

## *Supplementary Information*

# **for “network activity evaluation reveals significant gene regulatory architectures during SARS-CoV-2 viral infection from dynamic scRNA-seq data”**

Chuanyuan Wang, Zhi-Ping Liu

### **S1. Contributions of scNAE**

scNAE is a method for computing the empirical statistical significance of a gene regulatory network, i.e., network activity, in a cell-specific environment. It introduces the prior knowledge of gene regulations from literature as the constraints to make the objective function achieve both reasonability and flexibility. Briefly, the contribution of scNAE can be summarized as follows.

- (1) Time-course scRNA-seq data are modeled by ordinary differential equations (ODE). Based on the gene expression profiling information in individual cell-specific environment, scNAE simulates the regulatory rules between TFs and target genes with the differential trajectories of genes over time. Moreover, by constructing a linear regression loss function with regularization term, scNAE can numerically measure the degree of intimacy between network structure and expression data. It takes into account of specific network architectures in different cell types, and then its evaluation is more precise and targeted than that in the bulk gene expression data.
- (2) scNAE integrates prior gene regulatory knowledge and single-cell transcriptomic data by a regularized constraint programming model. Since scNAE evaluates the activity of a specific gene network architecture, we introduce the regulatory relationship of a given network as the knowledge-based constraints subjecting to the objective function. These constraints reduce the influence of non-knowledge regulations on network architecture. Moreover, the objective function is to be optimized by introducing a sparse penalty term to make GRN reasonable in line with the cell-specific environments.
- (3) Network activity evaluation is achieved based on the statistical significance of a non-parametric test procedure. We rewire the network linkages by randomly permuting the wiring rules between genes in the evaluating network. Then, scNAE tests these amount of generated networks as well as the evaluating network to compare the achieved values of objective function quantifying the consistency strength between network and data. Finally, an empirical *P*-value is derived to quantify the activity of the evaluating gene network in specific cell types. In a mathematical programming framework, scNAE unifies network inference and network evaluation from time series of scRNA-seq data. The active gene regulatory architectures can be revealed simultaneously in response to specific phenotypic state and external stimuli or perturbation.

### **S2. A detailed description of the SARS-CoV-2 dataset**

The data measures the transcriptomic states of human bronchial epithelial cells at 1, 2, and 3 days after SARS-CoV-2 viral infection. It contains a total of 24,482 gene expression levels in 77,650 cells across these days. These cells are classified into four states: "Mock", "1 dpi", "2 dpi" and "3 dpi", and the cell types have been identified as a total of 8 classes, BC/Club, Basal cells, Ciliated cells, Club cells, Goblet cells, Ionocytes, Neuroendocrine cells and Tuft cells.

### **S3. SARS-CoV-2-related pathways**

The SARS-CoV-2-related pathways are listed in Table S1, where ‘Gene’ (cropped) represents the overlapping gene ID in the pathway gene, gene expression profile and gene regulations in the RegNetwork database.

**Table S1.** Pathways related to SARS-CoV-2.

Pathway ID	Database	Pathway description	# of Gene	# of Gene (cropped)	# of Edge
hsa05171	KEGG	Coronavirus disease - COVID-19	84	84	271
R-HSA-9694516	Reactome	SARS-CoV-2 Infection	71	71	129
WP5039	WikiPathway	SARS-CoV-2 innate immunity evasion and cell-specific immune response	68	43	185
WP5115		Network map of SARS-CoV-2 signaling pathway	221	108	449
WP5098		T-cell activation SARS-CoV-2	87	54	146
WP4868		Type I interferon induction and signaling during SARS-CoV-2 infection	31	18	38
WP5038		Mitochondrial immune response to SARS-CoV-2	32	10	27
WP5113		Antiviral and anti-inflammatory effects of Nrf2 on SARS-CoV-2 pathway	32	20	59

#### S4. Solve the scNAE optimization problem by ADMM

In the regularized constraint programming, when  $p = 2$ , it can be solved by quadratic programming because the objective function is equivalent to a ridge regression. When  $0 \leq p < 1$ , the objective function is non-convex, so the ADMM algorithm can be employed to optimize the loss function. It is worth noting that the loss function contains inequality constraints, which are often difficult to solve directly by ADMM. Therefore, a relaxation constraint  $\xi$  is introduced here to convert the inequality constraints into equality constraints for easily solving. Then, the loss function is formulated as follows

$$\begin{aligned} \min \quad & \|\mathbf{X} - \mathbf{D}\mathbf{G}\|_F^2 + \lambda \|\mathbf{Q}\|_p \\ \text{s.t.} \quad & \mathbf{Q} = \mathbf{G}, \\ & \mathbf{N} \otimes \mathbf{G} + \xi^2 = 0, \\ & \mathbf{E} \otimes \mathbf{G} = 0, \end{aligned} \quad (\text{S1})$$

where  $\mathbf{Q}$  is introduced to make the constrained optimization problem of  $\mathbf{G}$  separable. Furthermore, rewrite Equation (S1) as the Lagrangian function

$$\begin{aligned} \mathcal{L} = & \|\mathbf{X} - \mathbf{D}\mathbf{G}\|_F^2 + \lambda \|\mathbf{Q}\|_p + \frac{\mu_N}{2} \left\| \mathbf{N} \otimes \mathbf{G} + \xi^2 + \frac{y_N}{\mu_N} \right\|_F^2 \\ & + \frac{\mu_E}{2} \left\| \mathbf{E} \otimes \mathbf{G} + \frac{y_E}{\mu_E} \right\|_F^2 + \frac{\mu_Q}{2} \left\| \mathbf{G} - \mathbf{Q} + \frac{y_Q}{\mu_Q} \right\|_F^2. \end{aligned} \quad (\text{S2})$$

where  $y_*$  is the Lagrangian multiplier. In addition,  $\mu_* = \mu_N = \mu_E = \mu_Q > 0$  is set to simplify the operation of Equation (S2). Further, the ADMM optimization of this problem can be divided into three steps.

(1) Update  $\mathbf{G}$  then  $\mathbf{Q}$ ,  $y_*$  and  $\xi$  are fixed:

$$\mathbf{G}^{r+1} = \arg \min_{\mathbf{G}} \mathcal{L}(\mathbf{G}, \mathbf{Q}^r, y_*^r, \xi^r). \quad (\text{S3})$$

Combining (S2) and (S3),  $\mathbf{G}$  can be optimized as follows.

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{G}} = & 2\mathbf{D}^T(\mathbf{D}\mathbf{G} - \mathbf{X}) + \mu_* (\mathbf{N} \otimes \mathbf{I}) \left( \mathbf{N} \otimes \mathbf{G} + \xi^2 + \frac{y_N}{\mu_*} \right) + \mu_* (\mathbf{E} \otimes \mathbf{I}) \left( \mathbf{E} \otimes \mathbf{G} + \frac{y_E}{\mu_*} \right) + \mu_* \left( \mathbf{G} - \mathbf{Q} + \frac{y_Q}{\mu_*} \right), \\ 0 = & 2\mathbf{D}^T\mathbf{D}\mathbf{G} - 2\mathbf{D}^T\mathbf{X} + \mu_* (\mathbf{N} \otimes \mathbf{I}) \left( \mathbf{N} \otimes \mathbf{G} + \xi^2 + \frac{y_N}{\mu_*} \right) + \mu_* (\mathbf{E} \otimes \mathbf{I}) \left( \mathbf{E} \otimes \mathbf{G} + \frac{y_E}{\mu_*} \right) + \mu_* (\mathbf{G} - \mathbf{Q}) + y_Q. \end{aligned} \quad (\text{S4})$$

Calculating the inverse matrix in the ADMM algorithm is a very time-consuming operation, so a gradient descent algorithm is introduced here to obtain an approximate solution [1]. The final  $\mathbf{G}$  update is as follow

$$\mathbf{G} = \mathbf{G} - \eta \left( \begin{aligned} & 2\mathbf{D}^T\mathbf{D}\mathbf{G} - 2\mathbf{D}^T\mathbf{X} + \\ & \mu_* (\mathbf{N} \otimes \mathbf{I}) \left( \mathbf{N} \otimes \mathbf{G} + \xi^2 + \frac{y_N}{\mu_*} \right) + \\ & \mu_* (\mathbf{E} \otimes \mathbf{I}) \left( \mathbf{E} \otimes \mathbf{G} + \frac{y_E}{\mu_*} \right) + (\mathbf{G} - \mathbf{Q}) \end{aligned} \right) + y_Q, \quad (\text{S5})$$

where  $\eta$  represents the speed of gradient descents.

(2) Update  $\mathbf{Q}$  then  $\mathbf{G}$ ,  $y_*$  and  $\xi$  are fixed:

$$\mathbf{Q}^{r+1} = \arg \min_{\mathbf{Q}} \mathcal{L}(\mathbf{G}^{r+1}, \mathbf{Q}, y_*^r, \xi^r). \quad (\text{S6})$$

Combining (S2) and (S6),  $\mathbf{Q}$  can be optimized as follows.

$$\mathbf{Q}^{r+1} = \arg \min_{\mathbf{Q}} \lambda \left\| \mathbf{Q}^r \right\|_p + \frac{\mu_s}{2} \left\| \mathbf{G}^{r+1} - \mathbf{Q}^r + \frac{\mathbf{y}_Q}{\mu_s} \right\|_F^2. \quad (\text{S7})$$

Solving the  $L_p$ -norm constraint of  $\mathbf{Q}$  is a non-convex problem (when  $0 \leq p < 1$ ), which is usually difficult to obtain an optimal solution. Therefore, the  $L_p$ -norm can be approached continuously to obtain an approximate solution. The detailed solutions can be divided into the following three parts.

-  $L_1$ -norm:

The objective function of  $L_1$ -norm optimization is minimized as follows.

$$\mathbf{Q}^{r+1} = \arg \min_{\mathbf{Q}} \lambda \left\| \mathbf{Q}^r \right\|_1 + \frac{\mu_s}{2} \left\| \mathbf{A}^{r+1} - \mathbf{G}^r + \frac{\mathbf{y}_Q}{\mu_s} \right\|_F^2. \quad (\text{S8})$$

Usually, the following shrinkage operator is used to solve the  $L_1$ -norm function [2, 3].

$$\begin{aligned} \mathbf{Q}^{r+1} &= \text{prox}_{(\lambda/\mu_s)\|\cdot\|_1} \left( \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right) \\ &= \text{sign}(\mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s}) \max \left\{ \left| \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right| - \frac{\lambda}{\mu_s}, 0 \right\}. \end{aligned} \quad (\text{S9})$$

-  $L_p$ -norm ( $0 < p < 1$ ):

The sparsity of optimal solutions is usually stronger than the  $L_1$ -norm when  $0 < p < 1$  [4]. In addition, this causes the optimization problem of (S7) becomes non-convex. Therefore, Chartrand *et al.* introduced a shrinkage operator to obtain an approximate solution, which is defined as follows [5].

$$\begin{aligned} \mathbf{Q}^{r+1} &= \text{prox}_{(\lambda/\mu_s)\|\cdot\|_p} \left( \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right) \\ &= \max \left\{ \left| \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right| - \frac{\lambda}{\mu_s} \left| \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right|^{p-1}, 0 \right\} \left| \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right|^{p-1}. \end{aligned} \quad (\text{S10})$$

Since the value of  $p$  is not specifically defined, the three cases of  $p = 1/3$ ,  $1/2$  and  $2/3$  are discussed in subsequent experiments to compare their performance with the  $L_p$ -norm.

-  $L_0$ -norm:

The  $L_0$ -norm aims to solve the number of non-zero elements in the penalty term, that is, the number of  $\mathbf{Q} \neq 0$  in (S6). Solving it is a challenging NP-hard problem, and the  $L_1$ -norm is often used as the relaxation solution of the  $L_0$ -norm [6]. However, this relaxed solution may lead to a lack of precision [7]. Therefore, a simple and convergent  $L_0$ -norm solution method was proposed by Zhao *et al.* [8]. For each element  $\mathbf{Q}_i$  in  $\mathbf{Q}$ , its shrinking operator is defined as follows.

$$\begin{aligned} \mathbf{Q}_i^{r+1} &= \text{prox}_{(\lambda/\mu_s)\|\cdot\|_0} \left( \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right)_i \\ &= \begin{cases} \left( \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right)_i, & \text{if } \left( \mathbf{G}^{r+1} + \frac{\mathbf{y}_Q}{\mu_s} \right)_i^2 \geq \frac{2\lambda}{\mu_s} \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (\text{S11})$$

(3) Update  $\mathbf{y}_*$  then  $\mathbf{G}$ ,  $\mathbf{Q}$  and  $\xi$  are fixed:

$$\begin{aligned} \mathbf{y}_N^{r+1} &= \max \left( \mathbf{y}_N^r + \mu_s (\mathbf{N} \odot \mathbf{G} + \xi^2), 0 \right) \\ \mathbf{y}_E^{r+1} &= \mathbf{y}_E^r + \mu_s (\mathbf{E} \odot \mathbf{G}) \\ \mathbf{y}_Q^{r+1} &= \mathbf{y}_Q^r + \mu_s (\mathbf{G} - \mathbf{Q}) \end{aligned} \quad (\text{S12})$$

To ensure that the optimal solution satisfies the Karush–Kuhn–Tucker (KKT) condition, Lagrange multiplier  $\mathbf{y}_N^{r+1} \geq 0$  need to be constrained.

(4) Update  $\xi$  then  $\mathbf{G}$ ,  $\mathbf{Q}$  and  $\mathbf{y}_*$  are fixed:

$$\xi^{r+1} = \arg \min_{\xi} \mathcal{L}(\mathbf{G}^{r+1}, \mathbf{Q}^{r+1}, \mathbf{y}_*^{r+1}, \xi^r). \quad (\text{S13})$$

Further, the function  $\mathcal{L}$  takes the partial derivative of the variable  $\xi$ , and the result is as follows.

$$\frac{\partial \mathcal{L}}{\partial \xi} = 2\xi \mu_s \left( \mathbf{N} \otimes \mathbf{G} + \xi^2 + \frac{\mathbf{y}_N}{\mu_s} \right). \quad (\text{S14})$$

Finally, the update rule of  $\xi^{r+1}$  is as follows.

$$\xi^{r+1} = \sqrt{-\left( \mathbf{N} \otimes \mathbf{G} + \frac{\mathbf{y}_N}{\mu_s} \right)}. \quad (\text{S15})$$

The above 4 variables are updated iteratively until the convergence condition is reached. The algorithm sets the convergence condition to reach the maximum number of iterations ( $r_{\max} = 1000$ ) or  $\|\mathbf{G} - \mathbf{Q}\|_2^2 \leq 10^{-5}$ . The parameters  $\log_{10}(\mu) = [-6, -4, \dots, 1]$  and  $\log_{10}(\eta) = [-2, -1, 0]$  are

empirically optimized. The parameter  $\lambda$  range of the regularization term is defined as 0 to  $\|2\mathbf{D}^T \mathbf{X}\|_{L_p}$  [9], and the optimal value can be selected by a grid search algorithm (e.g.,  $\log(\lambda/\lambda_{\max}) = [-6, -5, \dots, 0]$ ). The corresponding algorithm details are listed in Algorithm 1.

<b>Algorithm 1: ADMM for solving scNAE</b>	
Input:	$\mathbf{X}, \mathbf{P}, \mathbf{N}, \mathbf{E}$
Output:	$\mathbf{G}$
Parameters:	$\lambda, \mu$ and $\eta$
Data Initialize:	$\mathbf{Q} = \mathbf{0}, \mathbf{y} = \mathbf{0}, \varepsilon = \mathbf{0}$ ,
Repeat	
1).	Update $\mathbf{G}$ by (S5);
2).	Update $\mathbf{Q}$ by (S9), (S10) or (S11);
3).	Update $\mathbf{y}_s$ by (S12).
4).	Update $\xi$ by (S15).
End convergence	

## S5. The results of simulation datasets

The results of scNAE, topologyGSA and SPIA on the 12 simulation datasets are listed in Table S2 (Covers 2 and 3 cell types), Table S3 (Covers 4 cell types) and Table S4 (Covers 5 cell types), respectively. According to the numerical experiments, we summarize the results as follows:

- (1) scNAE is robust with different regularization terms. scNAE models with different penalty terms of  $L_p$  parameters  $p$  achieve similar evaluation results. The mean  $P$ -values are 0.038, 0.040, 0.037, 0.036, 0.033 and 0.034 for the six  $L_p$  norms respectively. From a global perspective, these results are all significant, which means scNAE can precisely obtain the consistency between network and data. Furthermore, its standard deviation of these mean  $P$ -values is 0.002, which proves that scNAE is not sensitive to parameters  $p$ .
- (2) scNAE outperforms the existing methods. As shown in Fig. 3(a), the average significance of the scNAE model is better than those of topologyGSA (0.217) and SPIA (0.639) in the 42 simulated networks. Although the two algorithms capture the dependencies between genes based on pathway topology, they still focus on the statistical analysis of gene sets and are with limited interests in network topology. More importantly, the cell-specific environment represented by scRNA-seq data has not fully considered in them. In contrast, scNAE is developed for cell-specific regulatory network with more flexibility and targeted.
- (3) The more cell types, the more difficult it is to evaluate the network activity in single cells. In the simulation study, the cellular gene expression profiles are generated based on a given GRN structure. In case of generating more cell types, it becomes more difficult to probe for the consistency of GRN architectures with gene expression profiles. The mean significance for the evaluating networks by scNAE and the other methods on *Supplementary Tables S2, S3, and S4* are 0.08, 0.138, and 0.187, respectively. The cell types in *Supplementary Tables S2, S3, and S4* are gradually increased from 2-3 to 5. Therefore, we can conclude that the network activity is more difficult to be estimated in the data with more cell types.

**Table S2.** Experimental results in the S1-S6 datasets.

Datasets	Nodes	Cell Types	scNAE						topologyGSA	SPIA
			$p=0$	$p=1/3$	$p=1/2$	$p=2/3$	$p=1$	$p=2$		
S1	30	Type I	0.001	0.001	0.001	0.001	0.001	0.001	0.599	0.559
S1	30	Type II	0.007	0.008	0.008	0.005	0.015	0.008	0.005	0.272
S2	50	Type I	0.001	0.001	0.001	0.001	0.006	0.001	0.008	0.065
S2	50	Type II	0.011	0.017	0.013	0.018	0.033	0.021	0.009	0.001
S3	100	Type I	0.001	0.001	0.001	0.001	0.015	0.001	1.000	0.272
S3	100	Type II	0.003	0.002	0.001	0.001	0.001	0.002	0.504	0.047
S4	30	Type I	0.001	0.001	0.001	0.001	0.001	0.001	0.136	0.786
S4	30	Type II	0.001	0.001	0.001	0.001	0.005	0.001	0.011	0.845
S4	30	Type III	0.002	0.002	0.002	0.005	0.003	0.003	0.009	0.562
S5	50	Type I	0.002	0.002	0.001	0.002	0.001	0.002	0.006	0.438
S5	50	Type II	0.044	0.054	0.055	0.052	0.064	0.042	0.068	0.330
S5	50	Type III	0.001	0.001	0.001	0.001	0.001	0.001	0.043	0.0738
S6	100	Type I	0.001	0.001	0.001	0.002	0.001	0.001	0.019	0.890
S6	100	Type II	0.001	0.001	0.001	0.001	0.001	0.001	0.012	0.565
S6	100	Type III	0.029	0.042	0.048	0.046	0.041	0.029	0.039	0.610

**Table S3.** Experimental results in the S7-S9 datasets.

Datasets	Nodes	Cell Types	scNAE						topologyGSA	SPIA
			$p=0$	$p=1/3$	$p=1/2$	$p=2/3$	$p=1$	$p=2$		
S7	30	Type I	0.011	0.013	0.011	0.013	0.004	0.008	0.001	0.991
S7	30	Type II	0.014	0.004	0.007	0.008	0.007	0.007	0.059	0.987
S7	30	Type III	0.003	0.001	0.001	0.001	0.001	0.001	0.82	0.489
S7	30	Type IV	0.005	0.004	0.049	0.052	0.059	0.045	0.015	0.538
S8	50	Type I	0.032	0.03	0.029	0.022	0.020	0.033	0.302	0.698
S8	50	Type II	0.012	0.023	0.027	0.022	0.012	0.025	0.048	0.565
S8	50	Type III	0.001	0.001	0.001	0.001	0.001	0.001	0.017	0.272
S8	50	Type IV	0.006	0.011	0.007	0.008	0.01	0.004	0.01	0.272
S9	100	Type I	0.001	0.001	0.001	0.001	0.001	0.001	0.039	0.992
S9	100	Type II	0.369	0.384	0.344	0.312	0.335	0.399	0.241	0.998
S9	100	Type III	0.001	0.002	0.002	0.001	0.001	0.001	0.081	0.321
S9	100	Type IV	0.011	0.017	0.013	0.009	0.013	0.013	1.000	0.485

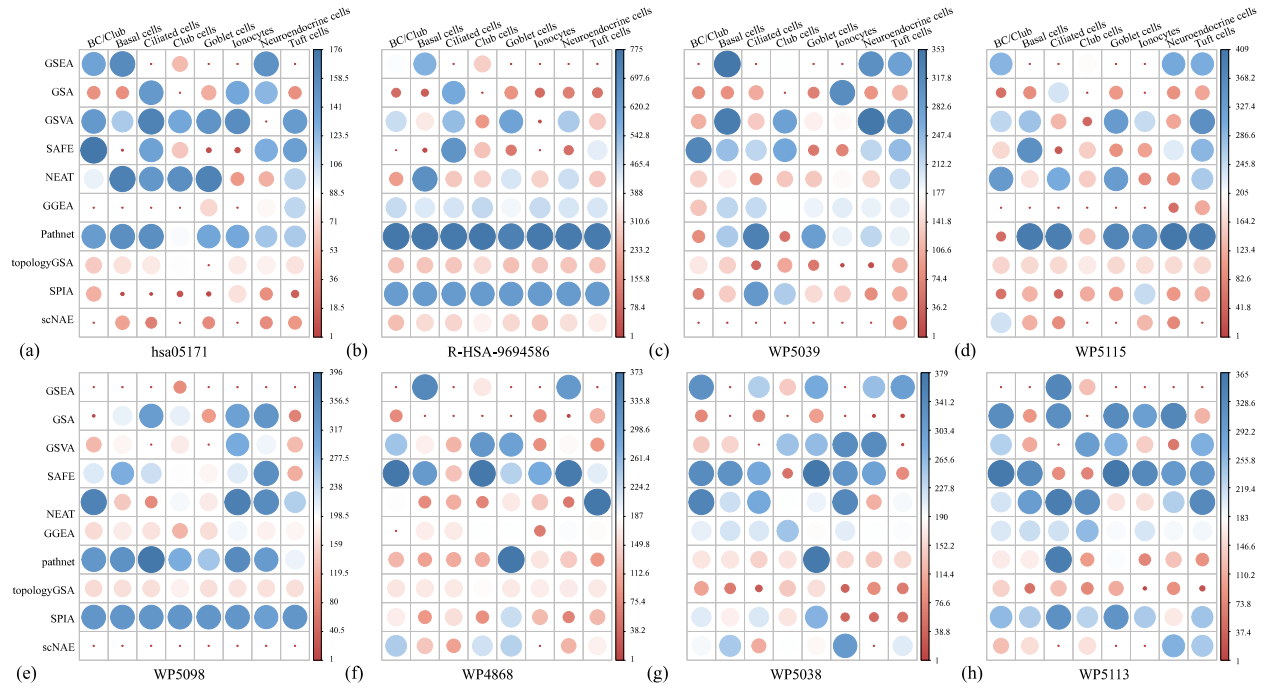
**Table S4.** Experimental results in the S10-S12 datasets.

Datasets	Nodes	Cell Types	scNAE						topologyGSA	SPIA
			$p=0$	$p=1/3$	$p=1/2$	$p=2/3$	$p=1$	$p=2$		
S10	30	Type I	0.106	0.145	0.125	0.123	0.120	0.136	0.045	0.963
S10	30	Type II	0.013	0.066	0.028	0.017	0.006	0.081	0.714	0.811
S10	30	Type III	0.012	0.001	0.006	0.007	0.015	0.001	0.007	0.927
S10	30	Type IV	0.001	0.001	0.001	0.001	0.001	0.001	0.005	0.594
S10	30	Type V	0.006	0.001	0.006	0.005	0.008	0.001	0.016	0.908
S11	50	Type I	0.468	0.408	0.402	0.413	0.156	0.123	0.030	0.995
S11	50	Type II	0.001	0.001	0.001	0.001	0.001	0.001	0.089	0.993
S11	50	Type III	0.001	0.001	0.001	0.001	0.001	0.001	0.007	0.919
S11	50	Type IV	0.004	0.002	0.001	0.004	0.005	0.004	0.044	0.562
S11	50	Type V	0.001	0.001	0.001	0.001	0.001	0.001	0.032	0.999
S12	100	Type I	0.010	0.007	0.009	0.013	0.011	0.010	1.000	0.928
S12	100	Type II	0.372	0.412	0.353	0.322	0.414	0.388	0.018	0.984
S12	100	Type III	0.001	0.001	0.001	0.001	0.001	0.001	1.000	0.687
S12	100	Type IV	0.001	0.001	0.001	0.001	0.001	0.001	1.000	0.738
S12	100	Type V	0.012	0.007	0.008	0.010	0.006	0.010	0.014	0.924

## S6. The results of SARS-CoV-2 datasets

We evaluate all pathways and examine the ranking of these pathways related to SARS-CoV-2 viral infection. The results are shown in Fig. S1, where the redder and smaller the circular area, the higher the pathway ranking. Our summary are as follows.

- (1) scNAE embraces the best performance, which is due to the fact that it effectively solves the regularized constraint programming problem with the knowledge-based regulations as constraints. It obtains the satisfactory ranking for hsa05171, WP5039 and WP5098 in the three databases, respectively. In addition, the enrichment results under different cell types are also significantly different, which further indicate the differences in gene regulatory responses caused by cell specificity.
- (2) The two set-based methods, GSEA and GSA, achieve satisfactory ranking results. Ant they have high sensitivity to R-HSA-9694516, WP5113, WP4868, WP5038 pathways. While the other two set-based methods, GSVA and SAFE, obtain low ranks for the SARS-CoV-2-related pathways. A possible reason is that these two algorithms prefer more to individual state data rather than the time series of cell state data.
- (3) As network-based methods, NEAT and Pathnet do not prioritize the key pathways that respond to specific states during viral infection. The results of other algorithms also have a certain gap compared with those of scNAE. Most of these methods can only evaluate gene activity based on existing gene regulations and have limited usage of these regulatory circuits. Furthermore, the COVID-19? pathway is inactive or micro-functioning by these methods even if the integrity of the evaluating regulatory architecture is high.



**Fig. S1.** Performance comparison of these methods on different cell types. (a)-(h) are the results analysis under different pathways.

## References

- [1] Z. Zha, X. Zhang, Y. Wu, Q. Wang, X. Liu, L. Tang, and X. Yuan, "Non-convex weighted  $\ell_p$  nuclear norm based ADMM framework for image restoration," *Neurocomputing*, vol. 311, pp. 209-224, 2018.
- [2] Z. Zha, X. Liu, X. Huang, H. Shi, Y. Xu, Q. Wang, L. Tang, and X. Zhang, "Analyzing the group sparsity based on the rank minimization methods," in 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 883-888.
- [3] S. Beygi, U. Mitra, and E. G. Ström, "Nested sparse approximation: Structured estimation of V2V channels using geometry-based stochastic channel model," *IEEE Transactions on Signal Processing*, vol. 63, no. 18, pp. 4940-4955, 2015.
- [4] F. Nie, X. Wang, and H. Huang, "Multiclass capped  $\ell_p$ -Norm-norm SVM for robust classifications," in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, California, USA, 2017, pp. 2415-2421.
- [5] R. Chartrand, "Nonconvex splitting for regularized low-rank+ sparse decomposition," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5810-5819, 2012.
- [6] E. J. Candes, and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203-4215, 2005.
- [7] J. Han, Z. Sun, and H. Hao, " $\ell_0$ -norm based structural sparse least square regression for feature selection," *Pattern Recognition*, vol. 48, no. 12, pp. 3927-3940, 2015/12/01/, 2015.
- [8] P. Zhao, S. Liu, Y. Wang, and X. Lin, "An ADMM-Based Universal Framework for Adversarial Attacks on Deep Neural Networks," in Proceedings of the 26th ACM international conference on Multimedia, Seoul, Republic of Korea, 2018, pp. 1065-1073.
- [9] K. Koh, S.-J. Kim, and S. Boyd, "An interior-point method for large-scale  $\ell_1$ -regularized logistic regression," *Journal of Machine Learning Research*, vol. 8, no. Jul, pp. 1519-1555, 2007.