ALZHEIMER'S SCRNA-SEQ DATA ANALYSIS USING MULTI-TYPE DEEP AUTOENCODERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell RNA sequencing (scRNA-seq) technology has been applied in Alzheimer's disease (AD) research to explore its pathogenic mechanism. The complexity of analyzing and utilizing sequencing data is significantly amplified by the high dimensionality and high noise levels of the data, as well as the presence of missing data. In order to tackle these problems, we suggest implementing a novel data processing framework that consists of two primary algorithms: the imputation algorithm scICLGAE and the clustering algorithm scCapsZB. scICLGAE employs two graph autoencoders (GAE) to fill in missing values by utilizing comparison learning to filter similar nodes from both global information and local structure. In order to verify the imputation impact and enhance the precision of the clustering outcomes, we have devised the scCapsZB algorithm. scCapsZB is a method that integrates a capsule network and a zero-inflated negative binomial distribution (ZINB) autoencoder. It incorporates prior knowledge through the capsule network and employs a self-attention routing mechanism to reduce the number of training parameters. Additionally, it uses the ZINB model to capture the feature representation of the data. The testing of our new framework on both generic and Alzheimer's datasets demonstrates substantial enhancements.

1 INTRODUCTION

Single-cell RNA sequencing (scRNA-seq) technology enables the understanding of the potential mechanisms of diseases through dynamic gene expression analysis Zhang et al. (2023) and has emerged as a transformative instrument for research in the fields of biology and medicine Luo et al. (2025). It significantly enhances the identification of a range of diseases, including cancer Huang et al. (2023), immune system abnormalities Yin et al. (2024), mental illnesses Zhou et al. (2023), and chronic diseases Nakayama et al. (2024). Recently, researchers have applied scRNA-seq to Alzheimer's disease (AD) prevention and treatment Cisterna-García et al. (2023). By the year 2050, it is projected that the number of individuals diagnosed with Alzheimer's disease will surpass 150 million, thereby exerting a considerable impact on both familial structures and public health systems Huang et al. (2024). The scRNA-seq technology facilitates the investigation of neural development associated with Alzheimer's disease and enables the examination of cellular alterations occurring prior to and following the onset of the disease Feng et al. (2024). Nonetheless, the high dimensionality, sparsity, and inherent noise associated with sequencing data pose significant challenges to data analysis within the context of Alzheimer's disease research Kharchenko (2021).

Recent advancements in imputation algorithms have been made to enhance the quality of scRNA-seq data through the integration of autoencoders (AE) with deep neural networks (DNN), such as DCA Eraslan et al. (2019), SAVER-X Wang et al. (2019), GraceImpute Wang et al. (2025), and scGMAI Yu et al. (2021). For the limitations of AE, these improved algorithms have proven effective. DeepImpute Arisdakessian et al. (2019) employs a divide-and-conquer strategy within a DNN model framework to forecast gene expression levels. scIGANs Xu et al. (2020), which are founded on the principles of generative adversarial network (GAN), address the issue of over-smoothing by producing synthetic cellular data. This approach effectively enhances performance consistency across various cell populations. Clustering algorithms such as Dhaka Rashid et al. (2021), scvis Ding et al. (2018), and scVAE Grønbech et al. (2020) primarily combine DNN with variational autoencoder (VAE). Another effective method, scDeepCluster Tian et al. (2019), simultaneously learns feature representations while explicitly modeling cell clusters. Additionally, models such as scVI,

LDVAE Svensson et al. (2020), SAUCIE Amodio et al. (2019), and scScope Deng et al. (2019) integrate AE and VAE with multiple analytical functions. However, these unsupervised methodologies frequently yield outcomes that are devoid of biological relevance and tend to place disproportionate emphasis on gene attributes, thereby overlooking the interrelationships among cellular genes.

In this paper, we introduce a new data processing framework that includes imputation algorithm scICLGAE and clustering algorithm scCapsZB to tackle these issues. The imputation uses two graph autoencoders (GAEs) for contrastive learning, which combines local and global information to find similar nodes.For clustering, we use zero-inflated negative binomial distribution (ZINB) autoencoder and capsule network for feature extraction and denoising to capture gene relationships, coordinated by supervisory module. Our model outperformed others across 12 scRNA-seq datasets. When applied to AD data, improved clustering revealed changes in disease-related cell proportions, providing valuable insights for prevention and treatment.

2 METHODS

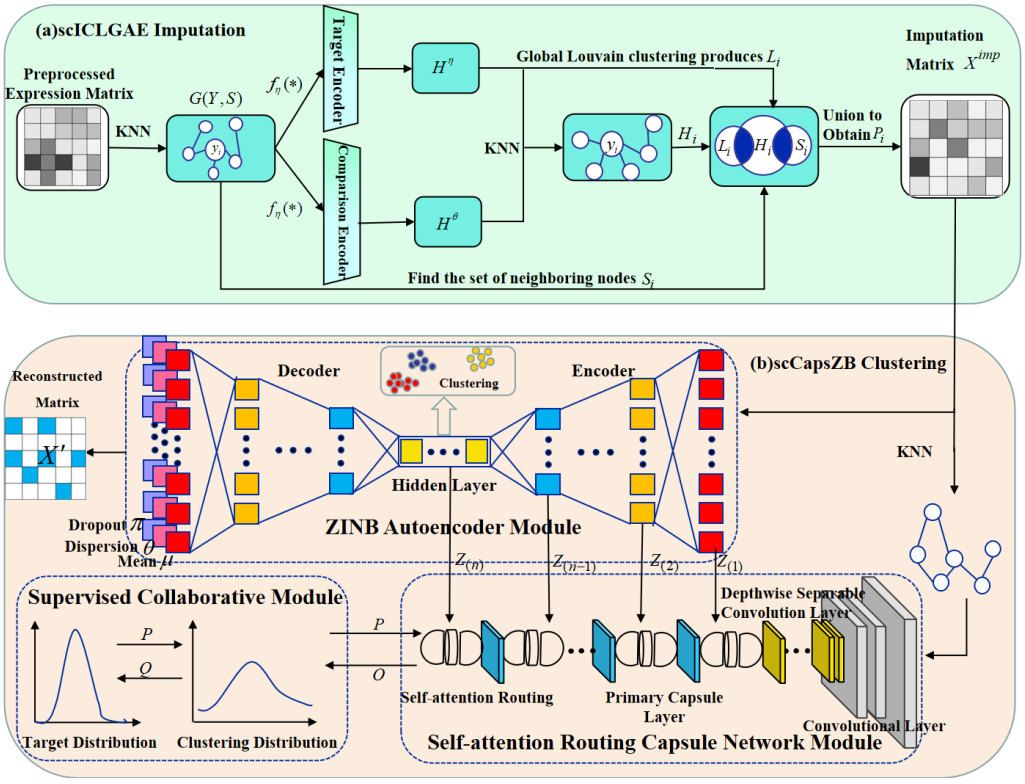


Figure 1: Model framework: The preprocessed expression matrix is transformed into a KNN graph via cosine similarity and fed into a contrastive encoder for new node representations. A recalculated KNN graph helps identify nodes similar to the target for data imputation. The imputed matrix is reanalyzed to obtain a new KNN graph, which serves as input for ZINB autoencoder and self-attention routing capsule network in cell clustering, with the overall process optimized by a supervised collaborative module.

2.0.1 CONSTRUCTING CELL MAPS.

We use the KNN algorithm to construct a cell graph, where nodes represent cells and edges represent connections between them. In this graph construction, the k value of the KNN algorithm is utilized to control the scale of adjacent nodes and measure the proximity of distances. Specifically, nodes within the k shortest distances from a given node are regarded as its neighbors and connected. To incorporate more potentially related cells in the analysis, a relatively larger k value is set. The

weights of these edges are calculated using cosine similarity, as shown in Equation (2):

$$S_{a,b} = \frac{\sum_{a=1,b=1}^{n} Z_a \cdot Z_b}{\sqrt{\sum_{a=1}^{n} Z_a^2}\sqrt{\sum_{b=1}^{n} Z_b^2}} \tag{1}$$

where Z_a and Z_b represent cells a and b, respectively. $S_{a,b}$ is the cosine similarity between these two nodes and its constituent matrix S, which is also the adjacency matrix that records the association information, while considering Y as its cell node matrix, the constructed KNN graph can be denoted as $G(Y, S)$.

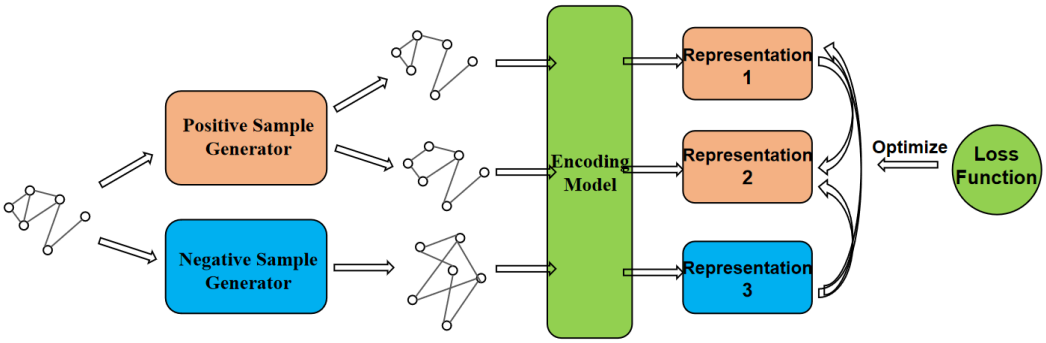


Figure 2: Graph contrastive learning framework: It consists of a positive sample generator, a negative sample generator, a encoding model, and a loss function. The positive and negative sample generators input the same graph. The encoding model learns the samples output by the generators and gets the corresponding feature representations. Finally, the parameters are trained by maximizing the difference of the positive and negative feature representations.

2.0.2 Contrast Learning Nodal Imputation.

The graph G is provided as input to the target encoder $f_\eta(*)$ and control encoder $f_\theta(*)$. Fig. 2 shows the graph contrastive learning framework Cen et al. (2023). These two autoencoders employ graph convolutional network (GCN) internally and generate corresponding node representations for each cell:

$$H^\eta = f_\eta(Y, S) \tag{2}$$

$$H^\theta = f_\theta(Y, S) \tag{3}$$

where H^η and H^θ is a summary of the individual cell representation. In order to get the whole set of sample nodes H_i that are similar to the target cell node y_i, it is necessary to recalculate the cosine similarity distance for both positive and negative samples:

$$sim\,(y_i, y_j) = \frac{h_i^\theta \cdot h_j^\eta}{\parallel h_i^\theta \parallel \parallel h_j^\eta \parallel}, \forall y_i \in Y \tag{4}$$

where h_i^θ represents the embedded output of the control encoder for cell y_i, and h_j^η represents the embedded output of the target encoder for the other cell y_j. By applying KNN with a smaller k value than that in cell graph construction (using a smaller value is to screen similar cells), we obtain H_i.

To get the set of locally most similar samples, we first define S_i as the collection of nodes that have a direct connection with the target cell y_i in the edge weight matrix S. Then, the set of locally most similar samples is achieved by computing the intersection of $H_i \cap S_i$.

To globally find other nodes similar to the target encoder embedding h_i^η, we apply the Louvain De Meo et al. (2011) algorithm to get the set L_i. The globally most similar set is then obtained by intersecting H_i with this globally-derived L_i. Given that L_i is based on positive samples, this intersection serves to enhance the positive sample representation. Finally take the union set:

$$P_i = (H_i \cap S_i) \cup (H_i \cap L_i) \tag{5}$$

For each cell i, if the expression of gene is not equal to zero, retain its value. If not, the imputed value should be the average expression of gene j from all cells p_k in the same node set P_i. The calculating process for the gene deletion value in cell i is illustrated by Equation (7), resulting in the estimated gene expression matrix X^{imp}.

$$\begin{cases} X_{i,j} & if X_{i,j} > 0 \\ \frac{\sum_{p_k \in P_i} p_{i,j}}{Count(P_i)} & if\ else \end{cases} \tag{6}$$

The contrastive loss function minimizes the cosine similarity distance of node representations from the target and contrastive autoencoders, enhancing their similarity during training and stabilizing the resultant similar node set. The loss function is shown in Equation (8).

$$L_{con} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{p_k \in P_i} \frac{f_\theta(y_i, S) f_\eta(p_k, S)^T}{\|f_\theta(y_i, S)\| \|f_\eta(p_k, S)\|} \tag{7}$$

2.1 Cluster Analysis

We enhance scCapsZB by using sub-labeled data for semi-supervised learning. The GNN is replaced with a capsule network connected to the ZINB autoencoder for cooptimization. Supervised module trains the entire network uniformly for clustering. See Fig.1(b) for the structure.

2.1.1 ZINB Autoencoder Modules.

The processed gene expression matrix is fed into the ZINB autoencoder, which consists of three parts: the encoder, the hidden layer, and the decoder. Gene features are embedded into a low-dimensional space using the similarity between the ZINB distribution and scRNA-seq data. The distribution functions of Negative Binomial (NB) and ZINB are shown in the followings.

$$NB\left(X; \mu, \theta\right) = \frac{\Gamma\left(X + \theta\right)}{\Gamma\left(\theta\right)} \left(\frac{\theta}{\theta + \mu}\right)^{\theta} \left(\frac{\mu}{\theta + \mu}\right)^{X} \tag{8}$$

$$ZINB\left(X; \pi, \mu, \theta\right) = \pi \delta_0\left(X\right) + \left(1 - \pi\right) NB\left(X; \mu, \theta\right) \tag{9}$$

where X is preprocessed matrix. μ, θ and π represent mean, variance, and dropout rate respectively, which are estimated by connecting three independent fully connected layers to the last layer of the decoder.

The encoder converts the preprocessing matrix X into the feature representation Z of the intermediate hidden layer using the following equation:

$$Z = f_{enc}\left(WX + b\right) \tag{10}$$

where W is the encoder weight vector; b is the encoder offset; f_{enc} represents the encoder function of the abstract representation. If the encoder has k layers, the learning process is as follows:

$$Z_{(k)} = \phi\Big(w_{(k)} Z_{(k-1)} + b_{(k)}\Big) \tag{11}$$

where $Z_{(k)}$ is the representation of the features learned in the k-th layer, $w_{(k)}$ is the weight of the layer, and $b_{(k)}$ is the offset term. The decoder transforms the feature representation Z from the intermediate hidden layer into the output matrix X' using the following process:

$$X' = f_{dec}\left(W'Z + b'\right) \tag{12}$$

where W' is the decoder weight vector; b' is the decoder offset; f_{dec} represents the decoder function of the abstract representation.

The loss function is defined as the sum of the negative logarithms of the ZINB distribution:

$$L_{zinb} = \sum -\log\left(ZINB\left(X|\pi, \mu, \theta\right)\right) \tag{13}$$

The decoder's final layer output is given by the following equation:

$$D = f_{dec}\left(f_{enc}\left(X\right)\right) \tag{14}$$

2.1.2 SELF-ATTENTION ROUTING CAPSULE NETWORK MODULE.

The module has four components: convolutional, depthwise separable convolutional, primary capsule and self-attention routing. For visualizing cell topology, a KNN graph is built on the processed gene expression matrix using Pearson correlation. The first two parts, like a regular convolutional network, use ordinary convolution and batch normalization for feature extraction and high-dim projection. Unlike traditional capsule nets Duarte et al. (2021); Chen et al. (2025); **?**, a depthwise separable convolution Balmez et al. (2025) is added in the second part to reduce capsule creation params. The last two parts are interleaved.

At the primary capsule layer, to make vector length denote entity probability and enable high-level capsule to predict its parameters from low-level output, a new activation function is used:

$$O_n^i = \text{squash}\left(g_n^i\right) = \left(1 - \frac{1}{e^{\|g_n^i\|}}\right) \frac{g_n^i}{\|g_n^i\|} \tag{15}$$

where g_n^i represent the input of the n-th capsule in the i-th layer, and O_n^i is the output vector with the same dimension and characteristics as g_n^i.

In this module, self-attention routing Mazzia et al. (2021) replaces traditional dynamic routing. It enables the output vectors of active capsules to reach corresponding high-level capsules. The input g_n^{i+1} of a high-level capsule is the weighted sum of the prediction vectors from the lower-level capsule O_n^i. When obtaining g_n^{i+1}, we incorporate the prior probability matrix L^i into the coupling coefficients C^i derived from the self-attention tensors E^i to get the self-attention routing weights. We integrate the knowledge from the ZINB autoencoder with the capsule network's outputs to optimize both modules:

$$O_n^i = \sigma O_n^i + (1 - \sigma)\, Z_{(i)} \tag{16}$$

Capsule network loss has two parts: sorting loss and KNN graph reconstruction loss, the former being the sum of losses by all digital capsule layers:

$$L_{clu} = T_{n^L} max(0, m^+ - \left\|O_n^L\right\|)^2 + \lambda\,(1 - T_{n^L})\, max(0, \left\|O_n^L\right\| - m^-)^2 \tag{17}$$

where O_n^L is the final layer's capsules. m^+, m^- and λ are hyperparameters. If the entity corresponding to the capsule exists, T_{n^L} is 1; otherwise, it's 0.

The reconstruction loss is defined by the Euclidean distance between the reconstructed and input graphs:

$$L_{recon} = \text{dist}\left(G, \hat{G}\right) = \sqrt{\sum_{i=1}^{n} \left(G_i - \hat{G}_i\right)^2} \tag{18}$$

where G is the input graph matrix, and $\hat{G}$ is the reconstructed graph matrix.

For semi-supervised network training and prior knowledge integration in capsule network clustering, this module uses a few real cell type labels instead of golden ones. Divide the dataset into sub-data and sub-label data with a certain ratio, ensuring much more sub-data. Mark real cell type numbers on sub-label data cells for accurate clustering while leaving sub-data cells unmarked. In each training round, input the marked sub-label data to initialize with its real cell types and calculate its loss.

$$L_{\text{sub-label}} = L_{\text{clu}}^{\text{sub-label}} + L_{recon}^{\text{sub-label}} \tag{19}$$

Then input the unlabeled sub-data. As it lacks labels, only the reconstruction graph loss can be computed. But with the prior sub-label data and its labels, the sub-data can be classified and its classification loss adopts the sub-label data loss. The total loss of the capsule network is shown as follows.

$$L_{\text{cap}} = L_{\text{clu}}^{\text{sub-label}} + L_{\text{sub-label}} \tag{20}$$

2.1.3 SUPERVISED COLLABORATIVE MODULE.

This module trains the entire network through the target distribution P, the clustering distribution Q, and the probability distribution O. Q use the Student's distribution to model the probability of

all cells being assigned to a K-means clustering center. The q_{it} element quantifies the similarity between the data representation z_i of the ZINB encoder layer i and the vector representation μ_t of the clustering center t:

$$q_{it} = \frac{\left(1 + \left\|\mu_t - z_i\right\|^2 / f\right)^{-\frac{f+1}{2}}}{\sum_{\mathrm{i}}(1 + \left\|\mu_t - z_i\right\|^2 / f)^{-\frac{f+1}{2}}} \tag{21}$$

where f is the degree of freedom. The component p_{it} of P is a more reliable representation of the data generated using q_{it}, which is the actual clustering center of the data. This computation is performed according to the Equation (23).

$$p_{it} = \frac{q_{it}^2 / \sum_t q_{it}}{\sum_{t'}(q_{it'}^2 / \sum_{t'} q_{it'})} \tag{22}$$

Clustering minimizes the log cross-entropy of target distribution P and clustering distribution Q to match Q's centers with the data's true centers, maximizing cluster compactness and separation. The loss is shown in Equation (24):

$$L_{cluster} = -p_{\mathrm{it}} \log\left(q_{\mathrm{it}}\right) - \left(1 - p_{\mathrm{it}}\right) \log\left(1 - q_{\mathrm{it}}\right) \tag{23}$$

P is computed from Q, meaning Q guides P's learning. From the capsule network, we get o_{it} which holds prior knowledge and cell details, representing the probability of cell i in cluster t. Using KL divergence Cui et al. (2023), we calculate the loss for P to supervise O, with the loss function in Equation (25).

$$L_k = \mathrm{KL}\left(\mathrm{P} \| \mathrm{O}\right) = \sum_i \sum_t p_{it} \log \frac{p_{it}}{o_{it}} \tag{24}$$

Dataset	Sequencing Platform	Cell Number	Gene Number	cell categories
10X_PBMCZheng et al. (2017)	10X	4271	16499	8
RomanovRomanov et al. (2017)	Smart-seq2	2881	24341	7
Human1Baron et al. (2016)	inDrop	1937	20125	14
Human2Baron et al. (2016)	inDrop	1724	20125	14
Human3Baron et al. (2016)	inDrop	3605	20125	14
Human4Baron et al. (2016)	inDrop	1303	20125	14
Mouse1Baron et al. (2016)	inDrop	822	14878	13
Mouse2Baron et al. (2016)	inDrop	1064	14878	13
CITE_CMBCMimitou et al. (2019)	10X	8617	2000	15
ZeiselZeisel et al. (2015)	Drop-seq	3005	19972	9
KleinKlein et al. (2015)	inDrop	2717	24175	4
Human_kidneyYoung et al. (2018)	10X	5685	33658	11

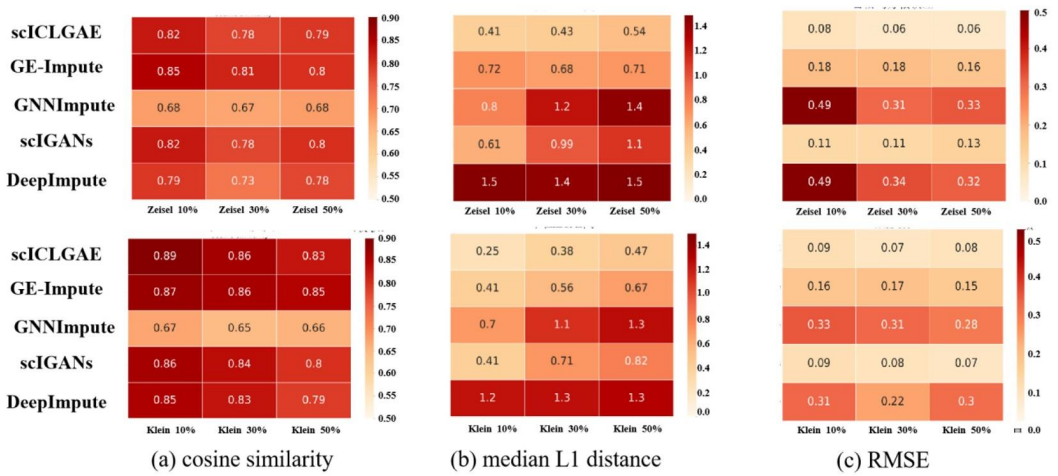
Table 1: Statistics of processed datasets.

3 Experiments

3.1 Experimental Settings

3.1.1 Datasets.

We evaluated our model using 12 scRNA-seq datasets from sequencing platforms, as shown in Table 1. Each dataset contains between 1000 and 9000 cells, all genetically annotated with known cell types. In the imputation section, we added a supplementary dataset, GSE138852 Grubman et al. (2019) which includes 6 Alzheimer's patients and 6 healthy controls, totaling 13,214 cells (6,541 control, 6,673 AD) and 10,850 genes across 8 cell types or subtypes.



(a) cosine similarity (b) median L1 distance (c) RMSE

Figure 3: Comparison and evaluation of imputation metrics.

3.1.2 BASELINE.

The algorithm's performance is assessed against various advanced imputation and clustering methods for scRNA-seq data. DeepImpute Wang et al. (2021b) utilizes sub-neural networks with a dropout layer for gene imputation. scIGANs Zheng et al. (2017) and GE-Impute Wu & Zhou (2022) apply adversarial and neural network models to fill missing values. GNNImpute Xu et al. (2021) and scCDG Wang et al. (2021a) leverage graph attention convolution and GNNs to consolidate similar cell information, reduce noise, and transform high-dimensional data into low-dimensional representations. scClust Chorbadjiev et al. (2020) implements hierarchical clustering with predetermined limits, while PARC Stassen et al. (2020) and graph-sc Ciortan & Defrance (2022) enhance processing speed through graph pruning and feature extraction using graph autoencoders. DCAk-means Eraslan et al. (2019) joins AE noise reduction with K-means clustering, and scDeepCluster Tian et al. (2019) minimizes loss using a denoising AE and ZINB model. Seurat Satija et al. (2015) adopts the Louvain method for clustering.

3.1.3 EVALUATION METRICS.

To evaluate the imputation effect of scICLGAE and the clustering performance of scCapsZB, we selected three common similarity metrics: cosine similarity, median L1 distance, root mean square error (RMSE) and two clustering evaluation indicators: normalized mutual information (NMI) and adjusted Rand index (ARI).

3.2 RESULTS

3.2.1 COMPARISON OF IMPUTATION METRICS:

To assess scICLGAE's imputation performance, we used the Zeisel and Klein datasets with gold-standard cell type labels. Following scVI's Leave-one-out strategy Chen et al. (2024), dropout events were simulated via Splatter Zappia et al. (2017) by randomly zeroing some non-zero gene expressions. Three missing rates (10%, 30%, 50%) were set. We evaluated scICLGAE against four methods (DeepImpute, scIGANs, GNNImpute, GE-Impute) using Cosine Similarity, Median L1 Distance, and RMSE between original and imputed gene expression matrices. Results are in Fig. 3.

scICLGAE shows comparable performance in cosine similarity to recent methods like those based on deep neural, generative adversarial, and graph neural networks. It attains optimal or near-optimal results on most datasets (excluding the Zeisel dataset with 50% missing rate). For L1 distance and RMSE, it excels except on the Klein dataset with 50% missing rate, with more significant improvement. This is because cosine similarity emphasizes vector direction, while L1 and RMSE focus on vector attributes, better reflecting the preservation of original cell expression. Overall, compared with recent deep learning imputation methods, scICLGAE has made notable progress and

can effectively impute missing values in the original gene expression matrix, facilitating subsequent analyses.

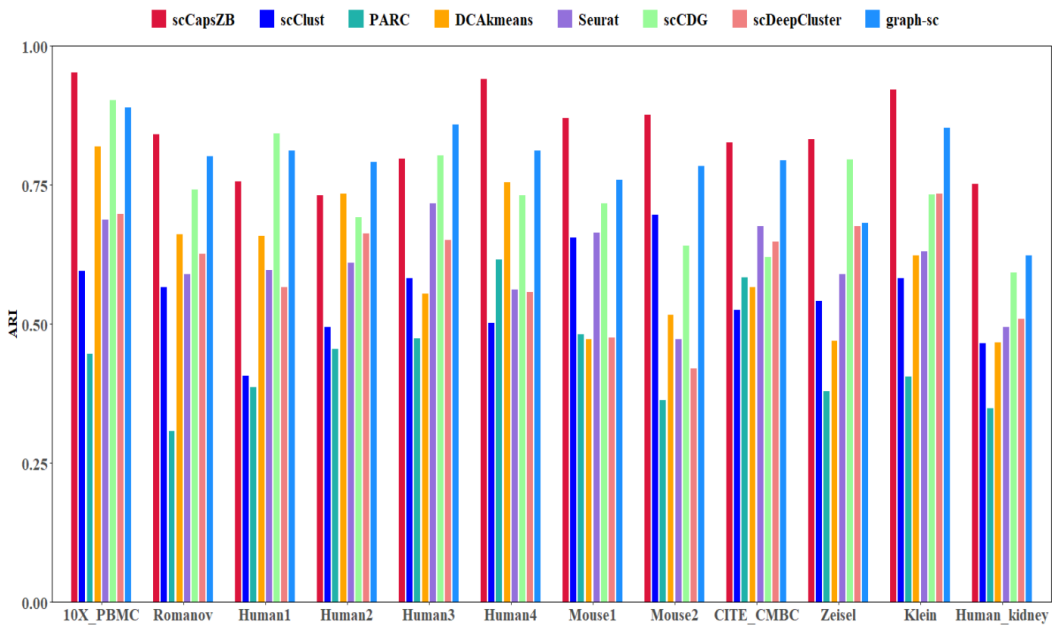


Figure 4: Comparison of ARI metrics.

3.2.2 Comparison of Clustering Metrics:

To evaluate scCapsZB's clustering, it was tested on 12 datasets with 7 other algorithms, using ARI and NMI. Among 9 datasets, scCapsZB led in ARI, surpassing neural network-based ones by 5.59% - 23.22% and statistical learning-based ones by up to 173.95%, as shown in the Fig. 5. Its ARI exceeded 0.9 on some datasets. Even ranking third on a few datasets, the gaps were minor. Performance on certain datasets may be due to training data or ZINB. The results shown in the NMI values were the same as those in the ARI.

Clustering metric comparison reveals scCapsZB's superiority over methods ignoring cell relations and topology, though its edge over advanced graph neural methods is subtle. The Romanov dataset (where scCapsZB's ARI is close to graph - sc and scCDG) was chosen. Their Embeddings were UMAP-projected for visualization, as shown in Fig. 6. scCapsZB shows good clustering with clear separation and compactness, highlighting its techniques' merits. graph - sc has one less cluster, leading to cell misclassification and biological misalignment despite decent overall metrics. scCDG's cluster number is correct, but some clusters crowd due to potential over-denoising, losing data heterogeneity and affecting dimensionality reduction.

3.2.3 Imputation and Clustering Analysis of AD Sequencing Data:

To investigate cellular changes before and after Alzheimer's disease (AD) onset and assess the performance of scICLGAE and scCapsZB on large-scale single-cell RNA-seq data, we first imputed the GSE138852 dataset using scICLGAE and then clustered the results with scCapsZB. As shown in Fig. 7(a), eight cell clusters were identified, with the right panel illustrating their distribution in control and AD samples. Unlike standard datasets, GSE138852 shows cluster adhesion and reduced separability, likely due to glial cell characteristics. However, the combination of imputation and capsule network prior enables biologically meaningful clustering; for instance, Oligo_1 and Oligo_2 remain close due to similar gene expression, while other cell types are more distinct. The observed shifts in cell-type proportions between groups provide insights into AD-related changes and algorithmic performance. In comparison, GE-Impute underestimates cluster numbers, sometimes merging biologically distinct subtypes and missing genuine heterogeneity.

Fig. 8 indicate significant changes in the proportion of different cell types. The number of Oligo_3, astrocytes, and oligodendrocyte precursor cells has decreased, while the number of Oligo_1, Oligo_2, and Oligo_4 has increased. Recent studies Kedia & Simons (2025) in AD have revealed the reasons for these changes.

3.2.4 Ablation Study.

The scCapsZB model combines a ZINB autoencoder and a capsule network. Two conditions were tested: only-ZINB and only-Caps, both using the same preprocessing. Fig. 9 shows the results of these tests. scCapsZB performed best on the 10X_PBMC, Romanov, Zeisel, and Klein datasets. On the Human1 dataset, scCapsZB and only-ZINB had similar results, while only-Caps performed the worst due to missing diverse gene cell data. On the Human2 dataset, only-Caps performed the best, and only-ZINB performed the worst because the dataset didn't fit the ZINB model well. Overall combining the capsule network and the ZINB autoencoder improved clustering performance, showing the validity and effectiveness of all its components.

3.3 KEGG Pathway Enrichment Analysis

In the R environment, the ClusterProfiler package was directly downloaded and used to perform KEGG enrichment analysis on the 2000 highly differentially expressed genes retained after imputation in AD cells. The gene pathways were sorted based on the enrichment factor, and the top 30 pathways were selected for visualization, as shown in Figure 10. The biological pathway enrichment analysis revealed that several pathways were highly positively enriched in AD cells, including Sphingolipid metabolism, Alzheimer disease, and Huntington disease-related pathways, all of which exhibited significant p-values, high enrichment factors, and a large aggregation of genes. These pathways represent the pathogenic routes that contribute to neural degeneration and even mortality in patients; thus, further research on the genes within these pathways could help elucidate the pathological mechanisms underlying Alzheimer's and other neurodegenerative diseases. Notably, due to recent extensive research on COVID-19 gene pathways and updates to the KEGG database, the Coronavirus disease (COVID-19) pathway was also significantly enriched in AD patients—likely reflecting the advanced age of these patients, which makes them more susceptible to COVID-19 and results in pronounced related gene expression characteristics.

4 Conclusion

Across multiple public datasets and an Alzheimer's disease cohort, our framework consistently improves imputation fidelity and clustering accuracy over existing state-of-the-art approaches, demonstrating its robustness across different tissue types, sequencing depths, and experimental conditions. The high-quality embeddings produced by our method enable not only the recovery of biologically meaningful cell states but also the detection of rare or previously overlooked subpopulations that may play critical roles in neurodegeneration. In particular, our analysis highlights transcriptional programs related to immune activation, synaptic dysfunction, and glial reactivity, which are highly relevant to Alzheimer's disease pathology and may represent potential therapeutic targets. These findings underscore the importance of integrating graph-based representation learning with biologically informed clustering to better resolve cellular heterogeneity in complex tissues. Furthermore, quantitative benchmarking shows that our approach achieves lower reconstruction error and higher adjusted Rand index than competing methods, even on noisy and sparse datasets, confirming its generalizability. The modular nature of our pipeline allows it to be easily extended to multimodal single-cell data, such as scATAC-seq or spatial transcriptomics, paving the way for comprehensive cross-omic integration and deeper mechanistic insights into disease progression.

Together, scICLGAE and scCapsZB form a robust and extensible pipeline for denoising, embedding, and interpreting large-scale single-cell datasets. The modular design of the framework allows seamless integration with downstream analyses such as trajectory inference, differential expression testing, and cell–cell communication modeling, thereby broadening its applicability to diverse biological questions. We anticipate that this work will facilitate mechanistic discovery and biomarker development for Alzheimer's disease and other neurodegenerative disorders, and we acknowledge the use of ChatGPT for language polishing and refinement during manuscript preparation.

REFERENCES

Matthew Amodio, David Van Dijk, Krishnan Srinivasan, William S Chen, Hussein Mohsen, Kevin R Moon, Allison Campbell, Yujiao Zhao, Xiaomei Wang, Manjunatha Venkataswamy, et al. Exploring single-cell data with deep multitasking neural networks. *Nature methods*, 16(11):1139–1145, 2019.

Cédric Arisdakessian, Olivier Poirion, Breck Yunits, Xun Zhu, and Lana X Garmire. Deepimpute: an accurate, fast, and scalable deep neural network method to impute single-cell rna-seq data. *Genome biology*, 20:1–14, 2019.

Raul Balmez, Alexandru Brateanu, Ciprian Orhei, Codruta O Ancuti, and Cosmin Ancuti. Depthlux: Employing depthwise separable convolutions for low-light image enhancement. *Sensors*, 25(5): 1530, 2025.

Maayan Baron, Adrian Veres, Samuel L Wolock, Aubrey L Faust, Renaud Gaujoux, Amedeo Vetere, Jennifer Hyoje Ryu, Bridget K Wagner, Shai S Shen-Orr, Allon M Klein, et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. *Cell systems*, 3(4):346–360, 2016.

Keting Cen, Huawei Shen, Qi Cao, and Xueqi Cheng. A survey on graph contrastive learning. *Journal of Chinese Information Processing*, 37(5):1–21, 5 2023.

Gangqi Chen, Zhaoyong Mao, Junge Shen, and Dongdong Hou. Enhancing classification efficiency in capsule networks through windowed routing: tackling gradient vanishing, dynamic routing, and computational complexity challenges. *Complex & Intelligent Systems*, 11(1):45, 2025.

Hegang Chen, Yuyin Lu, Zhiming Dai, Yuedong Yang, Qing Li, and Yanghui Rao. Comprehensive single-cell rna-seq analysis using deep interpretable generative modeling guided by biological hierarchy knowledge. *Briefings in Bioinformatics*, 25(4):bbae314, 2024.

Lubomir Chorbadjiev, Jude Kendall, Joan Alexander, Viacheslav Zhygulin, Junyan Song, Michael Wigler, and Alexander Krasnitz. Integrated computational pipeline for single-cell genomic profiling. *JCO Clinical Cancer Informatics*, 4:464–471, 2020.

Madalina Ciortan and Matthieu Defrance. Gnn-based embedding for clustering scrna-seq data. *Bioinformatics*, 38(4):1037–1044, 2022.

Alejandro Cisterna-García, Aleksandra Beric, Muhammad Ali, Jose Adrian Pardo, Hsiang-Han Chen, Maria Victoria Fernandez, Joanne Norton, Jen Gentsch, Kristy Bergmann, John Budde, et al. Cell-free rna signatures predict alzheimer's disease. *Iscience*, 26(12), 2023.

Jiequan Cui, Zhuotao Tian, Zhisheng Zhong, Xiaojuan Qi, Bei Yu, and Hanwang Zhang. Decoupled kullback-leibler divergence loss. *arXiv preprint arXiv:2305.13948*, 2023.

Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. Generalized louvain method for community detection in large networks. In *2011 11th international conference on intelligent systems design and applications*, pp. 88–93. IEEE, 2011.

Yue Deng, Feng Bao, Qionghai Dai, Lani F Wu, and Steven J Altschuler. Scalable analysis of cell-type composition from single-cell transcriptomics using deep recurrent learning. *Nature methods*, 16(4):311–314, 2019.

Jiarui Ding, Anne Condon, and Sohrab P Shah. Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. *Nature communications*, 9(1):2002, 2018.

Kevin Duarte, Brian Chen, Nina Shvetsova, Andrew Rouditchenko, Samuel Thomas, Alexander Liu, David Harwath, James Glass, Hilde Kuehne, and Mubarak Shah. Routing with self-attention for multimodal capsule networks. *arXiv preprint arXiv:2112.00775*, 2021.

Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):390, 2019.

De-Chao Feng, Wei-Zhen Zhu, Jie Wang, Deng-Xiong Li, Xu Shi, Qiao Xiong, Jia You, Ping Han, Shi Qiu, Qiang Wei, et al. The implications of single-cell rna-seq analysis in prostate cancer: unraveling tumor heterogeneity, therapeutic implications and pathways towards personalized therapy. *Military Medical Research*, 11(1):21, 2024.

Christopher Heje Grønbech, Maximillian Fornitz Vording, Pascal N Timshel, Casper Kaae Sønderby, Tune H Pers, and Ole Winther. scvae: variational auto-encoders for single-cell gene expression data. *Bioinformatics*, 36(16):4415–4422, 2020.

Alexandra Grubman, Gabriel Chew, John F Ouyang, Guizhi Sun, Xin Yi Choo, Catriona McLean, Rebecca K Simmons, Sam Buckberry, Dulce B Vargas-Landin, Daniel Poppe, et al. A single-cell atlas of entorhinal cortex from individuals with alzheimer's disease reveals cell-type-specific gene expression regulation. *Nature neuroscience*, 22(12):2087–2097, 2019.

Dezhi Huang, Naya Ma, Xinlei Li, Yang Gou, Yishuo Duan, Bangdong Liu, Jing Xia, Xianlan Zhao, Xiaoqi Wang, Qiong Li, et al. Advances in single-cell rna sequencing and its applications in cancer research. *Journal of hematology & oncology*, 16(1):98, 2023.

Dongqing Huang, Wang Liao, Jun Li, Tongkai Chen, Xinlu Wang, Ruiyue Zhao, Lingyan Zhang, Xuefeng Yu, Dong Zheng, and Ping Luan. Alzheimer's disease: Status of low-dimensional nanotherapeutic materials. *Advanced Functional Materials*, 34(4):2302015, 2024.

Shreeya Kedia and Mikael Simons. Oligodendrocytes in alzheimer's disease pathophysiology. *Nature Neuroscience*, pp. 1–11, 2025.

Peter V Kharchenko. The triumphs and limitations of computational methods for scrna-seq. *Nature methods*, 18(7):723–732, 2021.

Allon M Klein, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A Weitz, and Marc W Kirschner. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell*, 161(5):1187–1201, 2015.

Shiqi Luo, Jingying Li, Yan Yang, Yang Jiang, Ying Jie, and Wei Ge. Spatial transcriptomics and single-cell rna-sequencing revealed dendritic cell-mediated inflammation in keratoconus. *The Ocular Surface*, 2025.

Vittorio Mazzia, Francesco Salvetti, and Marcello Chiaberge. Efficient-capsnet: Capsule network with self-attention routing. *Scientific reports*, 11(1):14634, 2021.

Eleni P Mimitou, Anthony Cheng, Antonino Montalbano, Stephanie Hao, Marlon Stoeckius, Mateusz Legut, Timothy Roush, Alberto Herrera, Efthymia Papalexi, Zhengqing Ouyang, et al. Multiplexed detection of proteins, transcriptomes, clonotypes and crispr perturbations in single cells. *Nature methods*, 16(5):409–412, 2019.

Yukiteru Nakayama, Katsuhito Fujiu, Tsukasa Oshima, Jun Matsuda, Junichi Sugita, Takumi James Matsubara, Yuxiang Liu, Kohsaku Goto, Kunihiro Kani, Ryoko Uchida, et al. Heart failure promotes multimorbidity through innate immune memory. *Science Immunology*, 9(95):eade3814, 2024.

Sabrina Rashid, Sohrab Shah, Ziv Bar-Joseph, and Ravi Pandya. Dhaka: variational autoencoder for unmasking tumor heterogeneity from single cell genomic data. *Bioinformatics*, 37(11):1535–1543, 2021.

Roman A Romanov, Amit Zeisel, Joanne Bakker, Fatima Girach, Arash Hellysaz, Raju Tomer, Alan Alpar, Jan Mulder, Frederic Clotman, Erik Keimpema, et al. Molecular interrogation of hypothalamic organization reveals distinct dopamine neuronal subtypes. *Nature neuroscience*, 20 (2):176–188, 2017.

Rahul Satija, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature biotechnology*, 33(5):495–502, 2015.

Shobana V Stassen, Dickson MD Siu, Kelvin CM Lee, Joshua WK Ho, Hayden KH So, and Kevin K Tsia. Parc: ultrafast and accurate clustering of phenotypic data of millions of single cells. *Bioinformatics*, 36(9):2778–2786, 2020.

Valentine Svensson, Adam Gayoso, Nir Yosef, and Lior Pachter. Interpretable factor models of single-cell rna-seq via variational autoencoders. *Bioinformatics*, 36(11):3418–3421, 2020.

Tian Tian, Ji Wan, Qi Song, and Zhi Wei. Clustering single-cell rna-seq data with a model-based deep learning approach. *Nature Machine Intelligence*, 1(4):191–198, 2019.

Hai-Yun Wang, Jian-Ping Zhao, Yan-Sen Su, and Chun-Hou Zheng. sccdg: a method based on dae and gcn for scrna-seq data analysis. *IEEE/ACM transactions on computational biology and bioinformatics*, 19(6):3685–3694, 2021a.

Jingshu Wang, Divyansh Agarwal, Mo Huang, Gang Hu, Zilu Zhou, Chengzhong Ye, and Nancy R Zhang. Data denoising with transfer learning in single-cell transcriptomics. *Nature methods*, 16 (9):875–878, 2019.

Juexin Wang, Anjun Ma, Yuzhou Chang, Jianting Gong, Yuexu Jiang, Ren Qi, Cankun Wang, Hongjun Fu, Qin Ma, and Dong Xu. scgnn is a novel graph neural network framework for single-cell rna-seq analyses. *Nature communications*, 12(1):1882, 2021b.

Yueying Wang, Kewei Li, Ruochi Zhang, Yusi Fan, Lan Huang, and Fengfeng Zhou. Graceimpute: A novel graph clustering autoencoder approach for imputation of single-cell rna-seq data. *Computers in Biology and Medicine*, 184:109400, 2025.

Xiaobin Wu and Yuan Zhou. Ge-impute: graph embedding-based imputation for single-cell rna-seq data. *Briefings in Bioinformatics*, 23(5):bbac313, 2022.

Chenyang Xu, Lei Cai, and Jingyang Gao. An efficient scrna-seq dropout imputation method using graph attention network. *BMC bioinformatics*, 22:1–18, 2021.

Yungang Xu, Zhigang Zhang, Lei You, Jiajia Liu, Zhiwei Fan, and Xiaobo Zhou. scigans: single-cell rna-seq imputation using generative adversarial networks. *Nucleic acids research*, 48(15): e85–e85, 2020.

Xuejiao Yin, Yi Liu, Zuopo Lv, Shengnan Ding, Liya Ma, Min Yang, Meiqiu Yao, Li Zhu, Shuqi Zhao, Yu Chen, et al. scrna-seq reveals the landscape of immune repertoire of pbmncs in imcd. *Oncogene*, pp. 1–11, 2024.

Matthew D Young, Thomas J Mitchell, Felipe A Vieira Braga, Maxine GB Tran, Benjamin J Stewart, John R Ferdinand, Grace Collord, Rachel A Botting, Dorin-Mirel Popescu, Kevin W Loudon, et al. Single-cell transcriptomes from human kidneys reveal the cellular identity of renal tumors. *science*, 361(6402):594–599, 2018.

Bin Yu, Chen Chen, Ren Qi, Ruiqing Zheng, Patrick J Skillman-Lawrence, Xiaolin Wang, Anjun Ma, and Haiming Gu. scgmai: a gaussian mixture model for clustering single-cell rna-seq data based on deep autoencoder. *Briefings in bioinformatics*, 22(4):bbaa316, 2021.

Luke Zappia, Belinda Phipson, and Alicia Oshlack. Splatter: simulation of single-cell rna sequencing data. *Genome biology*, 18(1):174, 2017.

Amit Zeisel, Ana B Muñoz-Manchado, Simone Codeluppi, Peter Lönnerberg, Gioele La Manno, Anna Juréus, Sueli Marques, Hermany Munguba, Liqun He, Christer Betsholtz, et al. Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq. *Science*, 347(6226):1138–1142, 2015.

Shixiong Zhang, Xiangtao Li, Jiecong Lin, Qiuzhen Lin, and Ka-Chun Wong. Review of single-cell rna-seq data clustering for cell-type identification and characterization. *Rna*, 29(5):517–530, 2023.

Grace XY Zheng, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8(1):14049, 2017.

Yalan Zhou, Lan Xiong, Jianhua Chen, and Qingzhong Wang. Integrative analyses of scrna-seq, bulk mrna-seq, and dna methylation profiling in depressed suicide brain tissues. *International journal of neuropsychopharmacology*, 26(12):840–855, 2023.

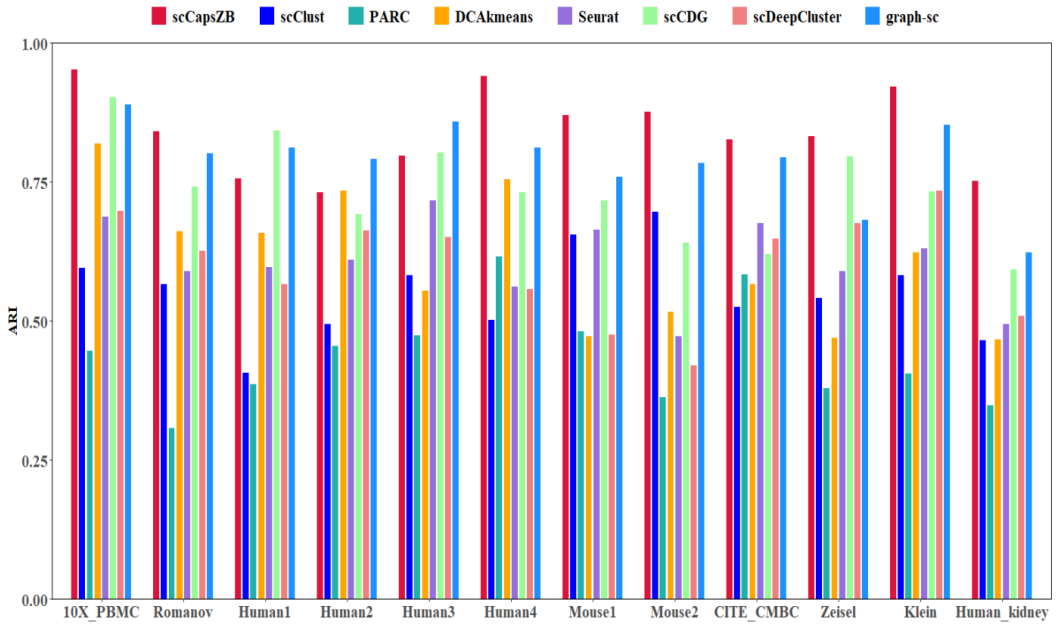


Figure 5: Comparison of ARI metrics.

A APPENDIX

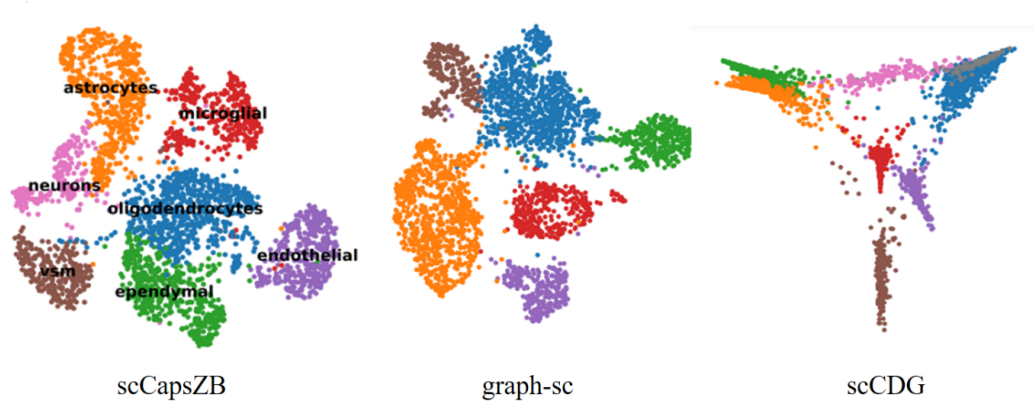


Figure 6: Visualization result comparison between scCapsZB and two graph neural network algorithms.

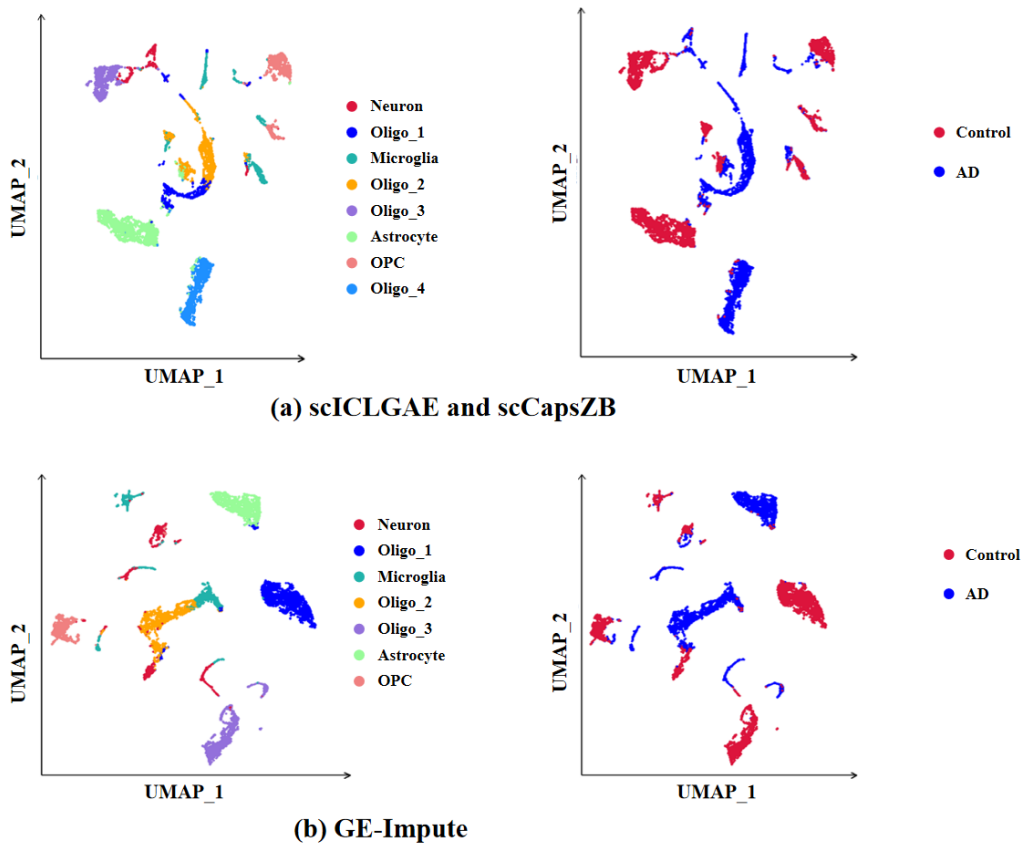


Figure 7: Distribution of imputation and clustering results in the control and patient group.

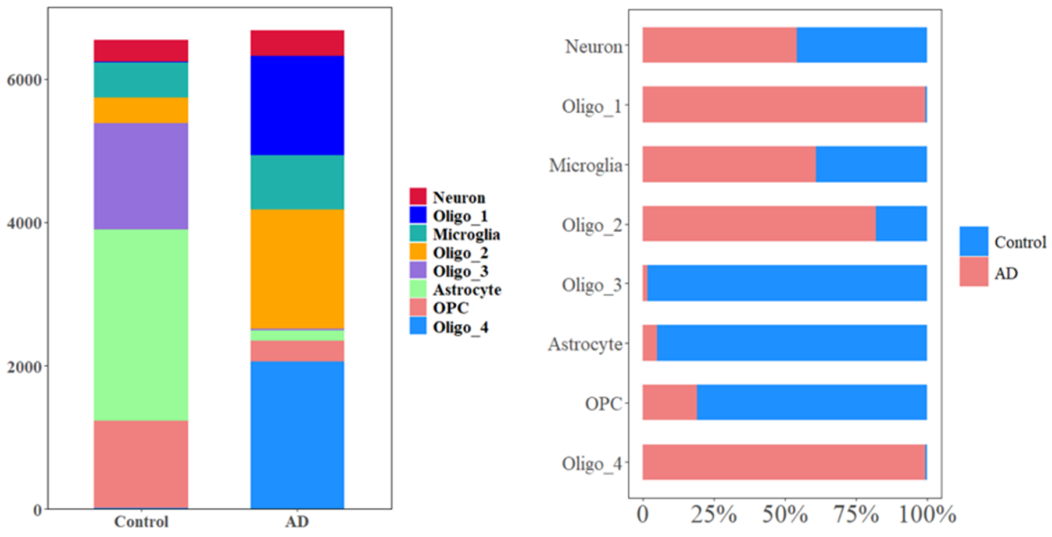


Figure 8: Comparison of the proportions of various cell numbers in the AD patient group and the control group.

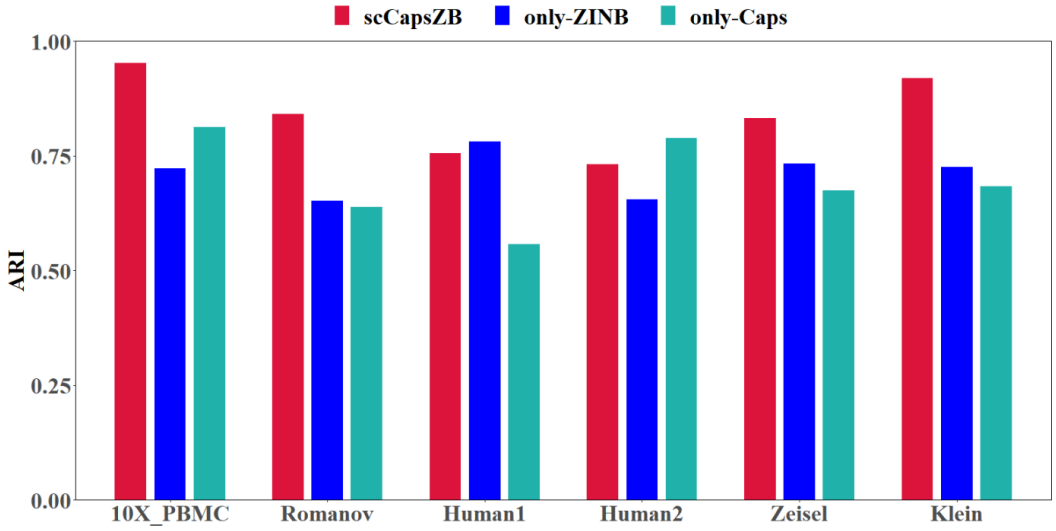


Figure 9: The results of ablation experiments.

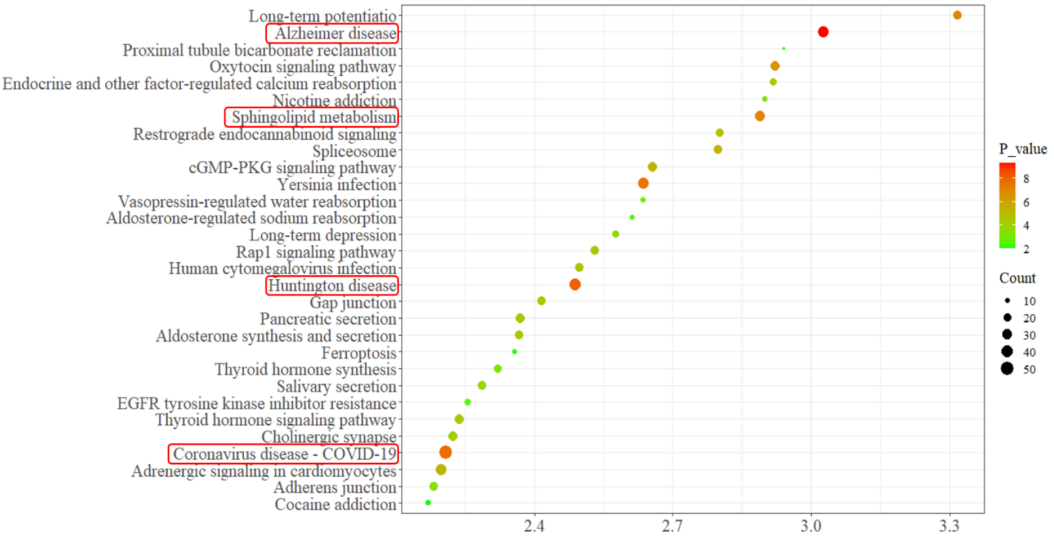


Figure 10: The results of KEGG Analysis.