We thank the reviewers for the insightful comments. Below are our responses to each reviewer's comments.

Reviewer #1:

Explanation on the Technical Conditions. We already provided some explanations and intuitions on the technical conditions but these explanations were buried in the text without concrete references to the corresponding conditions: Line 117-121 for condition (A.1) and (A.2); Line 126 for condition (A.3); Line 147-156 for conditions (i) - (iv) on the hyperparameters and sample size; Line 171-174 on the minimal signal condition (v). In the revision, we will re-organize them and make concrete references to make the discussion on the conditions clearer. We will also provide more detailed explanations of the technical conditions (A.1) - (A.3). While we think it is possible to relax some of our conditions, it would require some new technical machinery which is beyond the scope of the current article.

Choice of Classes and Its Influence on Estimation. In our setting, the classes are pre-specified by side information or domain knowledge; the same assumption is held in existing work on multiple graphical models. For estimation accuracy, the rate of convergence for estimating Θ_k (the precision matrix for the k-th class) is $O_p(\sqrt{(\log p)/n})$ regardless of the class size K, except that to achieve this rate, the sample size n has a lower bound $M_0^2 \max(d^2, K) \log p$ that depends on K (condition (ii) in Theorem 2). For our bike-sharing data, it is natural to expect the precision matrix changes over year due to annual policy decisions, economic conditions, and other aspects of the business. So classes are chosen to be different years.

Reviewer #2:

Computational Times. In the revision, we shall add the following table on average computational times of all the methods based on 10 replications. The computational time of our method is comparable to the competitors except the Pooled method, which restrictively assumes the same precision matrix for all classes and has much worse performance compared to our method. Therefore, our method is competitive even after considering the runtimes.

	Nearest-neighbor Network			Scale-free Network		
	$\rho = 0$	$\rho = 0.25$	$\rho = 0.5$	$\rho = 0$	$\rho = 0.25$	$\rho = 0.5$
Our method ($\alpha = 1$)	3.667(0.040)	3.645(0.087)	3.552(0.026)	3.556(0.037)	3.545(0.030)	3.537(0.033)
Our method ($\alpha = n$)	7.792(0.456)	4.596(0.643)	3.597(0.049)	5.285(2.623)	3.600(0.025)	3.578(0.023)
BAGUS	3.635(0.023)	3.572(0.027)	3.547(0.021)	3.553(0.012)	3.546(0.022)	3.534(0.018)
Pooled	1.211(0.010)	1.178(0.013)	1.169(0.008)	1.184(0.015)	1.173(0.008)	1.168(0.010)
GGL	8.715(0.314)	8.034(0.689)	5.482(1.528)	8.086(0.262)	6.139(0.678)	3.074(0.270)

Selection of Hyperparameters. For all methods, we use a grid search to select the set of hyperparamters that minimizes BIC. For BAGUS and Pooled methods, we follow the same tuning procedure in [10] and tune the spike and slab variances (v_0, v_1) with $v_0 = (0.25, 0.5, 0.75, 1) \times \sqrt{1/(n \log p)}$ and $v_1 = (2.5, 5, 7.5, 10) \times \sqrt{1/(n \log p)}$. For GGL, we tune the two penalty parameters (λ_1, λ_2) as in [5] with $\lambda_1 = (0.1, 0.2, \ldots, 1)$ and $\lambda_2 = (0.1, 0.3, 0.5)$. We shall add these details in the revision.

Bayesian Characterization and Prior. Our model is indeed a Bayesian model although we emphasize on the MAP estimator corresponding to our Bayesian model. i) Our model is formulated from a Bayesian perspective with a continuous spike and slab prior distribution. Although this prior does not directly place mass on sparse solutions, the latent binary indicators γ_{ij} introduced can distinguish between"signal" and "noise". This is a common technique used in the Bayesian literature to avoid the computational bottleneck of degenerate priors. ii) Although we use the MAP estimator that also has a penalization interpretation, it is not the only goal in our inference. Without the Bayesian machinery in the paper, we cannot extract the posterior inclusion probabilities for structure recovery using (2.5) and provide consequent strong guarantees for graph selection in Theorem 3. iii) For scalability, we compute the MAP estimator instead of sampling from the full posterior. Full posterior sampling for high-dimensional GGMs is computationally expensive, for example, in the two Bayesian papers we cited [18, 22], the dimension p in all empirical studies is less than 22. Further, although MCMC-based samplers are proposed to recover the full posterior in [18], for structure recovery, only one model is reported based on the same thresholding procedure as our eq (2.6).

Notations in (A.1). The notation is $(\log p)/n$. We shall add the parentheses in the revision.

Reviewer #3:

EM vs Direct Optimization. We agree that it is possible to directly optimize (2.4) after we show that the restricted optimization problem is strictly convex in Theorem 1. We use the proposed EM algorithm due to the following two reasons: 1) the E-step provides estimates of posterior inclusion probabilities (2.5), which will be used for structure recovery; 2) the computational complexity of our EM algorithm is $O(p^3)$, which is already as efficient as the state-of-the-art algorithms for Graphical Lasso problems [9, 10].

Comparison with Other Bayesian Alternatives. We did not compare with Bayesian approaches [18] and [22] since their MCMC samplers are not scalable with large p. Specifically, the largest p handled in [18] and [22] is only 20 and 22, respectively, and [22] states in their Section 6.6 that their method is not scalable to large p and reports the average computational time to be (12.4 ± 0.5) hours for its simulation design with $K = 2$, $n_1 = n_2 = 50$, and $p = 20$. Therefore, they are not computationally manageable to our simulation designs and real data application with large p.