Clustering of Diverse Multiplex Networks

Responses to the Reviewer 1

1. (Novelty) The method for between-layer clustering is standard, and the algorithm for within-layer clustering is adopted from [6].

There must be some misunderstanding here. There is no standard method for the between-layer clustering in the DIMPLE model since **nobody** has studied the model so far.

The major difficulty of this paper seems to lie in the proof. Although the between-layer clustering algorithm is reasonable, the proof of Theorem 1 does not appear to be technically precise. The authors apply Proposition 1 of [1] to obtain inequality (40) on page 22. However, the correct inequality seems to be

$$\left\|\sin\Theta(\widehat{\mathbf{W}},\mathbf{W})\right\| \le \frac{\sigma_M(\widehat{\mathbf{\Theta}}\mathbf{W})\,\|\Pi_{\widehat{\mathbf{\Theta}}\mathbf{W}}\widehat{\mathbf{\Theta}}\mathbf{W}_\perp\|}{\sigma_M^2(\widehat{\mathbf{\Theta}}\mathbf{W})-\sigma_{M+1}^2(\widehat{\mathbf{\Theta}})}$$

Since, according to inequality (42), $\sigma_M(\widehat{\mathbf{\Theta}}\mathbf{W})$ is not vanishing, we cannot use this inequality to bound $\left\|\sin\Theta(\widehat{\mathbf{W}},\mathbf{W})\right\|$.

Thanks a lot for pointing this out. We were using the ArXiv version of the [1] paper by (since it had the supplement attached), and the paper contained the error which we repeated in the paper. We checked later in the published version and, indeed, the error was corrected, and the inequality looks exactly the way you cited it. Actually, this has never happened to me in decades of my professional life, so I did not bother to check. Will be more careful from now on!

In any case, we have corrected the error, and now have an upper bound for the between layer clustering error which is valid.

2. (Error bound) The within-layer error rate bounds (34) and (35) are not sharp enough. If there is only one group ($M = 1$), then we know that having more layers would be beneficial to reduce the within-layer error rate. However, the probability lower bound involves the term $-Ln^{-\tau}$, which diverges to $-\infty$ as L goes to ∞, and thus fails to provide an informative bound. The bound provided by [6] is of the form

$$\mathbb{P}\left\{R_{WL} \le C\left(\frac{1}{n}+\frac{\log(L+n)}{Ln^2\rho^2}\right)\right\} \ge 1-O\left(\frac{1}{L+n}\right)$$

which is more reasonable.

Please, take into account the difference between the model in [6] and the DIMPLE model. While, [6] allows **any** relationship between parameters, inference in the DIMPLE model can be done only if $n\rho_n \to \infty$. The latter means that $n \to \infty$ and, hence, for τ large enough, $Ln^{-\tau} \to 0$ under very mild restrictions on L. On the other hand, $O\left(\frac{1}{L+n}\right)$ may not be small enough for our purposes, due to the need of application of the union bound.

3. (Comparison) I am not sure that the suggested DIMPLE method gives great advantages compared to existing MMLSBM.

Perhaps, there is some misunderstanding here. DIMPLE is not a new algorithm, it is a **new model** which generalizes the MMLSBM and the setting in Lei and Lin (2021). As our simulations show, when matrices $B^{(l)}$ are all different, algorithms designed for MMLSBM do not do a good job. On the other hand, [6] consider the case where community assignments are the same in all layers, so there is no need to do any between-layer clustering.

In order to better address your concerns (and concerns of the second reviewer), we added a section to the paper, where we study application of our algorithms and ALMA algorithm designed for the MMLSBM to data, generated from both the DIMPLE model and the MMLSBM. As our simulations show, while our algorithms work in the case of the MMLSBM (although less efficiently since they "do not know" about additional structure), algorithms designed for the MMLSBM have poor performance, when they are applied to the DIMPLE model.

First, algorithms for the between-layer and the within-layer clustering do not show much difference from MMLSBM under assortative assumption (A6). The suggested DIMPLE algorithm averages the probability matrices $B^{(l)}$ within the same group and performs spectral clustering. However, this procedure is not identifiable in the case where all the probability matrices $B^{(l)}$ are identical in the same group (MMLSBM). From this identifiability issue, the output of Algorithm 2 under DIMPLE will give the same output under MMLSBM. Therefore, I am wondering how different clustering results from Algorithm 2 would be from TWIST [4].

The difference between our between-layer clustering algorithm and both ALMA and TWIST is that we **cannot** cluster the layers of the adjacency tensor. We are actually clustering the results of the SVD, specifically, the matrices $sgn(\mathbf{A}^{(l)})$ in [2]. So, the between-layer clustering algorithm is **the most novel part of our paper**. After the groups of layers are identified, the problem is reduced to the setting, where the community assignment propagates within the group of layers, and this is the setting which has been studied in a number of papers. So, Algorithm 2 is just an approximate K-means algorithm, applied to averages, which, according to [7], works well under assortativity assumption. We believe that the within-layer clustering algorithms of TWIST or ALMA also fall into this category (since assortativity assumption always holds for MMLSBM). Since assortativity does not hold automatically for the DIMPLE model, in the absence of Assumption **A6**, we use the technique described in [6].

Second, when we consider the estimation of block connection probability in every layer, the larger samples of layers do not improve the estimation accuracy in DIMPLE. As author mentioned, the probability matrix $B^{(l)}$ is estimated by averaging over the estimated community assignments. Consider we want to estimate $B^{(l)}$ such that $z(i) = k_1 \in [K]$, $z(j) = k_1 \in [K]$, and $c(l) = m \in [M]$ where $z : [n] \to [K]$ and $c : [L] \to [M]$ are clustering functions. Then, we have $n_{k_1,m}\, n_{k_2,m}$ number of samples to estimate $B^{(l)}_{i,j}$ under DIMPLE while $L_m\, n_{k_1,m}\, n_{k_2,m}$ under MMLSBM. Therefore, MMLSBM seems to be more advantageous in estimating the connection probability matrix especially when we have many layers, small number of nodes, or large number of communities.

Your calculation is completely correct. However, note that the source of the problem is not the deficiency of our method but the simple fact that there are M different matrices $B^{(l)}$ in the MMLSBM,

and $L >> M$ different matrices $B^{(l)}$ in the DIMPLE model. And if the MMLSBM does not hold, then, as our simulations show, algorithms designed for the MMLSBM do not do a good job.

4. (Rank) On page 8, it is not straightforward for me to conclude that rank($\mathbf{\Theta}$) = M. The matrix $\mathbf{\Theta}$ is determined by the membership matrix, and the latter only has a finite number of possible choices. So the full rankness cannot be obtained by a simple almost sure argument. It is also not clear at the first sight whether the vectorized projection matrices cannot be linearly dependent. Since the validity of the method relies on rank($\mathbf{\Theta}$) = M, I would like to see some justification.

Matrix $\mathbf{\Theta}$ has only M different columns since $\mathbf{\Theta}(:,l) = \text{vec}\left(\mathbf{U}_z^{(c(l))}(\mathbf{U}_z^{(c(l))})^T\right)$ where $c : [L] \to [M]$ is a clustering function. Therefore, rank($\mathbf{\Theta}$) $\leq M$. The fact that rank($\mathbf{\Theta}$) $= M$ follows from the fact that $\mathbf{\Theta} = (\overline{\mathbf{U}} \otimes \overline{\mathbf{U}})\mathbf{F}\mathbf{W}^T$ where $(\overline{\mathbf{U}} \otimes \overline{\mathbf{U}}) \in \mathcal{O}_{n^2,r^2}$, $\mathbf{W} \in \mathcal{O}(L, M)$ and matrix $\mathbf{F} \in \mathbb{R}^{r^2 \times M}$ is of full rank, due to Lemma 1.

5. (Application) There is no real-world data application in this paper. Since the authors proposed a more flexible model, it would be interesting to see whether the new methods exhibit any advantages over existing algorithms.

Thanks a lot for your suggestion. We added a real data example to the paper. In the case of our real data example, the DIMPLE model fits better than the MMLSBM.

Minor points:

1. In Equation (17), the matrix $H^{(m)}$ seems to be scaled by $1/\sqrt{L_m}$. Is there any reasons for that?

This is due to the fact that we define $\mathcal{H}$ as $\mathcal{H} = \mathcal{P} \times_3 \mathbf{W}^T$ (i.e., we multiply the tensor by the orthogonal matrix $\mathbf{W}^T$ rather than $\mathbf{D}_c^{-1}\mathbf{W}^T$ that averages the layers). This is just done for the convenience of the proof. Due to Assumption **A1**, one can obtain similar results without this scaling.

2. Why is K-means clustering applied to reduced matrix $\widehat{\mathbf{W}}$ instead of full matrix $\widetilde{\mathbf{W}}$ in equation (15)? Does this reduction improve the result theoretically or it is just empirically better?

Matrix $\widehat{\mathbf{W}}$ has M columns while matrix $\widetilde{\mathbf{W}}$ has L. Hence, matrix $\widehat{\mathbf{W}}$ leads to lower errors. We could not show this theoretically (although we believe that this might be possible) but $\widehat{\mathbf{W}}$ works better in simulations.

3. On page 29, it seems that the inequalities should be

$$\sigma_1(F) \leq \sigma_1^2(\overline{\mathbf{D}})\, K \sqrt{\max_m L_m}, \quad \sigma_M(F) \geq \sigma_M^2(\overline{\mathbf{D}})\, K \sqrt{\min_m L_m}$$

Thank you! We corrected the error.

Responses to the Reviewer 2

1. It made sense to me that in the constant degree regime, the fraction of miss-clustered networks would not go to zero as the number of networks increased, as the overlap of the clusters does not go away. But it's less intuitive to me why the bound for the nodal classification is tight, and increasing L only helps up to a point, or if a different algorithm might do better. Can more intuition be given here? For example, it seems the networks in a given group could be combined (i.e., summed) into a potentially non-sparse network, whose degree was increasing with the number of group members. (However, there would be contamination in such an approach due to the misclassified group members)

Consider a simple case where $M = 2$, $L_1 = L_2 = L/2$ and all networks are perfectly balanced, i.e., $\mathbf{D}^{(l)} = n/K\,\mathbf{I}_K$ for $l = 1, ..., L$. For simplicity, let $c(l) = 1$ for $1 \le l \le L/2$ and $c(l) = 2$ for $L/2 + 1 \le l \le L$. Just for explanation purposes, let $\hat{c}(l) \ne c(l)$ for $L/2(1 - \delta_1) \le l \le L/2(1 + \delta_2)$, so that the overall between-layer clustering error is $\delta = (\delta_1 + \delta_2)/2$. Again, for simplicity, assume that all layer networks are assortative, so that one can base the within-layer clustering on averaging networks in the estimated layer.

Now, in order to make the effects of the between-layer clustering errors more evident, only for illustration purposes we assume that the layer probability matrices $\mathbf{P}^{(l)}$ are available rather than adjacency matrices $\mathbf{A}^{(l)}$. Then, the within-layer clustering for layer 1 will be based on the matrix

$$\widehat{\mathbf{P}} = \mathbf{Z}^{(1)} \sum_{l=1}^{(1-\delta_1)L/2} \boldsymbol{B}^{(l)} \, (\mathbf{Z}^{(1)})^T + \mathbf{Z}^{(2)} \sum_{l=L/2+1}^{(1+\delta_2)L/2} \boldsymbol{B}^{(l)} \, (\mathbf{Z}^{(2)})^T$$

where the SVD of $\boldsymbol{B}^{(l)}$ are $\boldsymbol{B}^{(l)} = \mathbf{V}^{(l)} \mathbf{\Lambda}^{(l)} (\mathbf{V}^{(l)})^T$, $\mathbf{V}^{(l)} \in \mathcal{O}_K$, and all diagonal entries of the diagonal matrices $\mathbf{\Lambda}^{(l)}$ are positive. Now,once more, only for illustration purposes, we assume that for some matrices $\mathbf{V}_1, \mathbf{V}_2 \in \mathcal{O}_K$, one has $\mathbf{V}^{(l)} = \mathbf{V}_{c(l)}$, $l = 1, ..., L$. Therefore, taking into account that $\mathbf{D}^{(l)} = n/K\,\mathbf{I}_K$, we obtain that

$$\widehat{\mathbf{P}} = \widetilde{\mathbf{U}}^{(1)} \, \widetilde{\mathbf{\Lambda}}^{(1)} \, (\widetilde{\mathbf{U}}^{(1)})^T + \widetilde{\mathbf{U}}^{(2)} \, \widetilde{\mathbf{\Lambda}}^{(2)} \, (\widetilde{\mathbf{U}}^{(2)})^T$$

where $\widetilde{\mathbf{U}}^{(m)} = \mathbf{U}_z^{(m)} \, \mathbf{V}_m$, $m = 1, 2$, and

$$\widetilde{\mathbf{\Lambda}}^{(1)} = \frac{n}{K} \sum_{l=1}^{(1-\delta_1)L/2} \mathbf{\Lambda}^{(l)}, \quad \widetilde{\mathbf{\Lambda}}^{(2)} = \frac{n}{K} \sum_{l=L/2+1}^{(1+\delta_2)L/2} \mathbf{\Lambda}^{(l)}$$

Hence, if the value of $\delta = (\delta_1 + \delta_2)/2$ is not infinitesimally small, the values of the diagonal elements in $\widetilde{\mathbf{\Lambda}}^{(1)}$ and $\widetilde{\mathbf{\Lambda}}^{(2)}$ may be comparable and, therefore, the main K eigenvectors of the SVD of $\widehat{\mathbf{P}}$ will be different from $\widetilde{\mathbf{U}}^{(1)}$, which will lead to within-layer clustering errors. Of course, when one uses the adjacency matrices rather than the probability matrices, the random errors may lead to even higher within-layer clustering errors.

Note that this is exactly the phenomenon we observe in simulations. Indeed, as Figures 9 and 10 show, when the between-layer clustering errors are relatively high, they affect the within-layer clustering errors. However, as soon as they fall below certain level, averaging of layers (or their squares) lead to a sharp decline of the within-layer clustering errors.

(a) On a related note, do existing error bounds for the MMLSBM, which is more restrictive than the setting of this paper, also have this behavior?

We have discussed the issue in the Discussion section. Specifically, we write:

"Incidentally, we observe that a similar phenomenon holds in the MMLSBM, where the block probability matrices are the same in all layers of each of the groups. To the best of our knowledge, so far, there have been only two papers that studied MMLSBM, specifically, [5] and [3]. Note that [5] simply assume that $L \leq n$, which makes the issue of error rates for a growing value of L inconsequential. Similarly, the ALMA clustering error rates in [3]

$$\begin{aligned} R_{BL}^{ALMA} &\lesssim C\left(\rho_n^{-1}\, n^{-2} + \rho_n^{-2}\, n^{-2}\, [\min(n, L)]^{-1}\right), \\ R_{WL}^{ALMA} &\lesssim C\left(n^{-1}\, L^{-1}\, \rho_n^{-1} + \rho_n^{-1}\, n^{-2} + \rho_n^{-2}\, n^{-2}\, [\min(n, L)]^{-1}\right), \end{aligned}$$

imply that, for given n and ρ_n, as L grows, the clustering errors flatten.

2. The paper solves a problem that is similar to a recent paper by the first author, in which an alternating minimization method (called ALMA) is used to estimate a restricted version of the current model. Readers would benefit from the author's insight on the two approaches, beyond knowing that the new approach is more general. For example, the new method could be used in the earlier more restricted setting – how would it's performance compare to ALMA? (To be clear, I agree that the generality of the new method is of practical importance, as the authors claim; I also think the aesthetic properties of the new method make it of interest as well.)

In order to better address your concerns (and concerns of the first reviewer), we added a section to the paper, where we study application of our algorithms and ALMA algorithm designed for the MMLSBM to data, generated from both the DIMPLE model and the MMLSBM. As our simulations show, while our algorithms work in the case of th MMLSBM (although less efficiently since they "do not know" about additional structure), the algorithms designed for the MMLSBM have poor performance when they are applied to the data generated from the DIMPLE model.

3. I was unable to find a definition for the simulation parameters c, d and w – assuming they are somewhere in the paper, adding a pointer to them in the simulation section would be helpful.

Thank you for your question. We added explanations about the parameters c, d and w to each of the figures. They indeed were defined in the Simulations section, but you are completely right: a reader should not need to search for them.

4. In the simulations the authors write that "Fig. 3, 4 reveal that when L is large, the within-layer clustering rates do not reduce when L grows" However, only four values of L are tried in these plots, so that there are not enough data points for the trend to be entirely clear to the reader. For algorithm 2, at least, it seems like many of the lines do decrease for all 4 values of L that were chosen. As a result, additional larger values of L should be added to the simulation, if it is computationally feasible.

We added more values of L to our simulations. Simulations confirm the pattern.

References

[1] T. T. Cai and A. Zhang. Rate-optimal perturbation bounds for singular subspaces with applications to high-dimensional statistics. *Ann. Statist.*, 46(1):60–89, 02 2018.

[2] Y. Chen, Y. Chi, J. Fan, and C. Ma. Spectral methods for data science: A statistical perspective. *Foundations and Trends in Machine Learning*, 14(5):566–806, 2021.

[3] X. Fan, M. Pensky, F. Yu, and T. Zhang. Alma: Alternating minimization algorithm for clustering mixture multilayer network. *ArXiv:2102.10226*, 2021.

[4] B.-Y. Jing, T. Li, Z. Lyu, and D. Xia. Community detection on mixture multi-layer networks via regularized tensor decomposition. *ArXiv:2002.04457*, 2020.

[5] B.-Y. Jing, T. Li, Z. Lyu, and D. Xia. Community detection on mixture multilayer networks via regularized tensor decomposition. *The Annals of Statistics*, 49(6):3181 – 3205, 2021.

[6] J. Lei and K. Z. Lin. Bias-adjusted spectral clustering in multi-layer stochastic block models. *ArXiv:2003.08222*, 2021.

[7] S. Paul and Y. Chen. Spectral and matrix factorization methods for consistent community detection in multi-layer networks. *Ann. Statist.*, 48(1):230–250, 02 2020.