

# SCGNN: scRNA-seq Dropout Imputation via Induced Hierarchical Cell Similarity Graph

Anonymous Authors<sup>1</sup>

## Abstract

Single-cell RNA sequencing provides tremendous insights to understand biological systems. However, the noise from dropout can corrupt the downstream biological analysis. Hence, it is desirable to impute the dropouts accurately. In this work, we propose a simple and powerful dropout imputation method (SCGNN) by applying a bottlenecked Graph Convolutional Neural Network on an induced hierarchical cell similarity graph. We show SCGNN has competitive performance against state-of-the-art baselines across three datasets and can improve downstream analysis.

## 1. Introduction

Single-cell RNA sequencing (scRNA-seq) revolutionizes the ability to study the biological processes by narrowing the study resolution to single cell level (Usoskin et al., 2015; Keren-Shaul et al., 2017; Stephenson et al., 2018; Gladka et al., 2018).

Despite tremendous success, scRNA-seq suffers from noisiness due to the low RNA capture rate (Kharchenko et al., 2014; Jia et al., 2017). Lots of missing values (zero) are observed in scRNA experiment and these values could be either truly zero RNA expression or false negatives due to the low capture rate and failure of amplification. The false negatives is called dropout events. These dropout could potentially affect the downstream analysis such as cell clustering (Hicks et al., 2018). Hence, to leverage the scRNA-seq for biological tasks, it is of tremendous necessity and importance to impute these dropouts accurately.

Several scRNA-seq imputation methods have shown initial success (Arisdakessian et al., 2019; Van Dijk et al., 2018; Huang et al., 2018; Gong et al., 2018) and there are two principles that are leveraged. The first one is the gene co-expression pattern. Through gene regulations, several genes

can be expressed together. Hence, if a gene is an incorrectly dropout event, it could be corrected after observing high values of co-expressed genes in the same cell. The second one is cell-cell similarity. As scRNA experiments sequence large number of cells where genes in the similar cells are expected to have similar expression values, a dropout event in one cell can be recovered by imputing the values from a similar cell. These two principles govern the majority of the methods such as MAGIC (Van Dijk et al., 2018), SAVER (Huang et al., 2018), DCA (Eraslan et al., 2019) and etc. The auto-encoder approaches are reported to have competitive performances in several recent studies (Arisdakessian et al., 2019; Eraslan et al., 2019).

Despite promising results, there are still several challenges that could be tackled in previous works:

**1. Hierarchical cell similarity is missing.** For cell similarity, previous works usually select a set of similar cells and then treat all of them as the same. However, it is likely that there is a hierarchy among the similar cells, i.e. a set of cells have a higher similarity or importance to impute than others.

**2. Method that combine cell similarity and gene co-expression principles is lacking.** We observed that most of the previous works focus on improving either cell similarity or gene similarity principle.

To tackle the above challenges, in this work, we propose a new method called SCGNN, which successfully addresses the above two challenges via:

**1. An induced hierarchical cell similarity graph.** The induced cell similarity graph is constructed in a way that preserves the cell similarity hierarchy such that the immediate neighbor of a cell is the top K most similar cells, with the n-hop neighbors' similarity decreasing as n goes larger. This network is induced from the raw scRNA-seq data and does not require additional sources.

**2. A novel bottle-necked GCN network to integrate cell similarity and gene co-expression patterns.** SCGNN leverages the neighborhood message passing mechanism of GCN. To allow GCN to capture the gene co-expression pattern, we inject the bottleneck effect into the GCN

<sup>1</sup>Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review at the ICML 2020 Workshop on Computational Biology (WCB). Do not distribute.

network, which has shown efficacy to capture gene co-expressions (Arisdakessian et al., 2019). SCGNN achieve this through a simple modifications by manipulating the weight matrix of GCNs. A one line equation is derived to conduct the entire process from raw data to imputed data.

SCGNN has competitive performances in dropout imputation on three real world datasets when compared to several state-of-the-art baselines. The imputed scRNA matrix reduces noise and improves downstream biological analysis.

## 2. Method

**Task Description.** Given a dataset of scRNA sequences  $\mathbf{X}$ , where  $\mathbf{X}_{ij}$  is the expression value of  $j$ -th gene in  $i$ -th cell, we want to find a function  $\mathcal{F}$  to generate an imputed scRNA seq matrix  $\tilde{\mathbf{X}}$  where  $\tilde{\mathbf{X}}_{ij}$  is the imputed expression value of  $j$ -th gene in  $i$ -th cell:  $\mathcal{F} : \mathbf{X} \rightarrow \tilde{\mathbf{X}}$ .

Since the dropout is unknown, we follow standard practice (Arisdakessian et al., 2019) to mask portions of non-zero values and make them zeros to simulate the dropout. This step and other data processing step is described in detail in Section. 3.1. We denote the preprocessed scRNA-seq  $\hat{\mathbf{x}}$ .

**Induced hierarchical cell similarity graph.** To aggregate gene information from similar cells, we construct a cell graph where the nodes are cells and edges connect similar cell. For a high quality graph, we filter the noise such as experiment artifacts by applying Principle Component Analysis (PCA) and using the top 50 dimensions to retain the majority of the variations. Then, we compute the euclidean distance among each cell-pair  $i$  and  $j$ , which accounts for their similarity. The similarity matrix  $\mathbf{S}$  is constructed where each position  $\mathbf{S}_{ij}$  is the similarity between cells  $i$  and  $j$ .  $\mathbf{S}$  is then used to construct a  $K$ -nearest neighbor (KNN) graph where each cell is connected to its  $K$  most-similar cells based on  $\mathbf{S}$ . This KNN graph is our cell similarity graph  $G$  with adjacency matrix  $\mathbf{A}$ .

Note that  $G$  preserves the cell similarity hierarchies. For target cell  $i$  that we want to impute, the immediate neighbors are the top  $K$  similar cells. Since we know for each neighbor  $j$  of cell  $i$ ,  $j$  has  $K$  neighbors ( $i$ 's second-hop neighbors) that are the most similar to  $j$ . By the transitivity of euclidean distance, the second-hop neighbors are also similar to our target cell, although less similar to the first-hop neighbors. Hence,  $G$  preserves the hierarchical structure of cell similarity. We call cells in the first-hop neighbors *first-order* similar cells and second-hop neighbors *second-order* similar cells and so forth.

**Graph convolutional encoding.** We use a two-layer graph convolutional neural network (Kipf & Welling, 2016) to aggregate the hierarchical cell similarity information in a simple propagation equation. GNN could be considered as

message passing across connected nodes where messages are node attributes (scRNA sequence in our case). For the first layer of GCN, for any node cell  $i$  with the attribute  $\hat{\mathbf{x}}_i$ , it aggregates node attributes from the connected neighbors  $j \in \mathcal{N}(i)$  with a learnable matrix  $\mathbf{W}_g^1$  and a non-linear function  $\sigma$  to obtain an updated cell representation  $\mathbf{H}_i$  for target cell  $i$ :

$$\mathbf{H}_i = \sigma \left( \frac{1}{\hat{K}} \sum_{j \in \mathcal{N}(i) \cup i} \hat{\mathbf{x}}_j \mathbf{W}_g^1 \right),$$

where  $\hat{K}$  is the KNN parameter  $K$  plus 1 for the node itself. It is also the degree of each cell in our similarity network. To interpret this, we see it takes the mean of the weighted first-order similar cells' scRNA-seq data. Intuitively, for dropout gene, this step imputes these gene expressions by taking the average of gene expressions from the most similar cells. For the second layer, the GCN applies the same propagation rule on top of the new  $\mathbf{H}_i$  to obtain the imputed  $\tilde{\mathbf{X}}_i$ :

$$\tilde{\mathbf{X}}_i = \sigma \left( \frac{1}{\hat{K}} \sum_{j \in \mathcal{N}(i) \cup i} \mathbf{H}_j \mathbf{W}_g^2 \right).$$

The second propagation layer utilizes the second-order similar cells. It is clear when we combine the two propagation layers by expanding  $\mathbf{H}_j$ :

$$\tilde{\mathbf{X}}_i = \sigma \left( \frac{1}{\hat{K}} \sum_{j \in \mathcal{N}(i) \cup i} \left( \sigma \left( \frac{1}{\hat{K}} \sum_{k \in \mathcal{N}(j) \cup j} \hat{\mathbf{x}}_k \mathbf{W}_g^1 \right) \right) \mathbf{W}_g^2 \right).$$

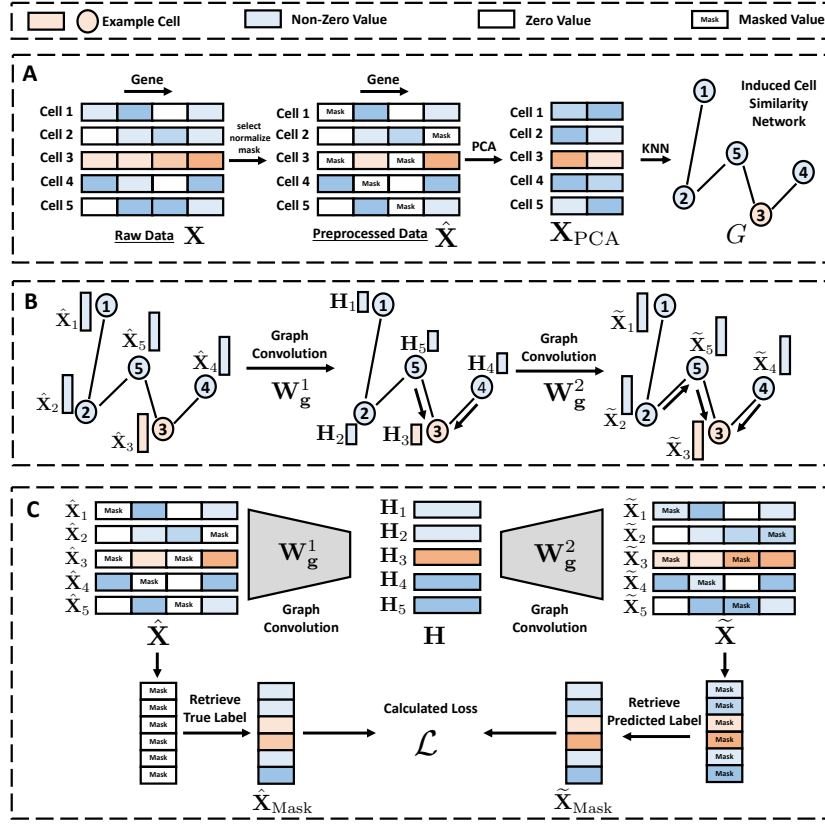
After the second layer, we see that GCN actually imputes the target gene expressions by aggregation first-order cells' expression by the factor  $\frac{1}{\hat{K}}$  and the second-order cells' expression by the factor  $\frac{1}{\hat{K}^2}$ . Intuitively speaking, it aggregates similar cells where different level of similarity will have different aggregation weights. Less similar cells have less weights for the final imputation vector. This is an ideal property for dropout imputation that previous methods do not have.

The hierarchical imputation equation can be vectorized for every cell in  $G$  through matrix multiplication for fast parallel computation (Kipf & Welling, 2016):

$$\tilde{\mathbf{X}} = \sigma \left( \hat{\mathbf{A}} \sigma \left( \hat{\mathbf{A}} \hat{\mathbf{X}} \mathbf{W}_g^1 \right) \mathbf{W}_g^2 \right)$$

where  $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}}$ ,  $\tilde{\mathbf{D}}$  and  $\tilde{\mathbf{A}}$  are the renormalized degree and adjacency matrices of cell similarity graph  $G$ , defined as:  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$  and  $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$  ( $\mathbf{I}$  is the identity matrix). We see this equation takes in the preprocessed data and outputs the imputed values.

**Bottle-necked GCN weights.** We present a novel but simple modifications on GCN to leverage gene co-expression



**Figure 1. Model Illustration.** **A.** The raw scRNA-seq data is first preprocessed and then we construct the cell-similarity graph  $G$  by applying KNN on the PCA embeddings. **B.** A Graph Convolutional Network is applied on top of  $G$ . In the first layer, imputation information from the most similar cells (4, 5) are propagated to the center cell (3). In the second layer, the second-order similar cell (2) imputation information is also propagated to the center cell (3) with a discount factor. This models the hierarchical cell similarity imputation principle. Simultaneous to the propagation, in the first layer of GCN, the input scRNA seq data is mapped to low-dimensional latent representation through  $W_g^1$  (e.g.  $\hat{X}_3 \rightarrow H_3$ ). Then, in the second layer of GCN, it is mapped back to the original scRNA dimension through  $W_g^2$  (e.g.  $H_3 \rightarrow \tilde{X}_3$ ). This process enables SCGNN to recognize gene co-expression patterns and recover it. **C.** After imputation, we retrieve the imputed masked values  $\tilde{X}_{Mask}$  and the input masked values  $\hat{X}_{Mask}$ . The MSE loss  $\mathcal{L}$  is then computed.

patterns. While the standard graph convolutional network enables aggregation over similar cells, to leverage gene co-expression pattern is not obvious. Previous successful methods (Arisdakessian et al., 2019; Talwar et al., 2018; Van Dijk et al., 2018) use auto-encoder to first map the scRNA-seq into a low dimensional latent space and then recover it to the original space. We borrow this idea of bottleneck into the GCN framework. A brute-force adaptation would be to include a separate auto-encoder and concatenate the latent embedding from both auto-encoder and GCNs. However, this requires additional computation memory and speed. Instead, we directly inject this bottleneck into the GCN framework by setting the weight matrix  $W_g^1$  to have dimension  $\mathbb{R}^{M \times D}$  and weight matrix  $W_g^2$  to have dimension  $\mathbb{R}^{D \times M}$ , where  $D \ll M$  ( $M$  is the number of genes in the input). This way, we inject an auto-encoder into the GCN without any additional computation time and it has the same effect of auto-encoder since the weight matrix is

applied on every cells.

**Loss and inference.** Given the input  $\hat{X}$  and the output  $\tilde{X}$ , we first retrieve the corresponding masked values  $\hat{X}_{Mask}$  and  $\tilde{X}_{Mask}$ . Then, we compute the reconstruction loss:  $\mathcal{L} = \|\hat{X}_{Mask} - \tilde{X}_{Mask}\|_2$ . This loss measures the accuracy of dropout imputation on the masked true gene expressions. During inference, we fill in the masking values and freeze the model weights. Then, we put the predicted values for the dropout position while leave the non-zero values intact. The imputed scRNA-seq is then obtained and ready for downstream biomedical analysis.

### 3. Experiment

#### 3.1. Setup

**Datasets and Data Processing.** We use three datasets. The first one is Zeisel dataset (Zeisel et al., 2015), which is scRNA-seq from mouse cortex and hippocampus and has

Table 1. Dataset Statistics.

Dataset	# Cells	#Genes	% Mask
Zeisel	3,005	19,972	18.79
Neuron1K	1,301	31,053	10.45
Jurkat	3,258	32,738	9.76

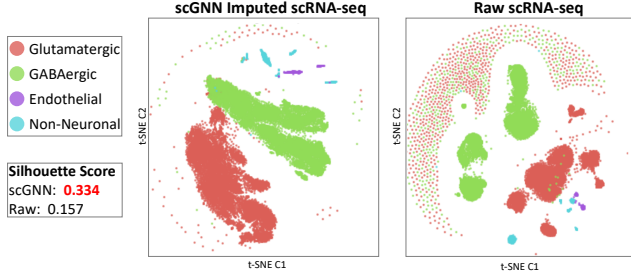


Figure 2. SCGNN improves downstream biomedical analysis. It obtains better clustering result than the raw scRNA-seq data, measured by more than 100% increase on clustering silhouette score.

3,005 cells and 19,972 genes. The second is Neuron1K. It derives from the brain cells from E18 mice (10X Genomics). It has 1,301 cells and 31,053 genes. The third one is Jurkat (10X Genomics), which is extracted from human blood cell lines and have 3,258 cells and 32,738 genes.

There are three steps for the standard data processing (Talwar et al., 2018; Arisdakessian et al., 2019). First we select important genes as many genes are not useful. We filter genes that have mean over variance ratio (MVR) less than 0.5. Second, we conduct median normalization of transcript abundance by forcing each cell has the same transcript counts as the median across cells. This way we eliminate cell size as the potential inaccurate imputation. In the last, we calculate the dropout percentage in the original scRNA-seq dataset and randomly mask the same percentage on the processed dataset.

**Evaluation Metrics.** We use two metrics for evaluating dropout imputation accuracy. The first one is Mean Squared Error (MSE), which equals to the loss function. The second one is the Pearson Correlation ( $R^2$ ), which measures the correlation between the predicted and true transcript counts.

**Baselines.** To compare SCGNN’s performance, we evaluate dropout imputation performance for 1. **MAGIC** (Van Dijk et al., 2018) constructs Markov affinity matrix to model cell similarity and 2. **DCA** (Eraslan et al., 2019) uses denoising auto-encoders, taking account into the count distribution, overdispersion and sparsity of the data by constructing a negative binomial loss.

Table 2. Five fold average and standard deviation of MSE and R2.

Dataset	Zeisel		Neuron1K		Jurkat	
Metric	MSE	R2	MSE	R2	MSE	R2
DCA	165.7	0.863	411.9	0.931	40.3	0.960
MAGIC	348.6	0.536	441.8	0.825	22.1	0.962
scGNN	98.6	0.871	113.0	0.943	32.7	0.933

### 3.2. Preliminary Result

**SCGNN is a robust, competitive dropout imputation algorithm.** The prediction result measured by R-squared and MSE is reported in Figure. (2). We see that in Zeisel and Neuron1K, SCGNN has the highest R-squared and lowest MSE. In Jurkat, SCGNN’s performance is worse than DCA and MAGIC in R-Squared and better than DCA in MSE. This suggests SCGNN has very good predictive performances, and in often times, better predictions. We see both DCA and MAGIC has higher variance in all the three datasets, especially MAGIC, it could not infer high-quality dropouts in the Zeisel dataset, suggested by the mediocre R-squared result. In the Neuron1K dataset, we found DCA and MAGIC perform significantly bad in one random data split fold while SCGNN obtains consistently good result, suggesting SCGNN is robust.

**SCGNN improves downstream function clustering analysis.** The purpose of doing dropout imputation is to enable more accurate downstream scRNA-seq analysis. In this experiment, we compare the cell types t-SNE visualization, before and after SCGNN imputation. We report the result in Figure. (2). We see that before SCGNN imputation, the Glutamatergic and GABAergic, two clusters, are separated into several small parts. SCGNN is able to identify the main variance and cluster them together. To quantify the improvement, we use the silhouette score, which measures the how similar a cell is to its own cluster compared to other clusters. We compute the scores, raw data achieves 0.157, and SCGNN achieves 0.334, a nearly 100% relative increase over raw data. This suggests the benefit of using SCGNN and its usefulness in improving downstream biomedical analysis.

## 4. Conclusion

In this work, we present a novel, simple, and efficient computational method that unifies cell similarity and gene co-expression patterns and also leverages the hierarchical nature of cell similarity. SCGNN achieves great dropout imputation predictive performance and improves downstream analysis.

## References

- Arisdakessian, C., Poirion, O., Yunits, B., Zhu, X., and Garmire, L. X. Deepimpute: an accurate, fast, and scalable deep neural network method to impute single-cell rna-seq data. *Genome biology*, 20(1):1–14, 2019.
- Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S., and Theis, F. J. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):1–14, 2019.
- Gladka, M. M., Molenaar, B., De Ruiter, H., Van Der Elst, S., Tsui, H., Versteeg, D., Lacraz, G. P., Huibers, M. M., Van Oudenaarden, A., and Van Rooij, E. Single-cell sequencing of the healthy and diseased heart reveals cytoskeleton-associated protein 4 as a new modulator of fibroblasts activation. *Circulation*, 138(2):166–180, 2018.
- Gong, W., Kwak, I.-Y., Pota, P., Koyano-Nakagawa, N., and Garry, D. J. Drimpute: imputing dropout events in single cell rna sequencing data. *BMC bioinformatics*, 19(1):220, 2018.
- Hicks, S. C., Townes, F. W., Teng, M., and Irizarry, R. A. Missing data and technical variability in single-cell rna-sequencing experiments. *Biostatistics*, 19(4):562–578, 2018.
- Huang, M., Wang, J., Torre, E., Dueck, H., Shaffer, S., Bonasio, R., Murray, J. I., Raj, A., Li, M., and Zhang, N. R. Saver: gene expression recovery for single-cell rna sequencing. *Nature methods*, 15(7):539–542, 2018.
- Jia, C., Hu, Y., Kelly, D., Kim, J., Li, M., and Zhang, N. R. Accounting for technical noise in differential expression analysis of single-cell rna sequencing data. *Nucleic acids research*, 45(19):10978–10988, 2017.
- Keren-Shaul, H., Spinrad, A., Weiner, A., Matcovitch-Natan, O., Dvir-Szternfeld, R., Ulland, T. K., David, E., Baruch, K., Lara-Astaiso, D., Toth, B., et al. A unique microglia type associated with restricting development of alzheimer’s disease. *Cell*, 169(7):1276–1290, 2017.
- Kharchenko, P. V., Silberstein, L., and Scadden, D. T. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7):740, 2014.
- Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
- Stephenson, W., Donlin, L. T., Butler, A., Rozo, C., Bracken, B., Rashidfarrokhi, A., Goodman, S. M., Ivashkiv, L. B., Bykerk, V. P., Orange, D. E., et al. Single-cell rna-seq of rheumatoid arthritis synovial tissue using low-cost microfluidic instrumentation. *Nature communications*, 9(1):1–10, 2018.
- Talwar, D., Mongia, A., Sengupta, D., and Majumdar, A. Autoimpute: Autoencoder based imputation of single-cell rna-seq data. *Scientific reports*, 8(1):1–11, 2018.
- Usoskin, D., Furlan, A., Islam, S., Abdo, H., Lönnerberg, P., Lou, D., Hjerling-Leffler, J., Haeggström, J., Kharchenko, O., Kharchenko, P. V., et al. Unbiased classification of sensory neuron types by large-scale single-cell rna sequencing. *Nature neuroscience*, 18(1):145, 2015.
- Van Dijk, D., Sharma, R., Nainys, J., Yim, K., Kathail, P., Carr, A. J., Burdziak, C., Moon, K. R., Chaffer, C. L., Pattabiraman, D., et al. Recovering gene interactions from single-cell data using data diffusion. *Cell*, 174(3):716–729, 2018.
- Zeisel, A., Muñoz-Manchado, A. B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., et al. Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq. *Science*, 347(6226):1138–1142, 2015.