6578–6592 *Nucleic Acids Research, 2023, Vol. 51, No. 13* *Published online 29 May 2023*
https://doi.org/10.1093/nar/gkad450

Gene knockout inference with variational graph autoencoder learning single-cell gene regulatory networks

Yongjian Yang [1,*], Guanxun Li[2], Yan Zhong[3], Qian Xu[4], Bo-Jia Chen[5], Yu-Te Lin[6], Robert S. Chapkin[7] and James J. Cai [1,4,8]

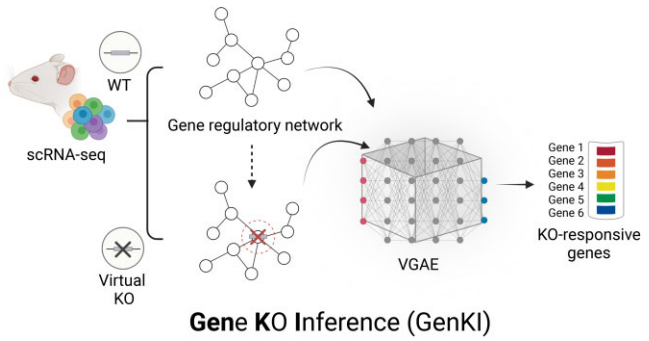
[1]Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA, [2]Department of Statistics, Texas A&M University, College Station, TX 77843, USA, [3]Key Laboratory of Advanced Theory and Application in Statistics and Data Science-MOE, School of Statistics, East China Normal University, 3663 North Zhongshan Road, Shanghai 200062, China, [4]Department of Veterinary Integrative Biosciences, Texas A&M University, College Station, TX 77843, USA, [5]Graduate Institute of Microbiology and Public Health, College of Veterinary Medicine, National Chung Hsing University, Taichung 402, Taiwan, [6]Graduate Institute of Biomedical Electronics and Bioinformatics, National Taiwan University, Taipei, Taiwan, [7]Program in Integrative & Complex Diseases, Department of Nutrition, Texas A&M University, College Station, TX 77843, USA and [8]Interdisciplinary Program of Genetics, Texas A&M University, College Station, TX 77843, USA

Received October 10, 2022; Revised May 02, 2023; Editorial Decision May 08, 2023; Accepted May 11, 2023

ABSTRACT

In this paper, we introduce Gene Knockout Inference (GenKI), a virtual knockout (KO) tool for gene function prediction using single-cell RNA sequencing (scRNA-seq) data in the absence of KO samples when only wild-type (WT) samples are available. Without using any information from real KO samples, GenKI is designed to capture shifting patterns in gene regulation caused by the KO perturbation in an unsupervised manner and provide a robust and scalable framework for gene function studies. To achieve this goal, GenKI adapts a variational graph autoencoder (VGAE) model to learn latent representations of genes and interactions between genes from the input WT scRNA-seq data and a derived single-cell gene regulatory network (scGRN). The virtual KO data is then generated by computationally removing all edges of the KO gene—the gene to be knocked out for functional study—from the scGRN. The differences between WT and virtual KO data are discerned by using their corresponding latent parameters derived from the trained VGAE model. Our simulations show that GenKI accurately approximates the perturbation profiles upon gene KO and outperforms the state-of-the-art under a series of evaluation conditions. Using publicly available scRNA-seq data sets, we demonstrate that GenKI recapitulates discoveries of real-animal KO experiments and accurately predicts cell type-specific functions of KO genes. Thus, GenKI provides an in-silico alternative to KO experiments that may partially replace the need for genetically modified animals or other genetically perturbed systems.

GRAPHICAL ABSTRACT



INTRODUCTION

Gene perturbation experiments are a proven powerful approach to elucidate the role of a gene in a biological process. Commonly used designs include gene knockout (KO) experiments with genetically altered animals and CRISPR gene perturbations. In a KO experiment, the function of a target gene is inferred by contrasting phenotypes between KO and wild-type (WT) animals and then identifying their differences. Often, gene expression profiles serve as a

*To whom correspondence should be addressed. Tel: +1 979 324 1715; Email: yjyang027@tamu.edu

quantitative phenotype at the molecular level (1). The recent advent of single-cell RNA sequencing (scRNA-seq) (2) allows the transcriptomic information from tens of thousands of cells to be gathered in parallel, and thus it greatly improves cellular phenotyping resolution. It has become a powerful method for molecular phenotyping and comparison in KO experiments.

Conventional KO experiments, often requiring significant amounts of experimental and animal resources, are labor-intensive and time-consuming (3). Recently developed techniques such as Perturb-seq (4) combine CRISPR perturbations and scRNA-seq to perform genetic screens, allowing gene function to be studied in many cells in a massively parallel manner. Nevertheless, the creation of large-scale CRISPR libraries presents a major technical challenge. For these reasons, computational tools serve as a possible alternative solution to facilitate or guide the experimental design through in-silico screening of perturbation responses. Such a computational tool would reduce the need for experimental measurements.

Indeed, several such computational tools (5–8) have been developed (Table 1). With only one exception—scTenifoldKnk (8), all these tools require extensive input data sets including outcomes of perturbation experiments or data from other modalities. scTenifoldKnk is the only protocol that does not require such expensive input data sets. Instead, it merely requires scRNA-seq data from the WT samples as its input and considers information from the gene regulatory network (GRN). The working principle of scTenifoldKnk is to simultaneously project WT and virtual KO single-cell gene regulatory networks (scGRNs) to a joint low-dimensional space and then calculate the projection differences of genes. However, the inference of scTenifoldKnk entirely relies on the WT scGRN, which is constructed using principal component (PC) regression from the WT scRNA-seq data. It is known that constructing high-quality scGRNs is technically challenging with respect to the presence of heterogeneous sources of noise (9). Also, a fully connected scGRN computed by the regression-based method may not correspond to real biological processes (10). A method that takes full advantage of scRNA-seq expression data and tolerates imperfect scGRN in a robust and unbiased manner is still lacking.

Here, we present GenKI (**Gen**e **K**O **I**nference), a virtual gene KO tool based on a variational graph autoencoder (VGAE) (11). GenKI simultaneously learns latent representations of scRNA-seq gene expression data of WT samples and the underlying scGRN responsible for observed phenotypes. The highly compressed representations of genes are then used for the subsequent inference. The scGRN can be constructed using the input gene expression data. GenKI propagates the transcriptomics information in the network during training and compares the WT data (including the expression data matrix and the scGRN) with its virtual KO counterpart to predict KO-responsive genes—i.e. genes functionally associated with or linked to KO gene. As a *de novo* inference tool, GenKI identifies KO-responsive genes without requiring prior knowledge of gene regulation or biological mechanisms.

The remainder of this paper is structured as follows: we first present an overview of the GenKI workflow and then compare its inference performance to several benchmarks using simulated data. Following these steps, we use publicly available scRNA-seq data sets (Supplementary Table S1) to predict KO-responsive genes and compare enriched functions of them with those introduced and validated in the original studies, to highlight the performance of GenKI in real-data applications. Next, we compare GenKI to the differential expression (DE) analysis. Finally, we study the robustness and scalability of GenKI.

MATERIALS AND METHODS

Simulated data sets and evaluation

The predefined GRNs were obtained from the GitHub repository of SERGIO (12) https://github.com/PayamDiba/SERGIO. The simulated data sets contained 100, 400, and 1200 genes (all containing 2700 cells), respectively. Edges in the predefined GRNs were treated as the ground truth. A random classifier that ranks genes by probabilities randomly drawn from a uniform distribution between 0 and 1, a classifier that ranks genes by the Pearson correlation with the KO gene, and scTenifoldKnk, which ranks genes by FC (used for the chi-squared test), were included for benchmarking purpose. For each data set, we randomly selected a target gene with more than ten edges and virtually knocked it out using GenKI and the other three benchmarks independently. Each run outputs a gene list with scores assigned by each method. *Roc_auc_score* and *average_precision_score* function from the Python package sklearn (v.1.1.1) were used to compute the Area Under Receiver Operating Characteristic (AUROC) and the average precision (AP) at each run for each method. We repeated the procedure above ten times for each data set. The simulated BEELINE (13) data sets were downloaded from Zenodo. GSD is the largest curated reference data set of BEELINE containing 19 genes and 2000 cells. Its underlying GRN was used to replace the GRN construction step in this evaluation. Since the ground truth GRN was known, we divided genes into two groups based on their shortest path to the KO gene, with the close neighbors group containing all genes within the two-hop neighborhood of the KO gene and the distant neighbors group containing all other genes. To compare the inference power of GenKI and scTenifoldKnk, we virtually knocked out each gene iteratively and obtained the scores of all the genes computed by both methods. For each method, we used the Wilcoxon Rank Sum test to quantify the difference in scores between the two groups of genes. A lower p-value indicates a larger difference, thus implying greater inference power of the method for detecting KO-responsive genes.

Processing of real data sets

The specifics and source of real scRNA-seq data sets used in this paper can be found in Supplementary Table S1. We performed regular preprocessing for all scRNA-seq data sets using Seurat (v.4.0.2) package (14). We first performed log normalization using the *NormalizeData* function. Highly

Table 1. Summary of existing virtual KO methods and feature comparison with GenKI

Name	Input data required	Method	Supervised / unsupervised	Description	Reference
scGen	scRNA-seq (WT and KO samples)	Transfer learning	Supervised	Train a variational autoencoder that learns to generalize the response of the cells in the training set of perturbations	(5)
CPA	scRNA-seq (KO samples)	Generative modeling	Supervised	Train an autoencoder with adversarial that decomposes the data into a collection of embeddings associated with the cell type, perturbation, and other external covariates to study combinatorial genetic perturbation	(6)
CellOracle	scRNA-seq and scATAC-seq (WT sample)	Graph-based modeling	Unsupervised	Simulate gene expressions in response to transcription factor (TF) perturbation by signal propagation through an inferred gene regulatory network	(7)
scTenifoldKnk	scRNA-seq (WT sample)	Manifold alignment	Unsupervised	Simultaneously project inferred WT and virtual KO gene regulatory networks to a joint low dimensional space	(8)
GenKI	scRNA-seq (WT sample)	VGAE	Unsupervised	Train a VGAE model that learns the latent gene representations of WT sample and virtually construct a virtual KO counterpart to discern similarity	This study

variable genes were selected using the *FindVariableFeatures* function (selection.method = 'vst') and by default, the top 3000 highly variable genes were included in subsequent analyses. We then standardized the data by the *ScaleData* function, and the resulting transformed data served as the gene expression profile for the GenKI input. Cell annotations from original studies were retained and used if provided.

Gene regulatory network construction

We constructed scGRNs using the PC regression method which was first proposed in scTenifoldNet (15). Let $\boldsymbol{X} \in \mathbb{R}^{p \times n}$ represent the scRNA-seq gene expression matrix of the WT samples, which contained gene expression levels for p genes in n cells. We used the PC regression method to build the scGRN denoted with its adjacent matrix $\boldsymbol{A}$. Specifically, each time one gene was selected as the response variable, while the remaining genes served as explanatory variables. Principal component analysis (16) was performed on the explanatory variables, and then we regressed the response variable on the first d leading PCs, where $d \ll n$. Next, we transformed the obtained regression coefficients of the d-leading PCs into the coefficients of the original explanatory variables, which should reflect the interaction strengths between the response gene and all other genes. In the final step, we assembled the coefficients of p regression models into a $p \times p$ adjacency matrix $\boldsymbol{A}$, where the (i, j) entry represents the regression coefficient of the i-th gene on the j-th gene. Therefore, $\boldsymbol{A}$ accumulates the interaction strength between each pair of genes.

Note that the output of this PC regression method is a fully connected scGRN, in which some links between genes might not correspond to real biological interactions, as in general, there are very few connections between TFs and genes (10). Therefore, for such an scGRN, we assumed that the edge is activated if the absolute value of its weight is greater than a certain threshold, i.e. edges with a greater weight are more likely to be the true regulatory relationships between genes than those with a lower weight. The average absolute weight between TF-target gene pairs constructed scGRNs was indeed significantly greater than that between random gene pairs, as described in (15). Based on these findings, for a particular scGRN, we filtered edges and, by default, conservatively only kept the top 15% of edges. A more thorough evaluation of the cutoff selection can be found in Supplementary Figure S1, which shows a heatmap of Spearman correlation coefficients between scores of Kullback–Leibler (KL) divergence given by GenKI across four different cutoffs. Within an optimal range of the cutoff, the ranking results given by GenKI were found to be highly consistent. However, we contend that extremely conservative choices of the cutoff would overlook potential links. Notably, we allow users to modify this default setting to accommodate their own biological scenarios. For example, those who believe their gene regulatory networks are scale-free are encouraged to use the poweRlaw package (17) to determine the best-fit threshold. Next, we converted the scGRN into an adjacent Boolean matrix as the input requested for the VGAE model of GenKI. As a result, although obtained without any information on TFs and their targets or knowledge of regulatory elements, these remaining edges could be deemed biologically responsive. By abuse of notations, we still denoted this new scGRN as $\boldsymbol{A}$ and we referred to it as the thresholded scGRN for later use. Although the filter step removed potential false positive edges, it inevitably introduced false negative findings, i.e. missing some truly connected edges. Therefore, we treated this thresholded scGRN as an incomplete network, and our goal was to reconstruct an scGRN from this incomplete network to learn the latent embeddings of nodes, namely, genes in our setting. This can be interpreted as a transductive link prediction task (18). Alternatively, users can supply their own GRN at this step to replace the PC regression-derived network.

VGAE model

The VGAE model used in GenKI is similar to the framework described in (11). It is made up of a two-layer graph convolutional network (GCN) encoder and an inner prod-

uct decoder. We utilized a two-layer GCN architecture because deeper graph convolutional networks are prone to over-smoothing (19). Recall that $\boldsymbol{X}$ is the gene expression matrix and $\boldsymbol{A}$ is the adjacent matrix, and we denoted the normalized adjacent matrix as $\tilde{\boldsymbol{A}} = \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}$, where $\boldsymbol{D} = diag(d_{11}, d_{22}, \ldots, d_{pp})$ is a diagonal matrix with entries $d_{ii} = \sum_{i=1}^{p} A_{ij}$, where $\boldsymbol{A}_{ij}$ is the (i, j)-th entry of the matrix $\boldsymbol{A}$. Then, the two-layer GCN is defined as:

$$GCN(\boldsymbol{X}, \boldsymbol{A}) = \tilde{\boldsymbol{A}} ReLU\left(\tilde{\boldsymbol{A}}\boldsymbol{X}\boldsymbol{W}_0\right)\boldsymbol{W}_1$$

where $ReLU(\boldsymbol{x}) = \max(0, \boldsymbol{x})$ is the activation function introduced in the first GCN layer, and $\boldsymbol{W}_0$ and $\boldsymbol{W}_1$ are parameters of the neural networks. We assumed that the data were generated by certain random processes involving an unobserved latent continuous random variable $\boldsymbol{Z}$. Let $p(\boldsymbol{Z})$ be the prior distribution of $\boldsymbol{Z}$, for which we chose a bivariate Gaussian distribution for convenience. For the encoder part, we introduced a recognition model $q(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{A}) = \prod_{i=1}^{p} q(\boldsymbol{z}_i|\boldsymbol{X}, \boldsymbol{A})$, where $q(\boldsymbol{z}_i|\boldsymbol{X}, \boldsymbol{A}) \sim \mathcal{N}(\mu_i, \Sigma_i)$, $\Sigma_i = diag(\sigma_{i1}^2, \sigma_{i2}^2)$ is a diagonal covariance matrix and

$$\boldsymbol{\mu} = \left(\mu_1^T, \cdots, \mu_p^T\right)^T = GCN_{\mu}(\boldsymbol{X}, \boldsymbol{A}),$$

$$log(\Sigma) = log\left(\left(\sigma_1^2, \cdots, \sigma_p^2\right)\right) = GCN_{\sigma^2}(\boldsymbol{X}, \boldsymbol{A}),$$

where $\sigma_i^2 = [\sigma_{i1}^2, \sigma_{i2}^2]^T$. For the decoder part, we used the inner product to reconstruct the scGRN $\boldsymbol{A}$ by

$$P\left(\boldsymbol{A}_{ij} = \boldsymbol{A}_{ji} = 1\right) = sigmoid\left(\boldsymbol{z}_i^T \boldsymbol{z}_j\right).$$

Here, by abuse of notations, $\boldsymbol{z}_i$ is the latent representation of the ith gene.

For any two distribution functions p and q, let $KL(p \parallel q) = \int p(x) log \frac{p(x)}{q(x)} dx$ be the KL divergence between p and q. The objective of the VGAE model is to maximize the evidence lower bound (ELBO):

$$\mathcal{L} = \mathbb{E}_{q(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{A})} \log p(\boldsymbol{A}|\boldsymbol{Z}) - \beta \cdot KL(q(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{A}) \parallel p(\boldsymbol{Z})),$$

where β is an adjustable hyperparameter that balances the independent constraints and reconstruction accuracy. Notice that here we adapted the loss from beta-VAE (20) and $\mathcal{L}$ would represent the standard ELBO when $\beta = 1$.

Hyperparameters, metrics and implementation

We randomly split the edges of a Boolean scGRN into three data sets for training (75%), validation (5%), and testing (20%). We labeled them as positive edges. Equal numbers of negative edges, composed of a set of fake edges not presented in the scGRN, were sampled for data balancing purposes. We used AUROC and AP to evaluate the model performance. We expected positive edges to have higher interaction probabilities compared to negative edges. Thus, the higher value of AP or AUROC would indicate better performance of training. To tune the hyperparameters, we performed random hyperparameters search of 100 trials by using the Tune module from the Python package Ray (21) (v.1.13.0). Specifically, the logarithm base 10 of hyperparameter β was sampled from a uniform distribution from $\{-5, -4, \ldots, -1\}$, the learning rate was sampled from a uniform distribution from $\{-4, -3, \ldots, -1\}$, and the weight decay of optimizer was sampled from a uniform distribution from $\{-7, -6, \ldots, -3\}$. To make our sampled hyperparameters more accurate, we multiplied each one by a scale factor randomly selected from integers 1 to 9. For each set of hyperparameters, we evaluated the model performance on the validation set and selected the hyperparameter set with the best performance based on the metrics AUROC and AP. Based on our experimental results, we set β of 1E-4 and weight decay of 9E-4 for all the data sets, and set learning rate of 7E-4 for the microglia, lung, intestine data set, 5E-3 for the COVID-19 data set. The maximum iteration number was set to 100, and early stopping was added when AP reached the maximum and began to decrease. The Adam optimizer (22) was used for all the trainings, and Xavier initialization (23) was used to initialize all the weights.

Determination of the rank of KO-responsive genes

After training the VGAE model using the WT data, for each fixed gene g, we obtained its latent distribution $N(\hat{\mu}_g, \hat{\sigma}_g^2)$, where $\hat{\mu}_g$ and $\hat{\sigma}_g^2$ were latent mean and covariance fitted by the VGAE model. We next fed the trained VGAE model with the virtual KO data and obtained the latent distribution of the g-th gene for the KO samples. Then, we calculated the KL divergence between these two normal distributions. The procedure was repeated for all genes. The top 5% of genes ranked by the KL divergence were preserved. Instead of using the raw ranks, we proposed a bagging-based method to improve the stability and accuracy of our inference. Specifically, each time we permutated the cell order of the WT gene expression matrix and obtained its corresponding virtual KO data. Without training a new model, we fed this pair of permutated WT and virtual KO data into our fitted VGAE model, calculated the KL divergence value for each gene, and bagged the top 5% of genes. We repeated this procedure 1000 times and compiled the genes which were bagged more than 95% times as KO-responsive genes.

Benchmarking GenKI's tolerance to random noise in gene expression profiles

To show the robustness of our method, we generated random noise in the log space, added it to gene expression profiles, and evaluated the training performance of GenKI. Specifically, for gene i in cell j, the regenerated expression $x'_{i,j}$ was defined as:

$$\frac{x'_{i,j}}{x_{i,j}} = 2^{\gamma}$$

where $\gamma \sim \mathcal{N}(0, \sigma^2)$ and $x_{i,j}$ represents the original expression. The fold change γ was used to approximate the noise level, which followed the normal distribution $\mathcal{N}(0, \sigma^2)$, whereas different σ values would result in different levels

of random noise. We conducted 30 independent runs with random splits of the data set at different noise levels.

Gene function annotation and function enrichment tests

Enrichr (24) with default setting was used for gene functional enrichment analyses. The protein-protein interaction enrichment tests were performed using the web tool of the STRING database (25). In the STRING network plots, isolated nodes were removed, and only edges labeled with confidence greater than the medium level were retrieved and shown. Enrichment *p*-values, which indicate whether input proteins have more interactions among themselves than what would be expected for a random set of protein-coding genes of the same size and degree distribution drawn from the genome, were computed with the default setting.

Prediction of KO gene's expression from WT cells with linear regression

For the microglia data set, a simple multivariate linear regression model was applied to evaluate the relationship between the KO gene Trem2 and other KO-responsive genes. Specifically, microglia cells' Trem2 expression profile was used as the response variable and the expression profiles of other genes as explanatory variables. The adjusted R^2 (coefficient of determination) was used to quantify how much variance of the KO gene can be explained by the other KO-responsive genes. In comparison, an equal number of the KO-responsive genes were randomly sampled as explanatory variables, and their R^2 was also calculated. This evaluation was repeated 30 times with different splits of the data set and random gene selections.

Differential gene expression analysis

DE analysis was performed using Scanpy (26)(v.1.9.1) function *rank_genes_groups* with the Wilcoxon rank-sum test. All parameters were set to default. Adjusted *p*-values were obtained after the Benjamini–Hochberg adjustment (27). DE genes were determined based on the condition of adjusted *p*-value < 0.05 and absolute log2(fold change) > 0.25. DE ranks of the DE genes were determined based on their adjusted *p*-value. To examine the expression level changes, for each data set, the KO-responsive genes and an equal number of randomly chosen unperturbed genes were used and their fold change (FC) of WT/KO was calculated. The absolute log2-transformed FC values of the KO-responsive genes and the unperturbed genes were used to perform the one-sided t-test.

RESULTS

The GenKI framework

The framework of GenKI is depicted in Figure 1. The pipeline starts with a single input, that is, the scRNA-seq gene expression matrix from WT samples of interest. For each virtual KO application, GenKI first constructs an scGRN from the WT gene expression data. The WT gene expression data matrix and the constructed WT scGRN are then used as input of WT data to train a VGAE model, which is a two-layer GCN encoder with an inner product decoder. The latent embedding of each node is defined to follow a bivariate Gaussian distribution. After training, the latent representations of genes under the WT setting are collected and the model with its weights is transferred. Next, to generate virtual KO data, the WT data is 'copied'. From the WT scGRN copy, the KO gene—i.e. the gene being knocked out for functional study—is virtually deleted. The deletion is achieved by setting the weight of all edges from and to the KO gene to zero. After the virtual deletion, the virtual KO data is generated, while the original WT scGRN remains untouched. The transferred model is fed with the virtual KO data to obtain the latent representations of genes under the KO setting. Two parameters, mean and covariance of each gene's latent distribution from the WT and KO settings are then collected to calculate the KL divergence between these two distributions. The higher the KL divergence value of a gene, the greater the impact of the KO on the gene. Finally, a bagging-based method is used to determine genes that tend to be significantly perturbed by the deletion of the KO gene. The enriched functions of these significantly perturbed genes (i.e. KO-responsive genes) are used to give prediction of the KO gene functions.

Performance of GenKI with simulated data

We used simulated data to evaluate the performance of our method (Figure 2A). To do so, we generated scRNA-seq data sets of different sizes (2700 cells with 200, 400, and 1200 genes, respectively) using single-cell expression simulator SERGIO (12). SERGIO's simulations were guided by predefined GRNs; therefore, the simulated scRNA-seq data sets had their underlying GRNs. Knowing these ground truths GRNs facilitated the performance evaluation of virtual KO methods, as genes linked with the KO gene were supposed to be perturbed by the KO and more likely to be KO-responsive genes. A good virtual KO tool should preferably identify those genes linked with the KO gene in the given GRN. For each of the simulated data sets, we applied GenKI and three other benchmarking methods, including scTenifoldKnk, with the same KO genes being knocked out (Materials and Methods). All the methods produced a ranked list of KO-responsive genes. Figure 2B shows the levels of AUROC for GenKI and other benchmarking methods. Figure 2C shows the levels of AP resulted from the same KO genes. Three additional ROC curves as examples of virtual KO experiments performed by GenKI and scTenifoldKnk for each data set are presented in Supplementary Figure S2. We found that GenKI outperformed all the other benchmark methods, including scTenifoldKnk, across all the data sets evaluated. We believe this is because GenKI incorporates information from both the gene expression matrix and GRN.

To demonstrate that GenKI learns higher-order neighborhood information from the underlying GRN through the VGAE model, which contributes to its greater performance than scTenifoldKnk, we systematically knocked out each of the 19 genes in the GSD network of BEELINE (13). In each virtual KO experiment, we obtained the perturbation scores of all genes. For a given KO gene, we used the Wilcoxon rank sum test to compare the difference in

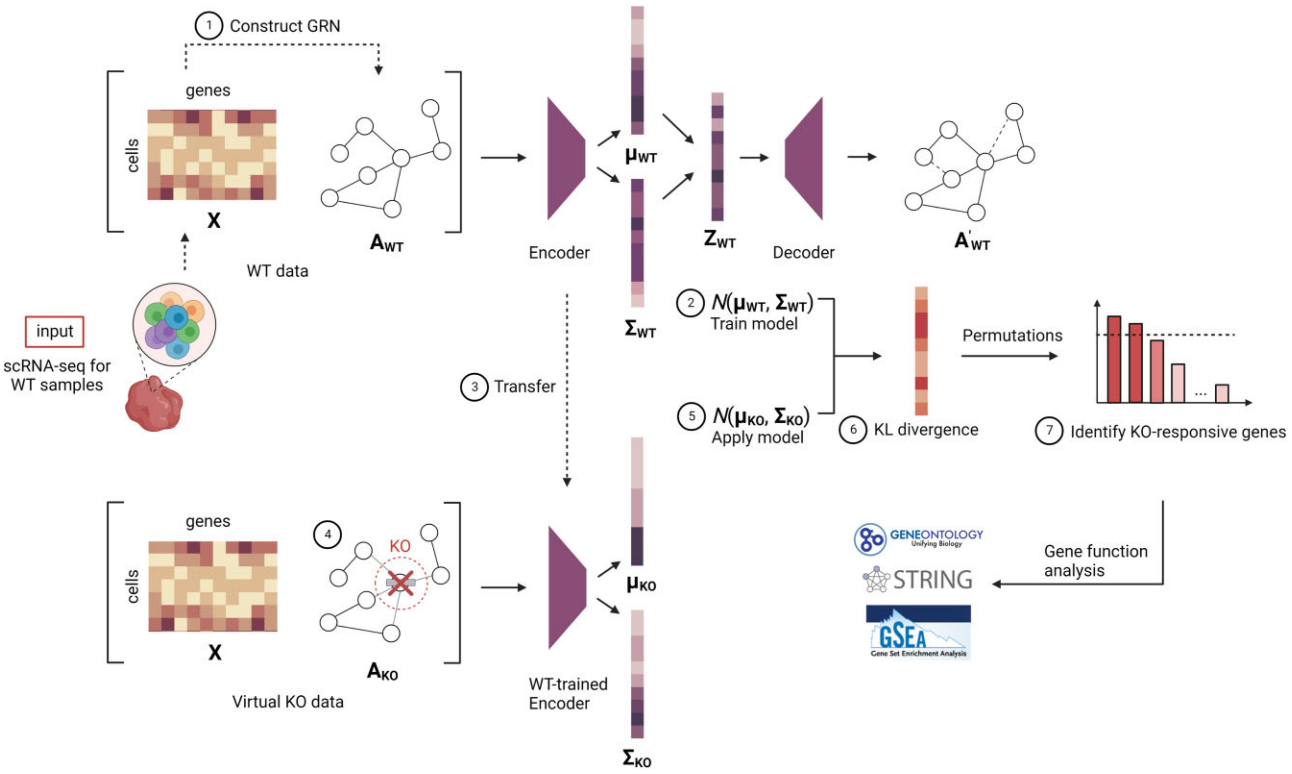


Figure 1. The pipeline contains seven steps: (1) construction of WT scGRN, (2) training VGAE model, (3) transfer the trained VGAE model, (4) construction of virtual KO data, (5) latent embeddings of WT and virtual KO data, (6) calculation of KL divergence, and (7) identification of KO-responsive genes for function annotation and analysis.

perturbation scores between the KO gene's two-hop neighbor genes and all the other distant genes (Materials and Methods). A smaller *p*-value indicates a greater inference power of the method for differentiation between these two groups. It is rational to expect that close neighbor genes have high perturbation scores. Compared to scTenifoldKnk, as expected, *p*-values obtained in GenKI are significantly lower (Supplementary Figure S3, Wilcoxon Rank Sum test, p-value < 0.05). This is attributed to manifold alignment in scTenifoldKnk only keeps track of the similarities between genes in the first-order neighborhood of GRN, while GenKI's two-layer GCN looks at similarities between genes up to the second-order neighborhood. This simulation study using the BEELINE network data also demonstrated that GenKI can take user input GRN as an optional rather than reconstructing GRN by its own.

Real-data GenKI analysis recapitulates findings of the trem2 KO experiment

GenKI, as a virtual KO tool, is expected to recapitulate the overall discoveries of real KO experiments. To validate its performance, we applied GenKI to several publicly available scRNA-seq data sets. The first data set was from the KO experiment conducted by Nugent *et al.* (28), in which scRNA-seq was performed with microglial cells isolated from $Trem2^{+/+}$ and $Trem2^{-/-}$ mice (Figure 3A). The study reported that Trem2 upregulates apolipoprotein E (Apoe) and other genes involved in cholesterol transport and metabolism, causing robust intracellular accumulation of a storage form of cholesterol upon chronic phagocytic activities (28). Trem2 is also known to regulate the expression of genes associated with cell damage response, lysosome and phagosome function, Alzheimer's disease, and oxidative phosphorylation (29). With this data set, we used the WT gene expression profile of 648 microglial cells as the input for GenKI and fed it along with the constructed scGRN to the VGAE model of GenKI.

We first evaluated the robustness of our model before performing prediction. The model robustness evaluation was performed to test the tolerance of the model by artificially adding different levels of random noise to the WT gene expression profile (Materials and Methods). A robust model would correctly capture the latent embeddings of genes, and thus more confidence for the inference regarding differences between WT and virtual KO samples. AUROC and AP were used to evaluate the reconstruction performance of the model. As shown in Supplementary Figure S4, our model was not compromised by high levels of noise ($\sigma = 1.5$), indicating the robustness of GenKI to the technical noise that naturally existed in the scRNA-seq data. We observed poorer performance under the conditions of very high levels of noise ($\sigma \geq 3$), which was expected as highly noisy gene expression profiles would mislead the training, and thus, the model could not be generalized to the testing data set. These results also indicated the lower bound of noiseless gene expression information needed to correctly reconstruct the scGRN and eventually infer the latent embeddings of genes.

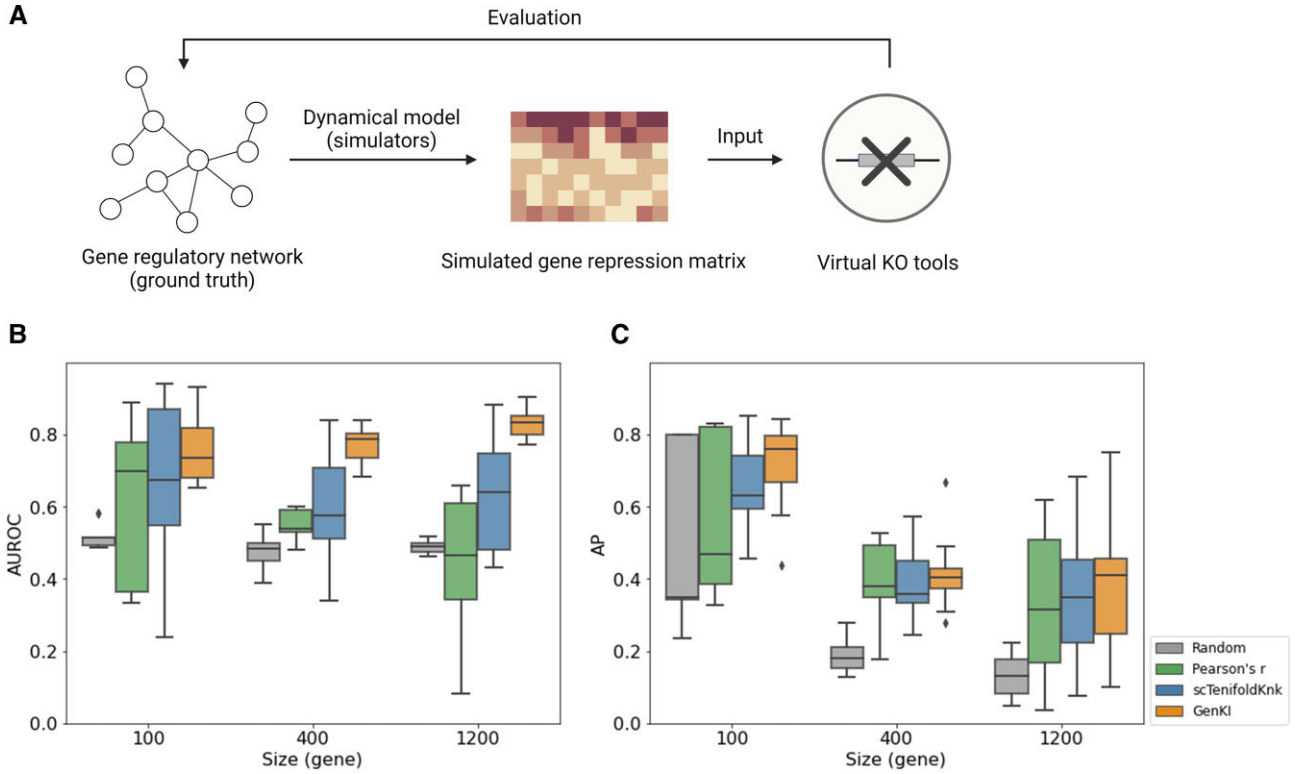


Figure 2. Three methods were included in the comparison including scTenifoldKnk and two baseline predictors, which are based on random rankings and Pearson's correlation, respectively (Materials and Methods). (**A**) The procedure of assessment of virtual KO tools using simulated data sets. (**B**) The levels of AUROC of virtual KO experiments using three simulated SERGIO data sets. (**C**) The levels of AP of virtual KO experiments using three simulated SERGIO data sets. Size represents the number of genes in each data set.

After the model robustness evaluation, we then trained the model and performed the virtual KO experiment. Specifically, we virtually knocked out Trem2 by removing all its edges in the scGRN of microglial cells and compared profiles of genes in the latent space between WT and virtual KO samples using KL divergence (Materials and Methods). The results of the analysis showed that 20 genes, including Trem2 itself, were detected as Trem2-KO responsive genes (Supplementary Table S2). Trem2 was ranked at the top of the KO-responsive genes, followed by Ctsd, the gene associated with lysosomal dysfunction (30), and Apoe, the key lipid transporter gene expressed in both the central nervous system and the periphery (31). Pathway enrichment analysis based on Enrichr (24) showed that Trem2-KO responsive genes were enriched with genes associated with *interleukin-2 signaling pathway*, *lysosome*, and *Alzheimer's disease* (Supplementary Table S3). Gene ontology (GO) enrichment analysis further ranked several enriched terms, including *macrophage activation involved in immune response* and *lipoprotein metabolic process*, on the top (Figure 3B and Supplementary Table S4). By modulating the macrophage transcriptome in adipose tissue, Trem2 was found to regulate blood cholesterol metabolism in obese mice, thereby indicating a connection between Trem2 and lipid metabolism (32). The overall results of our enrichment analyses revealed these functions of Trem2 with consistency. In addition, the Trem2-KO responsive genes were found to be biologically connected, as shown by the STRING interaction network (25) (Figure 3C, p-value < 0.01, STRING interaction enrichment test). Note that links in STRING interaction networks represent functional associations between genes. These associations include direct regulations as well as indirect interactions between genes or their products. Thus, our results suggest abundant functional connectivity between KO-responsive genes.

Next, we investigated whether Trem2's measurable gene expression was intrinsically interpreted by other KO-responsive genes. Indeed, the variance of Trem2 expression across cells could be substantially explained by the remainder of the KO-responsive genes (Figure 3D). We fitted a multivariable linear regression model by setting Trem2 as the response variable (Materials and Methods) and found that when using KO-responsive genes as explanatory variables, the adjusted R^2 of the model was significantly higher than when using an equal number of randomly selected genes as explanatory variables (p-value < 0.01, one-sided t-test). This finding suggests the KO gene and its KO-responsive genes predicated by GenKI tend to be transcriptionally associated.

Finally, we showed that one could not simply obtain the ranked gene list inferred by GenKI to identify KO-responsive genes using naïve network analysis metrics. We presented that, as an example, the KO-responsive genes could not be simply inferred either from ranking their gene expression or edge weight associated with the KO gene Trem2 in the inferred scGRN (Supplementary Figure S5).

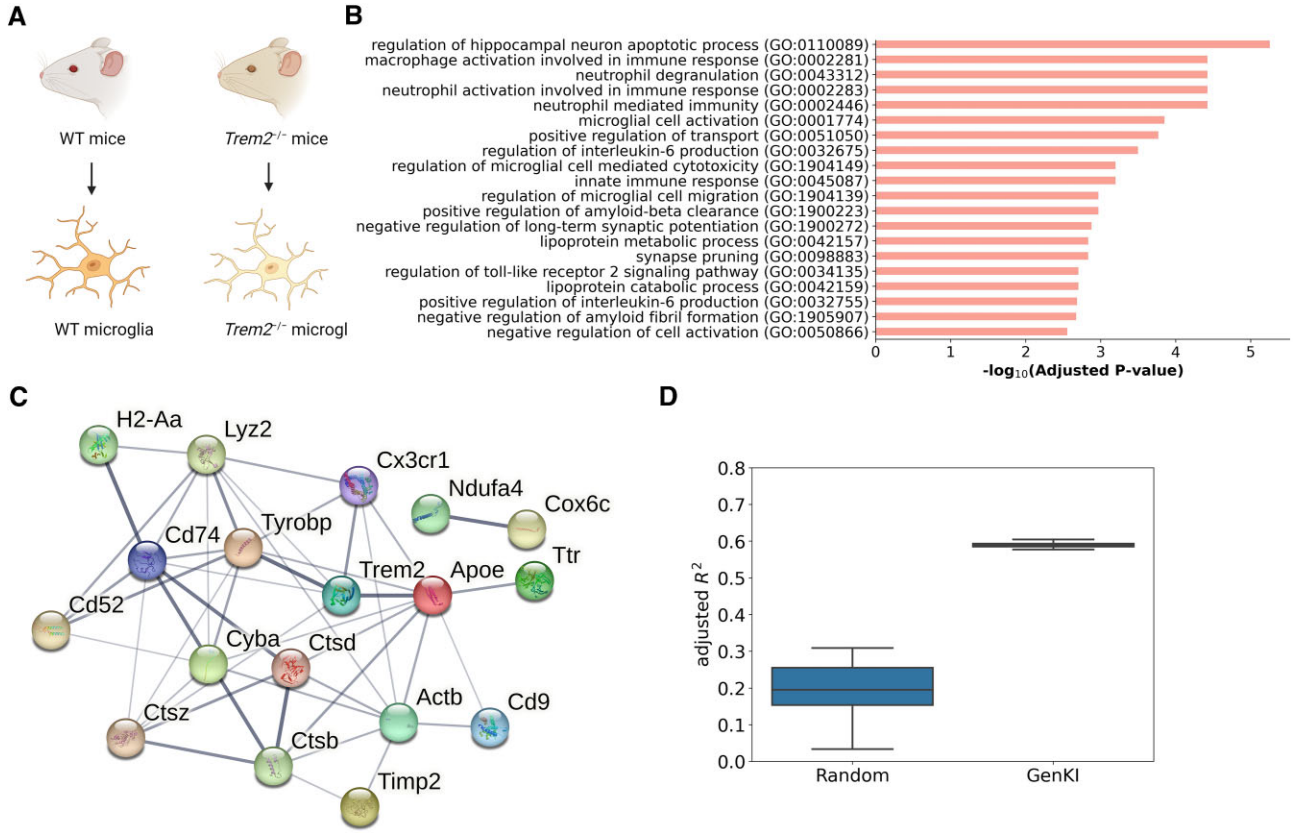


Figure 3. Trem2-KO responsive genes inferred by GenKI. (**A**) Illustration of Trem2-KO experiment generating the microglia data set. (**B**) GO terms significantly enriched in functions of Trem2-KO responsive genes. The $-\log_{10}$-transformed adjusted p-value indicates the strength of enrichment for each term. (**C**) STRING network of Trem2-KO responsive genes. Edge thickness indicates the strength of data support. (**D**) Adjusted R^2 score of the regression of expression levels by setting the Trem2 as a response variable and other KO-responsive genes as explanatory variables, compared to that of randomly selected genes as the explanatory variables.

The GenKI model nonlinearly learns both gene expression and edge weight information and infers from compressed embeddings of genes that it has learned. Thus, it ranks and infers the perturbed genes in a more comprehensive way than ranking methods based on any single observable property.

Collectively, our results shed light on Trem2-related functions by annotating the perturbed genes following its deletion. We showed that the inferred genes were functionally connected and, more importantly, predicted functions were consistent with those reported in the Trem2 studies.

Real-data GenKI analysis recapitulates findings of the nkx2-1 KO experiment

NK homeobox 2–1 (Nkx2-1) is highly expressed in lung epithelial cells and plays a crucial role in alveolar type 1 (AT1) cell development and maintenance (33). We collected the second scRNA-seq data from an *in vivo* KO experiment performed with lung epithelial cells of AT1 isolated from WT and Nkx2-1$^{-/-}$ mice. The study reported that the Nkx2-1 knocked-out AT1 cells lost their characteristics and abnormally turned into gastrointestinal fate (34). The study concluded that without Nkx2-1, developing AT1 cells lose three defining features—molecular markers, expansive morphology, and cellular quiescence—leading to alveolar simplification and lethality.

With this data set, we used the WT gene expression profile of 624 AT1 cells as the input for GenKI and virtually knocked out Nkx2-1 following the methods described above. The GenKI analysis discovered 82 KO-responsive genes (Supplementary Table S5). The KO gene, Nkx2-1, topped the gene list, followed by 13 marker genes of AT1 and AT2 cells offered by PanglaoDB (35), consistent with their downregulation in the Nkx2-1 mutant cells from the bulk RNA-seq experiment introduced in the original study. Previous research (36–39) discovered that Nkx2-1 binds to a group of AT1 cell-specific genes that regulate the cytoskeleton, membrane composition, and extracellular matrix. We found that Pdlim1, Clic5, Tuba1a, Krt8, Actn4, and Clu, which encode cytoplasmic proteins associated with the cytoskeleton, were highly ranked in our list. Ctsh, a gene involved in epithelial tube branching and lung morphogenesis (40), and a great number of genes related to membrane composition, such as Anxa1, were also observed among the KO-responsive genes. Two other significant genes, Napsa and Sftpc, collaborate with Ctsh to perform functions related to the collagen-containing extracellular matrix and alveolar lamellar body. Cldn33, Cldn7, and Epcam, which were shown to be involved in the apical junction complex (41), are in agreement with the observation that mutant AT1 cells

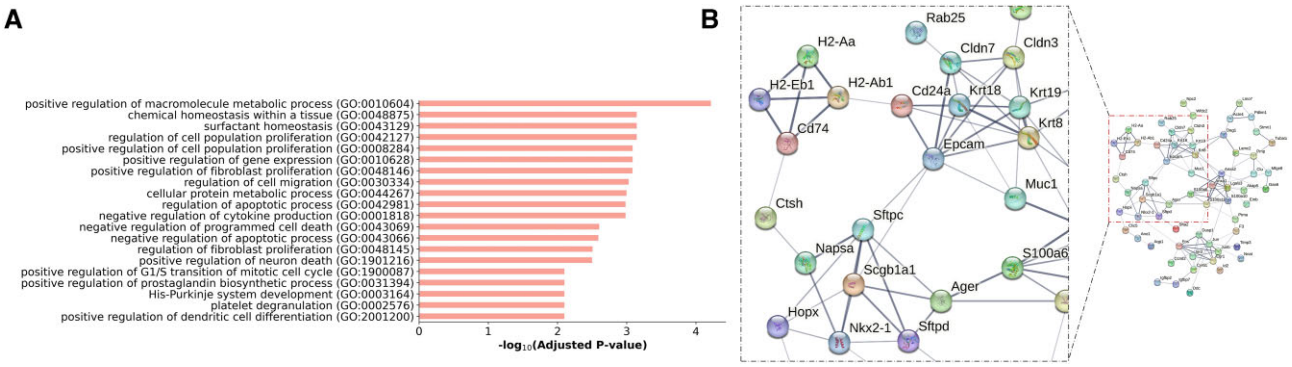


Figure 4. Nkx2-1-KO responsive genes inferred by GenKI. (**A**) GO terms significantly enriched in functions of Nkx2-1-KO responsive genes. The -log10-transformed adjusted *p*-value indicates the strength of enrichment for each term. (**B**) STRING network consists of Nkx2-1-KO responsive genes. The zoomed inset demonstrates a subnetwork module containing the KO gene.

form dense microvilli-like structures apically concluded in the original study.

GO enrichment analysis indicates these genes were enriched for functional categories led by *surfactant homeostasis* and *positive regulation of cell population proliferation* (Figure 4A, Supplementary Table S6), suggesting the role of Nkx2-1 in regulating surfactant production and suppressing AT1 cell proliferation validated in the study. HDAC3-dependent TGF-beta signaling is required for proper epithelium expansion and AT1 cell spacing (42,43), disruption of which significantly perturbed 13 genes from the list related to *TGF-beta regulation of extracellular matrix*. Additionally, due to mutant cells undergoing apoptosis, which was validated by staining in the original study, a few terms indicating the apoptotic process were observed. Many other GO terms, which are significant but not shown in Figure 4A, such as *epithelial tube branching involved in lung morphogenesis* and *epithelial cell morphogenesis* demonstrate the conclusion that Nkx2-1 defines the cell morphology of developing AT1 cells. The STRING interaction network of these 82 KO-responsive genes is shown in Figure 4B, suggesting that they tend to be biologically connected with a closely related functional relationship (p-value < 0.01, STRING interaction enrichment test).

Real-data GenKI analysis recapitulates findings of the hnf4a-smad4 double KO experiment

Using two real scRNA-seq datasets in which a single KO gene was knocked out, we have demonstrated the general performance of GenKI. Next, we investigated whether GenKI is able to virtually predict the effects of double KO (DKO). To accomplish this, we obtained a scRNA-seq data set performed with enterocytes isolated from WT and $Hnf4a^{KO}$-$Smad4^{KO}$ mice. The study reported that Smad4 and Hnf4 work together in a feed-forward loop to activate one another's expression and co-bind to differentiation gene regulatory regions. This feed-forward regulatory module supports and maintains enterocyte cell identity. Loss of this regulatory loop could impair enterocyte differentiation and destabilize enterocyte identity. This intersection of signaling and transcriptional regulation provides a framework for understanding the cellular plasticity of the regeneratable tissue (44).

In this experiment, we used the WT gene expression profile of 502 enterocytes as the input for GenKI and virtually knocked out Hnf4a and Smad4 simultaneously. 14 KO-responsive genes were reported by GenKI (Supplementary Table S7). The two KO genes, Hnf4a and Smad4, topped the gene list, followed by regenerating islet-derived 1 (Reg1), a regulator of cell growth that is required to generate and maintain the villous structure of the small intestine (45). Hnf4a regulates intestinal epithelium homeostasis and intestinal absorption of dietary lipids (46). Loss of this gene is likely to disrupt glucose metabolism, which is regulated by intestinal Reg3b (47), another significant gene. Also included was Gcg, a gene that may modulate gastric acid secretion and gastro-pyloro-duodenal activity (48).

Figure 5B depicts the STRING interaction network of these KO-responsive genes. Despite the network being split into two parts under the default setting, we found two disconnected genes, Dmbt1 and Gsta1, were indeed functionally connected—GO enrichment analysis indicates that these two genes were enriched for *epithelium cell differentiation* (Figure 5A, Supplementary Table S8), indicating the loss of enterocytes differentiation after the DKO measure discovered in the original study. Thus, these genes are statistically (p-value < 0.01, STRING interaction enrichment test) and biologically linked. Other significant GO terms, such as *negative regulation of cell growth* and *carbohydrate homeostasis* correlated with results of the enterocytes study, have also been illustrated in our analysis. Together, this virtual DKO experiment demonstrates that perturbation effects from multiple KO genes are nonlinearly accumulable and can be recapitulated by GenKI.

Are KO-responsive genes more likely to be differentially expressed?

We next set out to answer the following question: do KO-responsive genes exhibit differential expression? We first analyzed the expression level changes of predicted KO-responsive genes by comparing them to unperturbed genes across data sets (Materials and Methods). We discovered that the KO-responsive genes predicted by GenKI tend to have greater absolute FC values than unperturbed genes (Supplementary Figure S6, p-value < 0.05, one-sided t-test). Thus, we came to the conclusion that KO-responsive genes

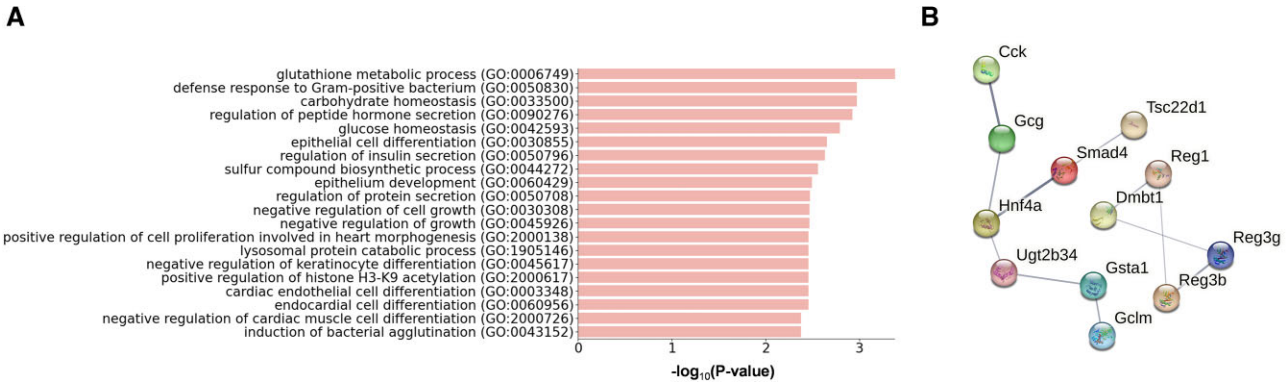


Figure 5. Hnf4a & Smad4-KO responsive genes inferred by GenKI. (**A**) GO terms significantly enriched in functions of inferred Hnf4a-Smad4-KO responsive genes. The $-\log_{10}$-transformed *p*-value indicates the strength of enrichment for each term. (**B**) STRING subnetwork consists of Hnf4a-Smad4-KO responsive genes.

predicted by GenKI are more likely to be differentially expressed.

Next we showed that GenKI analysis is different from the DE analysis: KO-responsive genes are not necessarily DE genes. We examined this by comparing the real KO data of each data set to their WT, where 126, 1129 and 1215 DE genes were identified, respectively (Materials and Methods). The overlap between the predicted KO-responsive genes and the top-ranked 50 DE genes in each data set is shown with a Venn diagram in Figure 6 left panel.

The eight overlapping genes of the microglia data set includes Trem2 and other lipoproteins-forming genes like Apoe (Figure 6A, left). The 17 intersection genes of the lung data set contain Nkx2-1, the pulmonary surfactant Sttpc and several AT1 and AT2 cell markers (Figure 6B, left). Thus, GenKI could be used to predict some of the DE genes. In addition, GenKI identified KO-responsive genes that are not ranked highly by the DE method. By using a barcode enrichment plot (Figure 6, right panel), we were able to visualize the exact locations of the KO-responsive genes across the DE ranks, with each black stick denoting a 'hit' of the KO-responsive genes. H2-Aa, a recognized DE gene but not ranked highly (82nd shown in Figure 6A, right), is known to function with other genes such as Cd74, Ctsb, and Ctsd in histocompatibility complex (MHC) class II presentation (49). Napsa, which functions together with Nkx2-1 and Ctsh in the processing of pneumocyte surfactant precursors, was likely to be underestimated (763rd, out of scope in Figure 6B, right). The double KO genes Hnf4a and Smad4, which were not included in the intersection of Figure 6C left, weakly ranked 108th and 235th, respectively (Figure 6C, right). These perturbed genes were prioritized by GenKI, whereas the DE analysis did not. GenKI further identified KO-responsive genes that are not DE genes. These genes are likely to be at least as important as the DE genes, if not more. For example, concerning the microglia data set, Ctsd is one leading gene involved in cholesterol metabolism (50), and Cx3cr1 and Tyrobp play an important role in macrophage activation (51–53). All of them were not the DE genes.

Do DE genes appear more adjacent to KO-responsive genes in a scGRN? To answer this question, we performed the STRING network analysis by combining the top-ranked DE genes with the KO-responsive genes using the microglia data set as an example. The outcome is depicted in Supplementary Figure S7, showing that 23 out of 42 DE genes are directly or indirectly linked to the KO genes. That is to say, in this given case, more than half of DE genes might be functionally involved in the perturbed KO gene network identified by GenKI.

Utilizing DE and GenKI analyses in a complementary manner might be a good idea. To illustrate our point, we applied seven different DE analysis methods and settings to the lung data and summarized the number of DE genes detected and their intersection with GenKI-identified genes (Supplementary Table S9). We found that the results of DE analysis were largely depend on what method was selected to use and what fold-change and *p*-value cutoffs were set, and the functional interpretation of the DE analysis results was also depended whether up- and down-regulated genes are pooled together. In general, we found different DE methods with varying model assumptions and thresholds could not converge to a consensus set of DE genes. The number of DE genes and their rankings changed greatly depending on many technical factors as mentioned. Furthermore, most DE methods with default settings produce excessive numbers of DE genes, making downstream functional enrichment analysis difficult and obscuring true signals caused by the perturbation itself to be detected. GenKI, on the other hand, as a method independent of DE methods, provides additional evidence for gene functions. Most of GenKI's KO-responsive genes overlapped with DE genes regardless of the DE method. With the default setting, GenKI produced fewer significant genes than DE methods, which may improve the interpretability of gene function. In this sense, we are not developing an alternative to DE, but rather a complementary technique that produces more targeted results.

Real-data GenKI analysis predicts function of key transcriptional factor STAT1

Above we have validated GenKI performance by comparing the inference results to DE genes using three scRNA-seq data sets that all included WT and KO groups. We questioned whether GenKI is able to reveal gene functions of

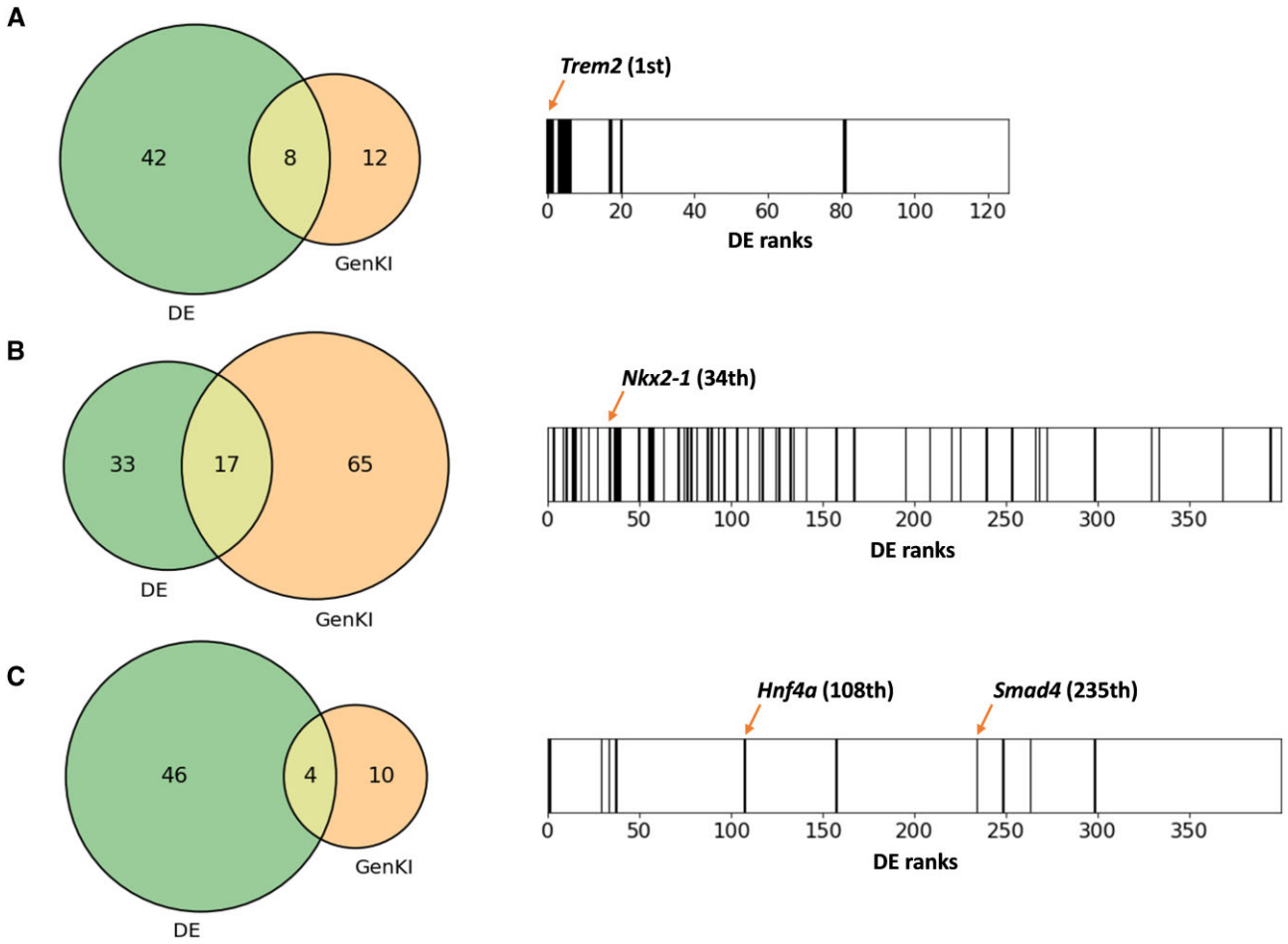


Figure 6. Venn diagrams and barcode enrichment plots showing the intersection and differences between the KO responsive genes given by GenKI and DE genes. Venn diagram and barcode enrichment plot of (**A**) microglia data set, (**B**) lung data set and (**C**) intestine data set. All the numbers of overlapped genes were significantly greater than random expectations (p-value < 10E-05, hypergeometric test).

any target gene from a standalone WT scRNA-seq data set without pairing it with a KO counterpart, which should be a more common occurrence when using virtual KO tools. We obtained a data set from a study of 19 patients with severe coronavirus disease 2019 (COVID-19) (54). It contains 8920 cells collected from nasopharyngeal and bronchial samples. The study found that epithelial cells of COVID-19 patients showed an average three-fold increase in expression of the SARS-CoV-2 entry receptor ACE2, and signal transducer and activator of transcription 1 (STAT1), a central transcription factor of the interferon response, was among the top predictors for ACE2 expression. Previous research also shows that STAT1 is critical for virus clearance and disease resolution, and STAT1-KO mice have impaired interferon gamma (IFNG) signaling (55). In this virtual KO experiment, we focused on a subpopulation of pulmonary epithelial cells differentiating from immature secretory cells to ciliated cells. The original study demonstrated an alternative differentiation pathway leading from immature secretory cells directly into ciliated cells mediated by these IFNG-responsive epithelial cells, suggesting that this direct differentiation pathway is dependent on the interferon response (54).

We virtually knocked out STAT1 in these epithelial cells. Firstly, we validated the robustness of our model by artificially adding different levels of random noise to the gene expression profile (Supplementary Figure S8). The GenKI analysis identified 28 STAT1-KO responsive genes (Supplementary Table S10). STAT1 was ranked at the top, followed by three human leukocyte antigen (HLA) genes (HLA-DRA, HLA-DRB1, HLA-DPA1), which are known to encode Class II major histocompatibility complex (class II MHC). Class II MHC, which are reported to be highly expressed only in antigen-presenting cells (APC), is induced in other cell types as well by inflammation or IFNG (56). Moreover, lysosomes are required for lysis of the protein into peptides for class II MHC presentation to the immune cells (57). In our inferred gene list, the lysosome-related genes CTSB, CTSD, and CSTB were included, and were related to the antigen-presenting process. Previous research indicates that the nuclear factor-κB (NF-κB) can be activated by IFNG (58). This is consistent with genes in the list believed to participate in NF-κB-related pathways and inflammation. For example, ANXA1 is reported to have anti-inflammation activity in lung endothelial cells and is able to prevent lung fibrosis (59). GPX1 participates in the NF-

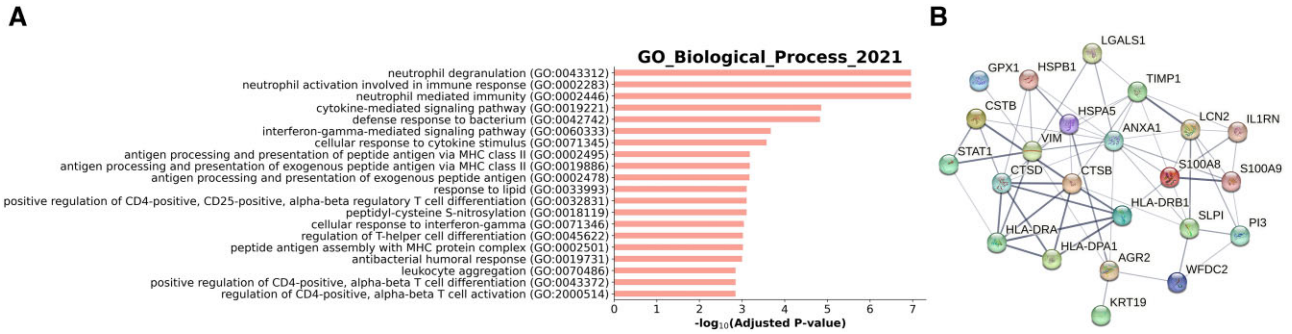


Figure 7. STAT1-KO responsive genes inferred by GenKI. (**A**) GO terms significantly enriched in functions of STAT1-KO responsive genes. The $-\log_{10}$-transformed adjusted *p*-value indicates the strength of enrichment for each term. (**B**) STRING network consists of STAT1-KO responsive genes.

κB pathway and is crucial for respiratory virus infection (60). S100 family proteins are well-characterized for their function in inflammation and innate immunity (61). Additionally, S100 proteins are damage-associated molecular patterns (DAMP) that promote inflammation by binding to the pattern-recognition protein (PRR) (62). HSPB1 and HSPB5 belonging to DAMP are also listed.

The result of GO enrichment analysis is presented in Figure 7A and Supplementary Table S11. The neutrophil-related pathways were ranked at the very top in the enrichment analysis, suggesting the communication between IRC and neutrophils, which is in agreement with the finding of the original study (54). The interferon-gamma, class II MHC antigen-presenting, NF-κB, and innate immunity-related pathways were also detected by the enrichment analysis. Thus, these results strongly suggest that the GenKI is able to accurately predict the potential perturbed genes and their shared functions. We further analyzed 28 genes using STRING to understand their interaction (Figure 7B). The resulting subnetwork, which contains significantly more interactions than expected (p-value < 0.01, STRING interaction enrichment test), again suggests that these genes are closely connected due to their shared biological functions. This virtual KO study demonstrates that GenKI can reliably predict gene functions and infer the molecular phenotypic consequences of genes of interest validated by previous studies without the need for an actual KO experiment.

GenKI is robust and scalable

To assess the robustness of GenKI inference, we collected scRNA-seq data (63) from mouse neurons with Rett syndrome (RTT), a severe neurodevelopmental disorder. Mutations in Mecp2, a transcriptional repressor required to maintain normal neuronal functions, are known to cause RTT (64,65). This data set contains two replicates with 2054 and 2156 neurons, respectively. We independently analyzed these two replicates with GenKI, in which we virtually knocked out the same KO gene Mecp2. Given the high similarity of these two biological replicates, GenKI would be robust if it generated roughly equivalent gene ranks across them. Indeed, we found high consistency between the rankings of the two reported rank lists (Spearman's correlation coefficient $\rho = 0.82$).

Finally, we evaluated the computation efficiency of GenKI. Supplementary Figure S9 shows the results of the analysis, comparing the total running time with respect to different sizes of input scRNA-seq data sets. The running time for GenKI consists of scGRN construction, training, and inference. We simulated four random data sets at different scales for this comparison. Without using GPUs, GenKI exhibited a 2.8- to 4.9-fold faster running speed than scTenifoldKnk tested on equivalent hardware. GenKI is expected to run even faster by enabling the fast GPU implementation optimized by PyTorch Geometric (66).

DISCUSSION

In this study, we showcased the functionality and performance of GenKI in virtual KO experiments. We first evaluated the inference performance of GenKI using simulated data sets (SERGIO and BEELINE). Next, we used scRNA-seq data sets generated in real KO experiments to show that GenKI could predict gene functions by identifying and annotating KO-responsive genes. The functional predictions were found to be consistent with original studies in which WT and KO scRNA-seq data sets were generated.

Our main contribution in this work is to provide a neural network-based virtual KO analytical tool, which encodes the gene expression matrix to a latent space given its underlying scGRN. To the best of our knowledge, GenKI is the first virtual KO tool using a graph-based generative model to infer KO-responsive genes and their shared functions. Several computational tools have been developed for similar purposes to predict the effects of genetic perturbation using single-cell data. scGen (5) and CPA (6), both running in a supervised manner, require massive training data labeled with various perturbations to train their autoencoder-based models. CellOracle (7) can simulate gene expression in response to TFs perturbation by signal propagation through its inferred scGRN. However, this simulation is linear and does not quantify the level of perturbation at individual gene level. More importantly, it requires scATAC-seq data along with the corresponding scRNA-seq data to build the scGRN prior to making such an inference, which may limit its application. scTenifoldKnk (8) is the only virtual KO tool with the identical input requirements as GenKI. Like GenKI, scTenifoldKnk only requires WT scRNA-seq data for its prediction analysis. It employs manifold align-

ment (67) to project WT and virtual KO scGRNs to a joint low-dimensional space and calculate the differences between them. Given the minimalistic design, GenKI shares with scTenifoldKnk several key advantages such as being species agnostic—that is, they both work with scRNA-seq data from humans and animal models alike. By applying these tools directly to human data instead of surrogate animals, researchers may avoid pitfalls caused by overextending their conclusions from animal models to humans. Additionally, both GenKI and scTenifoldKnk allow any gene to be knocked out, regardless as to whether the KO genes are functionally vital or not. Knocking out a vital gene tends to cause fatal consequences and is, therefore, impractical to generate animal models for its KO.

GenKI outperforms scTenifoldKnk in the following aspects. First, scTenifoldKnk only utilizes the WT scGRN, while GenKI takes into account both the WT gene expression profile and scGRN. Second, the VGAE model, which consists of two message passing layers, collects information up to the second-order neighborhood of the network. In contrast, manifold alignment adopted in scTenifoldKnk only maintains the similarity of directly connected neighbors of the network, which results in different levels of inference power. In addition, GenKI shows better scalability, being able to process tens of thousands of cells within a reasonable time. Once the GenKI model is trained, the model can be reused for virtual KO of any genes in the data. While in order to do the same, scTenifoldKnk must re-solve the manifold alignment problem for each KO gene by eigen decomposition, which is considered computationally intensive and time-consuming. Last, GenKI avoids a pitfall in numerical computation in scTenifoldKnk. scTenifoldKnk performs a virtual KO experiment by removing the edges of a KO gene in the scGRN, which results in an asymmetric Laplacian matrix containing negative values. This potentially leads to eigenvectors of the Laplacian matrix with imaginary parts when solved by eigen decomposition. scTenifoldKnk practically adds 1 to all entries in obtained scGRNs to guarantee that all the entries are positive and only uses the real parts of obtained eigenvectors. GenKI's architecture allows it to bypass this problem because it employs neural networks to solve the optimization problem, which has been shown to be numerically more stable than eigen decomposition (68).

We addressed the question that end users may often have, i.e. 'Are KO-responsive genes more likely to be differentially expressed?' DE analysis, followed by gene function enrichment analysis, are often used to identify the perturbed gene expression programs in order to understand the function of the KO gene. The problem is that the perturbation effect of the KO gene may propagate on the underlying network but may not direct reflected as observable and measurable changes in gene expression. GenKI, on the other hand, works on scGRNs directly to leverage unobservable network-level information—GenKI identifies perturbed genes through modelling underlying networks. Therefore, in contrast to DE analysis that can only detect perturbed genes with significant expression level changes, GenKI is likely to detect perturbed genes even there are less or no significant expression level changes. Perturbed genes without expression level changes are not uncommon. For instance, given a gene that is under control of multiple regulators, even if one of its regulators is knocked out, the remaining regulators may still be functioning to compensate and stabilize the given gene's expression. Additionally, with the default setting, GenKI produced fewer significant genes than a typical DE analysis, which may improve the interpretability of gene function. In conclusion, GenKI is not an alternative to DE analysis, but rather a complementary technique that produces more targeted results.

The limitations of GenKI are mostly inherited from it being virtual. GenKI cannot be used to predict the regulatory direction of KO-responsive genes, which is important in learning cell responses to external stimuli (69). If future refinements enable directional predictions, GenKI may improve with its potential ability to simulate the effect of overexpression. Also, GenKI, like scTenifoldKnk, currently performs a virtual KO experiment by removing all the edges of a KO gene in the WT scGRN. This action might be naïve given the complexity of a biological system. A virtual KO scGRN could be better modeled by simulating the virtual KO effect in a more probabilistic manner. Alternatively, there are many available priors involved in many different types of KO; hence a Bayesian treatment may facilitate the KO inference. GenKI is also inapplicable to bulk RNA-seq data, as genes in such data lose their variability in terms of gene expressions, which results difficulty in scGRN construction using PC regression and assigning expression values to node attributes in a graph. Recent advances in cell pseudo-temporal ordering enable us to map the underlying scGRNs throughout time (70,71) and eventually learn temporal KO effects including cell-cell communication (72) in a dynamic manner. GenKI can be improved by incorporating a dynamic inference module to investigate such effects on cell or organ development.

DATA AVAILABILITY

The sources of data sets underlying this article can be found in Supplementary Table S1. No new data were generated in support of this research. A Python implementation of the GenKI framework is available at https://github.com/yjgeno/GenKI (permanent DOI: 10.5281/zenodo.7915654).

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We would like to express our appreciation to Dr. Daniel Osorio for his valuable technical support and feedback on the manuscript writing. We also acknowledge the use of advanced computing resources provided by Texas A&M High Performance Research Computing in conducting parts of this research.

FUNDING

National Institute of Environmental Health Sciences [P30 ES029067]; National Cancer Institute [R35 CA197707, RO1 CA245514]; Allen Endowed Chair in Nutrition &

Chronic Disease Prevention (to R.S.C.); U.S. Department of Defense [GW200026 to J.J.C.]. Funding for open access charge: Texas A&M University.
Conflict of interest statement. None declared.

REFERENCES

1. Quake,S.R. (2021) The cell as a bag of RNA. *Trends Genet.*, **37**, 1064–1068.
2. Hwang,B., Lee,J.H. and Bang,D. (2018) Single-cell RNA sequencing technologies and bioinformatics pipelines. *Exp. Mol. Med.*, **50**, 1–14.
3. Hall,B., Limaye,A. and Kulkarni,A.B. (2009) Overview: generation of gene knockout mice. *Curr. Protoc. Cell Biol.*, **Chapter 19**, 19.12.1–19.12.17.
4. Dixit,A., Parnas,O., Li,B., Chen,J., Fulco,C.P., Jerby-Arnon,L., Marjanovic,N.D., Dionne,D., Burks,T., Raychowdhury,R. *et al.* (2016) Perturb-Seq: dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell*, **167**, 1853–1866.
5. Lotfollahi,M., Wolf,F.A. and Theis,F.J. (2019) scGen predicts single-cell perturbation responses. *Nat. Methods*, **16**, 715–721.
6. Lotfollahi,M., Susmelj,A.K., De Donno,C., Ji,Y., Ibarra,I.L., Wolf,F.A., Yakubova,N., Theis,F.J. and Lopez-Paz,D. (2021) Cold Spring Harbor Laboratory.
7. Kamimoto,K., Hoffmann,C.M. and Morris,S.A. (2020) Cold Spring Harbor Laboratory.
8. Osorio,D., Zhong,Y., Li,G., Xu,Q., Yang,Y., Tian,Y., Chapkin,R.S., Huang,J.Z. and Cai,J.J. (2022) scTenifoldKnk: an efficient virtual knockout tool for gene function predictions via single-cell gene regulatory network perturbation. *Patterns (N Y)*, **3**, 100434.
9. Eraslan,G., Simon,L.M., Mircea,M., Mueller,N.S. and Theis,F.J. (2019) Single-cell RNA-seq denoising using a deep count autoencoder. *Nat. Commun.*, **10**, 390.
10. Ye,J. and Liu,J. (2012) Sparse methods for biomedical data. *SIGKDD Explor*, **14**, 4–15.
11. Kipf,T.N. and Welling,M. (2016) Variational graph auto-encoders. arXiv doi: https://arxiv.org/abs/1611.07308, 21 November 2016, preprint: not peer reviewed.
12. Dibaeinia,P. and Sinha,S. (2020) SERGIO: a Single-Cell Expression Simulator Guided by Gene Regulatory Networks. *Cell Syst.*, **11**, 252–271.
13. Pratapa,A., Jalihal,A.P., Law,J.N., Bharadwaj,A. and Murali,T.M. (2020) Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat. Methods*, **17**, 147–154.
14. Hao,Y., Hao,S., Andersen-Nissen,E., Mauck,W.M. 3rd, Zheng,S., Butler,A., Lee,M.J., Wilk,A.J., Darby,C., Zager,M. *et al.* (2021) Integrated analysis of multimodal single-cell data. *Cell*, **184**, 3573–3587.
15. Osorio,D., Zhong,Y., Li,G., Huang,J.Z. and Cai,J.J. (2020) scTenifoldNet: a machine learning workflow for constructing and comparing transcriptome-wide gene regulatory networks from single-cell data. *Patterns (N Y)*, **1**, 100139.
16. Jolliffe,I.T. and Cadima,J. (2016) Principal component analysis: a review and recent developments. *Philos. Trans. A Math. Phys. Eng. Sci.*, **374**, 20150202.
17. Alstott,J., Bullmore,E. and Plenz,D. (2014) Powerlaw: a Python package for analysis of heavy-tailed distributions. *PLoS One*, **9**, e85777.
18. Ravindra,N., Sehanobish,A., Pappalardo,J.L., Hafler,D.A. and van Dijk,D. (2020), *Proc. ACM Conf. Health Inference Learn.*, pp. 121–130.
19. Yang,C., Wang,R., Yao,S., Liu,S. and Abdelzaher,T. (2020) Revisiting over-smoothing in deep GCNs. arXiv doi: https://arxiv.org/abs/2003.13663, 30 March 2020, preprint: not peer reviewed.
20. Higgins,I., Matthey,L., Pal,A., Burgess,C., Glorot,X., Botvinick,M., Mohamed,S. and Lerchner,A. (2016) beta-vae: learning basic visual concepts with a constrained variational framework.
21. Liaw,R., Liang,E., Nishihara,R., Moritz,P., Gonzalez,J.E. and Stoica,I. (2018) Tune: a research platform for distributed model selection and training. arXiv doi: https://arxiv.org/abs/1807.05118, 13 July 2018, preprint: not peer reviewed.
22. Kingma,D.P. and Ba,J. (2014) Adam: a method for stochastic optimization. arXiv doi: https://arxiv.org/abs/1412.6980, 22 December 2014, preprint: not peer reviewed.
23. Glorot,X. and Bengio,Y. (2010) In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, pp. 249–256.
24. Kuleshov,M.V., Jones,M.R., Rouillard,A.D., Fernandez,N.F., Duan,Q., Wang,Z., Koplev,S., Jenkins,S.L., Jagodnik,K.M., Lachmann,A. *et al.* (2016) Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Res.*, **44**, W90–W97.
25. von Mering,C., Jensen,L.J., Snel,B., Hooper,S.D., Krupp,M., Foglierini,M., Jouffre,N., Huynen,M.A. and Bork,P. (2005) STRING: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Res.*, **33**, D433–D437.
26. Wolf,F.A., Angerer,P. and Theis,F.J. (2018) SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.*, **19**, 15.
27. Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Series B Stat. Methodol.*, **57**, 289–300.
28. Nugent,A.A., Lin,K., van Lengerich,B., Lianoglou,S., Przybyla,L., Davis,S.S., Llapashtica,C., Wang,J., Kim,D.J., Xia,D. *et al.* (2020) TREM2 regulates microglial cholesterol metabolism upon chronic phagocytic challenge. *Neuron*, **105**, 837–854.
29. Shi,Y. and Holtzman,D.M. (2018) Interplay between innate immunity and Alzheimer disease: APOE and TREM2 in the spotlight. *Nat. Rev. Immunol.*, **18**, 759–772.
30. Reifschneider,A., Robinson,S., van Lengerich,B., Gnorich,J., Logan,T., Heindl,S., Vogt,M.A., Weidinger,E., Riedl,L., Wind,K. *et al.* (2022) Loss of TREM2 rescues hyperactivation of microglia, but not lysosomal deficits and neurotoxicity in models of progranulin deficiency. *EMBO J.*, **41**, e109108.
31. Li,R.Y., Qin,Q., Yang,H.C., Wang,Y.Y., Mi,Y.X., Yin,Y.S., Wang,M., Yu,C.J. and Tang,Y. (2022) TREM2 in the pathogenesis of AD: a lipid metabolism regulator and potential metabolic therapeutic target. *Mol. Neurodegener.*, **17**, 40.
32. Jaitin,D.A., Adlung,L., Thaiss,C.A., Weiner,A., Li,B., Descamps,H., Lundgren,P., Bleriot,C., Liu,Z., Deczkowska,A. *et al.* (2019) Lipid-associated macrophages control metabolic homeostasis in a Trem2-dependent manner. *Cell*, **178**, 686–698.
33. Liebler,J.M., Marconett,C.N., Juul,N., Wang,H., Liu,Y., Flodby,P., Laird-Offringa,I.A., Minoo,P. and Zhou,B. (2016) Combinations of differentiation markers distinguish subpopulations of alveolar epithelial cells in adult lung. *Am. J. Physiol. Lung Cell. Mol. Physiol.*, **310**, L114–L120.
34. Little,D.R., Gerner-Mauro,K.N., Flodby,P., Crandall,E.D., Borok,Z., Akiyama,H., Kimura,S., Ostrin,E.J. and Chen,J. (2019) Transcriptional control of lung alveolar type 1 cell development and maintenance by NK homeobox 2-1. *Proc. Natl. Acad. Sci. U.S.A.*, **116**, 20545–20555.
35. Franzen,O., Gan,L.M. and Bjorkegren,J.L.M. (2019) PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data. *Database (Oxford)*, **2019**, baz046.
36. Lee,E.J., Park,M.K., Kim,H.J., Kim,E.J., Kang,G.J., Byun,H.J. and Lee,C.H. (2016) Epithelial membrane protein 2 regulates sphingosylphosphorylcholine-induced keratin 8 phosphorylation and reorganization: changes of PP2A expression by interaction with alpha4 and caveolin-1 in lung cancer cells. *Biochim. Biophys. Acta*, **1863**, 1157–1169.
37. Bruggeman,L.A., Martinka,S. and Simske,J.S. (2007) Expression of TM4SF10, a Claudin/EMP/PMP22 family cell junction protein, during mouse kidney development and podocyte differentiation. *Dev. Dyn.*, **236**, 596–605.
38. Lopez-Anido,C., Poitelon,Y., Gopinath,C., Moran,J.J., Ma,K.H., Law,W.D., Antonellis,A., Feltri,M.L. and Svaren,J. (2016) Tead1 regulates the expression of Peripheral Myelin Protein 22 during Schwann cell development. *Hum. Mol. Genet.*, **25**, 3055–3069.
39. Weisenhaus,M., Allen,M.L., Yang,L., Lu,Y., Nichols,C.B., Su,T., Hell,J.W. and McKnight,G.S. (2010) Mutations in AKAP5 disrupt dendritic signaling complexes and lead to electrophysiological and behavioral phenotypes in mice. *PLoS One*, **5**, e10325.
40. Chang,D.R., Martinez Alanis,D., Miller,R.K., Ji,H., Akiyama,H., McCrea,P.D. and Chen,J. (2013) Lung epithelial branching program

antagonizes alveolar differentiation. *Proc. Natl. Acad. Sci. U.S.A.*, **110**, 18042–18051.
41. Wu,C.J., Mannan,P., Lu,M. and Udey,M.C. (2013) Epithelial cell adhesion molecule (EpCAM) regulates claudin dynamics and tight junctions. *J. Biol. Chem.*, **288**, 12253–12268.
42. Wang,Y., Frank,D.B., Morley,M.P., Zhou,S., Wang,X., Lu,M.M., Lazar,M.A. and Morrisey,E.E. (2016) HDAC3-Dependent Epigenetic Pathway Controls Lung Alveolar Epithelial Cell Remodeling and Spreading via miR-17-92 and TGF-beta Signaling Regulation. *Dev. Cell*, **36**, 303–315.
43. Wang,X., Wang,Y., Snitow,M.E., Stewart,K.M., Li,S., Lu,M. and Morrisey,E.E. (2016) Expression of histone deacetylase 3 instructs alveolar type I cell differentiation by regulating a Wnt signaling niche in the lung. *Dev. Biol.*, **414**, 161–169.
44. Kim,T.H., Li,F., Ferreiro-Neira,I., Ho,L.L., Luyten,A., Nalapareddy,K., Long,H., Verzi,M. and Shivdasani,R.A. (2014) Broadly permissive intestinal chromatin underlies lateral inhibition and cell plasticity. *Nature*, **506**, 511–515.
45. Ose,T., Kadowaki,Y., Fukuhara,H., Kazumori,H., Ishihara,S., Udagawa,J., Otani,H., Takasawa,S., Okamoto,H. and Kinoshita,Y. (2007) Reg I-knockout mice reveal its role in regulation of cell growth that is required in generation and maintenance of the villous structure of small intestine. *Oncogene*, **26**, 349–359.
46. Baraille,F., Ayari,S., Carriere,V., Osinski,C., Garbin,K., Blondeau,B., Guillemain,G., Serradas,P., Rousset,M., Lacasa,M. *et al.* (2015) Glucose Tolerance Is Improved in Mice Invalidated for the Nuclear Receptor HNF-4gamma: a Critical Role for Enteroendocrine Cell Lineage. *Diabetes*, **64**, 2744–2756.
47. Bluemel,S., Wang,L., Martino,C., Lee,S., Wang,Y., Williams,B., Horvath,A., Stadlbauer,V., Zengler,K. and Schnabl,B. (2018) The Role of Intestinal C-type Regenerating Islet Derived-3 Lectins for Nonalcoholic Steatohepatitis. *Hepatol Commun*, **2**, 393–406.
48. UniProt,C. (2023) UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Res.*, **51**, D523–D531.
49. Sala Frigerio,C., Wolfs,L., Fattorelli,N., Thrupp,N., Voytyuk,I., Schmidt,I., Mancuso,R., Chen,W.T., Woodbury,M.E., Srivastava,G. *et al.* (2019) The Major Risk Factors for Alzheimer's Disease: age, Sex, and Genes Modulate the Microglia Response to Abeta Plaques. *Cell Rep.*, **27**, 1293–1306.
50. Deczkowska,A., Keren-Shaul,H., Weiner,A., Colonna,M., Schwartz,M. and Amit,I. (2018) Disease-associated microglia: a universal immune sensor of neurodegeneration. *Cell*, **173**, 1073–1081.
51. Burgess,M., Wicks,K., Gardasevic,M. and Mace,K.A. (2019) Cx3CR1 expression identifies distinct macrophage populations that contribute differentially to inflammation and repair. *Immunohorizons*, **3**, 262–273.
52. Liang,T., Chen,J., Xu,G., Zhang,Z., Xue,J., Zeng,H., Jiang,J., Chen,T., Qin,Z., Li,H. *et al.* (2021) TYROBP, TLR4 and ITGAM regulated macrophages polarization and immune checkpoints expression in osteosarcoma. *Sci. Rep.*, **11**, 19315.
53. Dang,D., Taheri,S., Das,S., Ghosh,P., Prince,L.S. and Sahoo,D. (2020) Computational approach to identifying universal macrophage biomarkers. *Front Physiol*, **11**, 275.
54. Chua,R.L., Lukassen,S., Trump,S., Hennig,B.P., Wendisch,D., Pott,F., Debnath,O., Thurmann,L., Kurth,F., Volker,M.T. *et al.* (2020) COVID-19 severity correlates with airway epithelium-immune cell interactions identified by single-cell analysis. *Nat. Biotechnol.*, **38**, 970–979.
55. Sun,J., Zhuang,Z., Zheng,J., Li,K., Wong,R.L., Liu,D., Huang,J., He,J., Zhu,A., Zhao,J. *et al.* (2020) Generation of a broadly useful model for COVID-19 pathogenesis, vaccination, and treatment. *Cell*, **182**, 734–743.
56. Muhlethaler-Mottet,A., Otten,L.A., Steimle,V. and Mach,B. (1997) Expression of MHC class II molecules in different cellular and functional compartments is controlled by differential usage of multiple promoters of the transactivator CIITA. *EMBO J.*, **16**, 2851–2860.
57. Roche,P.A. and Furuta,K. (2015) The ins and outs of MHC class II-mediated antigen processing and presentation. *Nat. Rev. Immunol.*, **15**, 203–216.
58. Pfeffer,L.M. (2011) The role of nuclear factor kappaB in the interferon response. *J. Interferon Cytokine Res.*, **31**, 553–559.
59. Damazo,A.S., Sampaio,A.L., Nakata,C.M., Flower,R.J., Perretti,M. and Oliani,S.M. (2011) Endogenous annexin A1 counter-regulates bleomycin-induced lung fibrosis. *BMC Immunol.*, **12**, 59.
60. Seale,L.A., Torres,D.J., Berry,M.J. and Pitts,M.W. (2020) A role for selenium-dependent GPX1 in SARS-CoV-2 virulence. *Am. J. Clin. Nutr.*, **112**, 447–448.
61. Singh,P. and Ali,S.A. (2022) Multifunctional role of S100 protein family in the immune system: an update. *Cells*, **11**, 2274.
62. Yang,H., Wang,H., Czura,C.J. and Tracey,K.J. (2005) The cytokine activity of HMGB1. *J Leukoc Biol*, **78**, 1–8.
63. Zeisel,A., Hochgerner,H., Lonnerberg,P., Johnsson,A., Memic,F., van der Zwan,J., Haring,M., Braun,E., Borm,L.E., La Manno,G. *et al.* (2018) Molecular architecture of the mouse nervous system. *Cell*, **174**, 999–1014.
64. Nan,X., Campoy,F.J. and Bird,A. (1997) MeCP2 is a transcriptional repressor with abundant binding sites in genomic chromatin. *Cell*, **88**, 471–481.
65. Lyst,M.J. and Bird,A. (2015) Rett syndrome: a complex disorder with simple roots. *Nat. Rev. Genet.*, **16**, 261–275.
66. Fey,M. and Lenssen,J.E. (2019) Fast graph representation learning with PyTorch Geometric. arXiv doi: https://arxiv.org/abs/1903.02428, 06 March 2019, preprint: not peer reviewed.
67. Wang,C., Krafft,P., Mahadevan,S., Ma,Y. and Fu,Y. (2011) In: *Manifold Learning: Theory and Applications*. CRC Press Boca Raton, FL, USA, pp. 95–120.
68. Nguyen,N.D., Huang,J. and Wang,D. (2022) A deep manifold-regularized learning model for improving phenotype prediction from multi-modal data. *Nat Comput Sci*, **2**, 38–46.
69. Anders,S. and Huber,W. (2010) Differential expression analysis for sequence count data. *Genome Biol.*, **11**, R106.
70. Reid,J.E. and Wernisch,L. (2016) Pseudotime estimation: deconfounding single cell time series. *Bioinformatics*, **32**, 2973–2980.
71. Xu,Q., Li,G., Osorio,D., Zhong,Y., Yang,Y., Lin,Y.T., Zhang,X. and Cai,J.J. (2022) scInTime: a computational method leveraging single-cell trajectory and gene regulatory networks to identify master regulators of cellular differentiation. *Genes (Basel)*, **13**, 371.
72. Yang,Y., Li,G., Zhong,Y., Xu,Q., Lin,Y.T., Roman-Vicharra,C., Chapkin,R.S. and Cai,J.J. (2023) scTenifoldXct: a semi-supervised method for predicting cell-cell interactions and mapping cellular communication graphs. *Cell Syst.*, **14**, 302–311.