To Reviewer #1: Thank you for your supportive comments.

A1: Special insights in proofs. The proof strategy of Theorem 3.1 is based on the construction of an inverse M-matrix. To the best of our knowledge, this is the first work to discuss and prove this special behavior of the ℓ_1 norm. In the proof of Theorem 3.8, we introduce a linear operator to handle the Laplacian constraints and derive a concentration bound for Laplacian GGM, which are new and necessary for the proof. The previous approaches do not solve these problems.

Other comments: The initialization of Algorithm 1 is generated by converting the lower triangle matrix of $(\boldsymbol{S}+\epsilon\boldsymbol{I})^{-1}$ into a vector and removing the negative elements. For the k-th iteration, the initialization of Step 3 is set as $\hat{\boldsymbol{w}}^{k-1}$. The c_0 only depends on the true graph weights. The papers Wang-Ren-Gu AISTATS 2016 and Xu-Ma-Gu NeurIPS 2017 will be discussed. The performance of GGM methods is usually not satisfactory due to lack of Laplacian constraints.

To Reviewer #2: Thank you for your supportive comments.

B1: COVID-19 data with Laplacian GGM. In the case of the data following Laplacian GGM, our formulation in Eq. (3) can be viewed as a regularized maximum likelihood estimation of precision matrix. In a more general setting with non-Gaussian distribution such as COVID-19 data, Eq. (3) can be related to the log-determinant Bregman divergence regularized optimization, and the learned graph weights can quantify the similarity between nodes. This is because the trace term in Eq. (3) can be written as Laplacian quadratic [10, 20], which tends to assign a large weight between nodes if their signal values are similar. The COVID-19 data consists of two groups, red and green nodes. It is natural to assume that the nodes belonging to different groups are dissimilar from each other, while the nodes in the same group are similar. In this sense, the performance of our learned graph in Figure 4 is significant, because most connections are between nodes within the same group, and only a few connections (gray edges) are between nodes from distinct groups.

Other comments: The insight behind the unexpected behavior of ℓ_1-norm is related to the Laplacian constraints, and related discussions will be added. We will further clarify the steps in Section 3.2, and update the broader impact section.

To Reviewer #3: Thank you for your helpful comments.

C1: Introduction and Related Work. We will include additional discussions on GGM and clarify the difference between GGM and Laplacian GGM. Some related works on GGM methods are discussed in the Introduction, and thus not repeated in the Related Work due to limited pages. We will reorganize the first two sections in the final version.

C2: Left boundary in Theorem 3.1. We apologize that we did not make it clear our statement that a large regularization parameter of the ℓ_1-norm method will lead to learn a fully connected graph. We rephrase it here: As long as the regularization parameter is large enough (larger than the left boundary in Theorem 3.1), the learned graph by the ℓ_1-norm method must be fully connected. In practice, the bound leading to a fully connected graph may be smaller than the left boundary in the theorem.

C3: Experiments on COVID-19 data. We indeed tested many values of λ for GLE-ADMM. The presented result with $\lambda = 0$ in the paper is the sparsest one, and the other estimated graphs are denser. Due to the limited space, here we only show a result with $\lambda = 0.1$ in Figure 1 that is already a fully connected graph. We will add additional results in Appendix of the final version. We agree that the COVID-19 data set is small. This data set was collected while the pandemic was in its infancy, and thus the data availability was limited. We will update our results with a bigger data set in the final version.

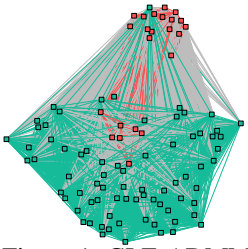

Figure 1: GLE-ADMM with $\lambda = 0.1$.

Other comments: Thank you for pointing out these typos and we will correct them.

To Reviewer #4: Thank you for your supportive comments.

D1: Theorem 3.1 and its empirical results. 1. We agree that the scaling of the sample data impacts λ^*. The theoretical characterization of the function of sparsity with respect to λ for $\lambda \in [0, \lambda^*]$ is challenging, because the sparsity involves counting the number of nonzero elements which is always highly non-convex and non-continuous. 2. It is precisely the point of our paper to show under solid mathematical grounds and illustrate via extensive numerical results that the ℓ_1-norm method is an incorrect approach and many papers are following it improperly.

D2: Scalability of the algorithm. The computational complexity of our algorithm is dominated by Cholesky decomposition, which can save computation for logdet and matrix inverse. Table 1 shows the running time for different numbers of nodes p, where the graph weight $\boldsymbol{w}$ has the dimension $p(p-1)/2$. The numbers of outer iterations for different p are 28, 28, 29 and 26, respectively, and the average numbers of inner iterations are usually less than 5. It is of great interest to develop more efficient algorithms and implement algorithms in faster programming languages such as C++ in order to accommodate larger data sets in future work.

Table 1: Comparison of computational time (seconds)

p	50	100	500	1000
GLE-ADMM	0.117	0.676	57.032	485.465
NGL-SCAD	0.023	0.097	9.000	67.018

D3: COVID-19 data. 1. Regarding the concern that the data does not follow Laplacian GGM, please refer to **B1**; 2. We agree that the quality of the results cannot be precisely verified, because there is no underlying true graph in practice. Nonetheless, the synthetic experiments has shown the ability of our method to learn the ground truth. The significance of our result on COVID-19 data can be verified in some sense (Refer to **B1**). 3. Regarding GLE-ADMM with other parameter values and the size of data set, please refer to **C3**.