \frac{1}{a}A \geq 0$. It follows from Assumption 1 that $A_{ii} > 0$ for $i \in [N]$, hence $P_{ii} > 0$ for $i \in [N]$. Next, we prove that P is irreducible by contradiction. If P is reducible, there exists permutation matrix P_1 s.t.,

$$P_1 P P_1^{-1} \neq \begin{pmatrix} P_{11} & P_{12} \\ 0 & P_{22} \end{pmatrix}.$$

Combining $P \geq \frac{1}{c}A \geq 0$, we have

$$P_1AP_1^{-1} \neq \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix}.$$

Since A is a symmetric matrix and $P_1^{-1} = p_1^\top$, $P_1AP_1^{-1}$ is also a symmetric matrix. It follows that

$$P_1AP_1^{-1} \neq \begin{pmatrix} A_{11} & 0 \\ 0 & A_{22} \end{pmatrix}.$$

This means that graph \mathcal{G} can be divided into two parts with no edge to each other. It contradicts Assumption 1, thus P is irreducible. According to Lemma 10 and Lemma 11, P has dominant eigenpair $(1, v_1)$ where all components of v_1 are positive and any other eigenvalue λ of P satisfies $|\lambda| < 1$. Suppose $P = V\Lambda V^{-1}$ with $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_N\}$, $1 = \lambda_1 > |\lambda_2| \geq \dots, |\lambda_N|$ and $V = [v_1, v_2, \dots, v_N]$. If $x = \sum_{i=1}^N \theta_i v_i$, then

$$P^k x = \theta_1 v_1 + \sum_{i=2}^N \theta_i \lambda_i^k v_i. \quad (19)$$

Therefore, we conclude that

$$\begin{aligned} \lim_{k \rightarrow +\infty} P^k x &= \theta_1 v_1, \\ \|P^k x - \theta_1 v_1\|_2 &= O(|\lambda_2|^k). \end{aligned}$$

■

Then we can give the proof of Proposition 2 in the following.

Proof [Proof of Proposition 2] Armed with Proposition 12, we can get the results directly.

■

A.2. Proof of Proposition 3.

The proof of Proposition 3 mainly follows the proof of Lemma 3 in Daneshmand et al. (2020) which is based on Lemma 15. So we first introduce some definitions and lemma in the following.

Definition 13 A set of $d \times d$ matrices T is contracting if there exists a sequence $\{M_n \in T\}_{n=1}^\infty$ such that $M_n / \|M_n\|$ converges to a rank one matrix.

Definition 14 The set $d \times d$ matrices T is strongly irreducible if there does not exist a finite family of proper linear subspaces $V_1, \dots, V_k \subset \mathbb{R}^d$ such that the union of these subspaces is invariant with respect to T i.e., $Mv \in V_1 \cup V_2 \cup \dots \cup V_k$ holds for $\forall v \in V_1$ or V_2 or \dots or V_k and $\forall M \in T$.

Lemma 15 (Theorem 3.1 in Bougerol et al. (2012)) Let W_1, W_2, \dots be random $d \times d$ matrices which drawn independently from a distribution \mathbb{P} . Let $B_n = \prod_{k=1}^n W_k$. If the support of \mathbb{P} is strongly irreducible and contracting, then any limit point of $\{B_n / \|B_n\|\}_{n=1}^\infty$ is a rank one matrix almost surely.

Then we can give the proof for Proposition 3 as follows:

Proof [Proof of Proposition 3] Without loss of generality, we assume that each independent and identically distributed element of the weight matrices is drawn from the standard Gaussian distribution $\mathcal{N}(0, 1)$ or the uniform distribution $\text{Uniform}[-1, 1]$. And the supports of two distributions are $T_G = \mathbb{R}^{d \times d}$ and $T_U = [-1, -1]^{d \times d}$, respectively. Since $T_U \subset T_G$, we only need to prove that T_U is contracting and strongly irreducible.

First, we prove that T_U is contracting. Since $M_n = \mathbf{1}\mathbf{1}^\top \in T_U, n = 1, 2, \dots$, and the limit of $\{\frac{M_n}{\|M_n\|}\}_{n=1}^\infty$ is a rank one matrix, we have that T_U is contracting.

For strongly irreducible. For any finite family of proper linear subspaces $V_1, \dots, V_k \subset \mathbb{R}^d$, suppose $v \in V_1$ such that $\|v\| = 1$. Let e_i be the i -th standard basis in $\mathbb{R}^d, i = 1, 2, \dots, d$. Define $M_i = e_i v^\top, i = 1, 2, \dots, d$. Obviously, $M_i \in T_U$ and $M_i v = e_i$. Hence, $M_i v_{i=1}^d = e_{i=1}^d$ is not contained in any union of the finite proper linear subspace of V_1, \dots, V_k for $k < d$. And this follows that T_U is strongly irreducible.

Combining Lemma 15, we finite the proof. ■

Remark. It is easy to verify that for any $\alpha \in (0, 1]$, $(I - \alpha L_{sym}), (I - \alpha L_{rw}) \in \mathcal{P}$ which mentioned by Theorem 1 in Li et al. (2018). $(I - \alpha L_{sym})$ has dominant eigenpair $(1, D^{\frac{1}{2}}\mathbf{1})$ and $(I - \alpha L_{rw})$ has dominant eigenpair $(1, \mathbf{1})$. And the aggregation matrix in GAT also belongs to \mathcal{P} and has dominant eigenpair $(1, \mathbf{1})$.

A.3. Proof of Theorem 4

Proof Assume that P has eigendecomposition $P = V\Lambda V^{-1}$, where $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_N\}$, $|\lambda_1| > |\lambda_2| \geq \dots, |\lambda_N|$ and $V = [v_1, v_2, \dots, v_N]$. Without loss of generality, we assume that $\text{Rank}(X) = d < n, |\lambda_d| > 0$ and $X = V_d Q$, where $V_d = [v_1, v_2, \dots, v_d]$ and Q is a nonsingular matrix. Let $\Lambda_d = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_d\}$,

$$W^{(1)} = Q^{-1}\Lambda_d^{-1}, W^{(k)} = \Lambda_d^{-1}, k \geq 2. \quad (20)$$

Then we have the following:

$$X^{(1)} = X, X^{(k)} = V_d, \forall k \geq 2. \quad (21)$$

It follows that columns of $X^{(k)}$ are linearly independent for any $k \geq 2$. Therefore, there exists $a > 0$ such that Expression (11) holds. ■

A.4. Proof of Proposition 6

Proof [Proof of Proposition 6] It is straightforward to verify that $X^{(k)\top} X^{(k)} = I_d, i.e., X^{(k)}$ is column orthogonal. Hence, there exists $a > 0$ s.t.,

$$E(X^{(k)}), E\left(\frac{X^{(k)}}{\|X^{(k)}\|_F}\right), \text{MAD}(X^{(k)}) > a. \quad \blacksquare$$

A.5. Proof of Proposition 8

We first give a lemma which is modified from Theorem 4 in [Zhuo et al. \(2023\)](#).

Lemma 16 *Let $X, Y \in \mathbb{R}^{N \times d}$ have singular values $\sigma_1^X \geq \sigma_2^X \geq \dots \geq \sigma_m^X > 0$ and $\sigma_1^Y \geq \sigma_2^Y \geq \dots \geq \sigma_m^Y > 0$ ($m = \min\{N, d\}$), respectively. Assume that $\frac{\sigma_i^Y}{\sigma_i^X}$ is monotonically increasing with index i , then we have $\text{Erank}(Y) \geq \text{Erank}(X)$. Further, if $\frac{\sigma_i^Y}{\sigma_i^X}$ are non-constant, the inequality holds strictly, i.e., $\text{Erank}(Y) > \text{Erank}(X)$.*

The proof for Proposition 8 is as follows:

Proof [Proof of Proposition 8] Suppose $\Sigma = \hat{X}^\top \hat{X} = U \Lambda U^\top$ and the singular value decomposition of \hat{X} is $\hat{X} = V \Sigma^{\hat{X}} U^\top$, where $\Sigma^{\hat{X}}$ is a diagonal matrix with singular values $\sigma_i^{\hat{X}} = \lambda_i^{1/2}, i = 1, 2, \dots, d$. $X = \hat{X} \Sigma_f$, it follows that the singular values of X are $\sigma_i^X = \lambda_i^{1/2+p}, i = 1, 2, \dots, d$. Therefore, with $-1/2 \leq p < 0$, $\sigma_i^X / \sigma_i^{\hat{X}} = \lambda_i^p, i = 1, 2, \dots, d$ are strictly monotonically increasing with index i . Lemma 16 tells us that

$$\text{Erank}(\hat{X}) > \text{Erank}(X).$$

■

Appendix B. Experiment Details

In this section, we provide details and hyperparameters for WeightRep.

B.1. Datasets Details

We evaluate the performance of our methods on nine benchmark datasets. And we use `torch_geometric.datasets` in PyTorch Geometric to load these datasets. Dataset statistics are summarized in Table 4.

- The citation network datasets Cora, CiteSeer, and Pubmed ([Yang et al., 2016](#)) are homophily graph datasets in which nodes represent documents and edges represent citation links.
- The Amazon datasets Computers and Photo ([Shchur et al., 2018](#)) are the Amazon co-purchase graph datasets in which nodes represent products and edges indicate two products frequently bought together.
- The Coauthor datasets CS and Physics ([Shchur et al., 2018](#)) are co-authorship graphs based on the Microsoft Academic Graph in which nodes represent authors, edges indicate two authors co-authored a paper, node features represent paper keywords for each author’s papers, and class labels indicate each author’s respective field of study.
- The Wikipedia graphs Chameleon and Squirrel ([Rozemberczki et al., 2021](#)) are heterophily datasets in which nodes represent web pages and edges indicate hyperlinks between two web pages.

Table 4: Statistics of graph datasets in this paper.

Dataset	#nodes	#edges	#classes	#features
Cora	2708	10556	7	1433
CiteSeer	3327	9104	6	3703
Pubmed	19717	88648	3	500
Computers	13752	491722	10	767
Photo	7650	238162	8	745
Coauthor CS	18333	163,788	15	6805
Coauthor Physics	34493	495924	5	8415
Chameleon	2277	36101	6	2325
Squirrel	5201	217073	5	2089

B.2. Parameter Settings

For all methods, we use standard GCNs as the backbone. We provide our parameter settings in Table 5.

Table 5: Training hyperparameters.

Config	optimizer	learning rate	weight decay	training epochs	hidden dimension	dropout rate
Value	Adam	$5e - 3$	$5e - 4$	200	32	0.6

Appendix C. Additional Results

C.1. Additional Comparison of GCN and MLP

Here, we add more experiments on CiteSeer and Pubmed datasets (see Figure 3).

C.2. Additional Results on GCN with 64 Hidden Dimensions

We conduct more experiments on GCN with 64 hidden dimensions. The results are shown in Table 6.

C.3. Different Calculation Methods of Spectral Decomposition

Note that one step in WeightRep is to get the spectral decomposition of the covariance matrix. We use the common linear algebra PyTorch library `torch.linalg` to implement this operator. There are two ways to do this, the eigenvalue decomposition (EIG) and the singular value decomposition (SVD). Then we compare the performances of different implementations in Table 7. Roughly speaking, the performance is not very sensitive to

Table 6: Node classification accuracies (%) for nine datasets. We use GCN as the backbone with various depths: 2, 4, 8, 16. The hidden dimension is set to **64**. Reported results are averaged over 5 runs. For every layer setting, the highest accuracy is in bold.

Datasets	model	#L=2	#L=4	#L=8	#L=16
Cora	GCN	82.05 \pm 0.44	78.47 \pm 1.11	18.18 \pm 6.46	7.97 \pm 2.60
	WeightRep	81.99 \pm 0.34	80.62 \pm 0.48	78.11 \pm 0.72	47.42 \pm 0.61
CiteSeer	GCN	69.19 \pm 0.43	58.71 \pm 2.08	19.65 \pm 0.00	19.3 \pm 0.71
	WeightRep	68.50 \pm 0.24	64.61 \pm 0.45	61.65 \pm 1.68	38.83 \pm 1.29
Pubmed	GCN	77.71 \pm 0.15	75.94 \pm 0.49	28.32 \pm 9.21	35.71 \pm 7.46
	WeightRep	77.48 \pm 0.21	75.92 \pm 0.32	74.82 \pm 0.52	73.72 \pm 1.11
Computers	GCN	80.74 \pm 0.42	45.32 \pm 6.56	33.39 \pm 19.29	37.20 \pm 24.27
	WeightRep	87.02 \pm 0.22	83.03 \pm 0.6	77.58 \pm 1.00	62.76 \pm 6.94
Photo	GCN	84.57 \pm 1.45	76.67 \pm 14.47	7.06 \pm 1.64	13.50 \pm 13.44
	WeightRep	92.57 \pm 0.09	91.35 \pm 0.78	89.07 \pm 0.25	76.37 \pm 5.18
Coauthor CS	GCN	93.57 \pm 0.10	88.91 \pm 0.33	2.58 \pm 2.32	3.74 \pm 2.84
	WeightRep	93.65 \pm 0.03	92.8 \pm 0.18	91.4 \pm 0.14	89.67 \pm 0.45
Coauthor Physics	GCN	95.68 \pm 0.13	96.02 \pm 0.30	90.88 \pm 6.39	84.3 \pm 0.00
	WeightRep	96.77 \pm 0.05	96.46 \pm 0.10	96.09 \pm 0.33	95.47 \pm 0.30
Chameleon	GCN	50.88 \pm 1.47	39.08 \pm 2.27	22.37 \pm 0.00	22.37 \pm 0.00
	WeightRep	64.61 \pm 0.58	57.98 \pm 0.67	48.29 \pm 0.76	36.58 \pm 0.87
Squirrel	GCN	29.70 \pm 0.44	19.31 \pm 0.00	19.31 \pm 0.00	19.31 \pm 0.00
	WeightRep	48.24 \pm 1.15	39.21 \pm 0.72	35.52 \pm 0.93	27.44 \pm 1.58

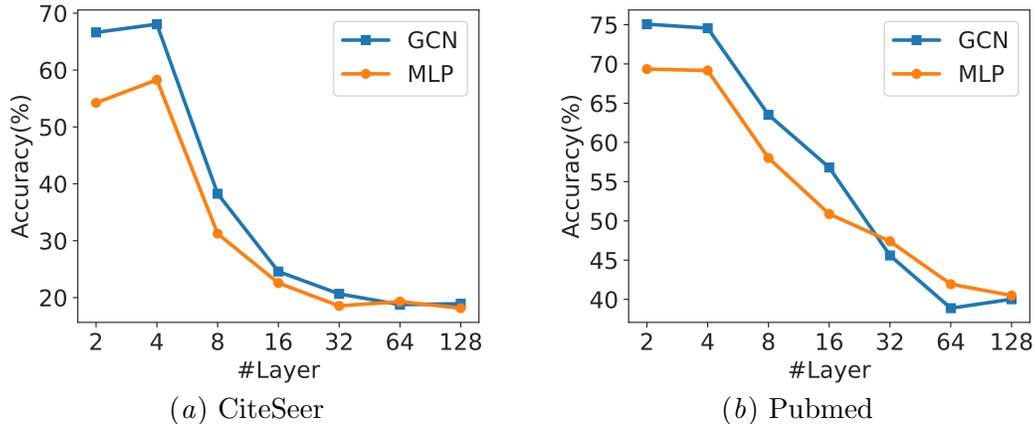


Figure 3: The impact of message aggregation and weight transformation when varying the depth of models on CiteSeer, and Pubmed datasets. MLP has the same structure as GCN except for the absence of message aggregation.

Table 7: Comparison of different calculation methods for the spectral decomposition. EIG and SVD are denoted the eigenvalue decomposition and the singular value decomposition, respectively.

Datasets	Method	#L=2	#L=4	#L=8	#L=16
Cora	EIG	81.79	80.70	77.85	22.01
	SVD	82.21	80.71	76.35	25.62
CiteSeer	EIG	68.61	65.36	58.15	31.56
	SVD	69.04	65.56	57.63	34.38
Pubmed	EIG	77.00	75.95	74.84	75.63
	SVD	77.68	75.58	75.23	73.41

the implementation ways. However, compared to the eigenvalue decomposition, the singular value decomposition performs better in most cases and is more numerically stable. Therefore, we use the singular value decomposition as our default implementation.

C.4. Expanded Table 2

We attach the expanded Table 2 in Table 8, which contains more datasets and standard deviation. It can be seen that our method is superior to other methods or comparable.

Table 8: Test accuracy (%) comparison of different normalizations. The highest accuracy is in bold and the second one is underlined. OOM: out of memory.

Datasets	Models	#L=2	#L=4	#L=8	#L=16
CiteSeer	GCN	68.96 ± 0.51	53.74 ± 4.13	19.65 ± 0.0	19.65 ± 0.0
	LayerNorm	64.89 ± 0.5	61.74 ± 0.81	25.4 ± 3.09	19.65 ± 0.0
	PairNorm	46.38 ± 1.93	40.17 ± 1.81	39.57 ± 2.04	24.09 ± 1.19
	ContraNorm	63.67 ± 0.65	60.69 ± 0.9	60.18 ± 0.22	43.63 ± 4.02
	WeightRep	69.04 ± 0.33	65.56 ± 0.24	57.63 ± 3.16	34.38 ± 2.72
Pubmed	GCN	77.94 ± 0.12	75.96 ± 0.51	20.8 ± 0.0	32.02 ± 9.16
	LayerNorm	75.04 ± 0.35	73.82 ± 0.61	43.51 ± 20.67	35.97 ± 7.59
	PairNorm	72.81 ± 0.76	67.99 ± 2.06	64.05 ± 1.96	72.27 ± 1.1
	ContraNorm	75.42 ± 0.39	OOM	OOM	OOM
	WeightRep	77.68 ± 0.12	75.58 ± 0.36	75.23 ± 0.8	73.41 ± 1.71
Coauthor CS	GCN	93.25 ± 0.33	72.3 ± 14.99	2.58 ± 2.32	1.42 ± 0.0
	LayerNorm	93.8 ± 0.14	92.14 ± 0.62	16.94 ± 12.51	1.42 ± 0.0
	PairNorm	93.4 ± 0.25	91.97 ± 0.62	85.25 ± 4.98	43.71 ± 21.94
	ContraNorm	93.73 ± 0.36	OOM	OOM	OOM
	WeightRep	93.81 ± 0.13	92.49 ± 0.07	91.24 ± 0.14	84.47 ± 0.57
Coauthor Physics	GCN	95.74 ± 0.1	95.97 ± 0.11	91.53 ± 3.71	84.3 ± 0.0
	LayerNorm	96.49 ± 0.33	96.82 ± 0.25	94.85 ± 3.33	84.3 ± 0.0
	PairNorm	96.5 ± 0.23	96.15 ± 0.2	95.54 ± 0.44	93.93 ± 0.36
	ContraNorm	OOM	OOM	OOM	OOM
	WeightRep	96.62 ± 0.09	96.15 ± 0.07	95.7 ± 0.12	94.76 ± 0.18
Chameleon	GCN	47.89 ± 1.5	36.89 ± 1.03	22.37 ± 0.0	22.37 ± 0.0
	LayerNorm	61.36 ± 1.19	51.75 ± 3.35	22.37 ± 0.0	22.37 ± 0.0
	PairNorm	63.11 ± 0.47	58.9 ± 1.34	48.86 ± 0.78	37.63 ± 2.83
	ContraNorm	64.91 ± 1.04	58.68 ± 2.23	46.89 ± 2.33	38.99 ± 1.72
	WeightRep	63.38 ± 0.95	59.42 ± 1.52	49.79 ± 1.76	35.22 ± 1.04
Squirrel	GCN	28.66 ± 0.83	19.31 ± 0.0	19.31 ± 0.0	19.31 ± 0.0
	LayerNorm	42.19 ± 1.37	25.65 ± 5.22	19.4 ± 0.19	19.31 ± 0.0
	PairNorm	43.61 ± 0.4	39.44 ± 0.51	35.41 ± 1.5	23.11 ± 2.6
	ContraNorm	45.48 ± 1.23	39.21 ± 0.79	35.16 ± 1.33	25.26 ± 2.08
	WeightRep	45.9 ± 0.44	39.47 ± 1.02	34.33 ± 0.77	26.07 ± 1.45

Table 9: The MAD metric for representation of each layer on 16-layer GNNs (higher is better).

Datasets	Method	#L=1	#L=3	#L=5	#L=7	#L=9	#L=11	#L=13	#L=15
Cora	GCN	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.15
	WeightRep	0.23	0.35	0.36	0.38	0.39	0.39	0.40	0.40
CiteSeer	GCN	0.02	0.00	0.00	0.00	0.00	0.00	0.02	0.14
	WeightRep	0.25	0.40	0.36	0.33	0.35	0.36	0.38	0.45
Pubmed	GCN	0.03	0.00	0.00	0.00	0.00	0.00	0.01	0.04
	WeightRep	0.35	0.39	0.38	0.42	0.40	0.43	0.48	0.48
Computers	GCN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
	WeightRep	0.17	0.16	0.24	0.26	0.20	0.20	0.26	0.29
Photo	GCN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05
	WeightRep	0.13	0.29	0.29	0.35	0.31	0.32	0.34	0.44
Coauthor CS	GCN	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.20
	WeightRep	0.36	0.46	0.44	0.46	0.47	0.47	0.49	0.49
Coauthor Physics	GCN	0.01	0.00	0.00	0.00	0.00	0.00	0.03	0.32
	WeightRep	0.35	0.39	0.43	0.45	0.49	0.47	0.49	0.46
Chameleon	GCN	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.01
	WeightRep	0.34	0.24	0.31	0.32	0.33	0.40	0.46	0.52
Squirrel	GCN	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	WeightRep	0.11	0.19	0.27	0.28	0.20	0.30	0.20	0.29

C.5. Quantify Oversmoothing by MAD

We add the MAD metric to quantify oversmoothing. The experiment was conducted on a 16-layer network, and we measured MAD at each layer. From the following table, we note that the MAD of WeightRep is much higher than GCN.

C.6. The Benefits of Deeper GNNs

From Table 1 and Table 6, one can see that increasing the depth does not lead to better performance. However, in the case that we need to extract multi-hop information to complete the prediction, the deeper graph neural network can extract multi-hop information to solve the problem. For example, deeper GNNs perform better than shallow ones in datasets with missing features which often happens in the real world, such as recommendation systems, and traffic prediction. Here we give a concrete example: we randomly remove $p\%$ the node features in validation and test set following the setting in Zhao and Akoglu (2020). Our experiments are concluded on Pubmed datasets. From the table, we can see that the

Table 10: Node classification accuracies (%) on Pubmed dataset with missing features. The missing percentage $p\%$ means we randomly remove $p\%$ the node features in validation and test set

Missing Percentage	Method	#L=2	#L=4	#L=8	#L16
50%	GCN	76.83 ± 0.11	75.59 ± 1.17	28.45 ± 9.36	35.71 ± 7.46
	WeightRep	76.83 ± 0.28	76.17 ± 0.43	75.31 ± 1.14	76.23 ± 0.88
100%	GCN	34.47 ± 9.0	39.96 ± 0.46	28.19 ± 9.05	32.27 ± 9.36
	WeightRep	35.26 ± 8.54	44.04 ± 6.68	45.7 ± 2.69	59.63 ± 3.35

performance increases with the depth under 100% missing percentage. Hence, increasing the depth will lead to better performance in some cases.