A model-based constrained deep learning clustering approach for spatial-resolved single-cell data

Xiang Lin[1], Le Gao[1], Nathan Whitener[2], Ashley Ahmed[3], Zhi Wei[1*]

1. Department of Computer Science, New Jersey Institute of Technology, NJ, USA.
2. Department of Computer Science, Wake Forest University, NC, USA
3. Department of Chemistry and Chemical Biology and Biological Sciences, College of Arts and Sciences, Cornell University, NY

* Corresponding author Email: zhiwei@njit.edu

Running title: Clustering spatial-resolved single-cell data

Abstract

Spatial-resolved scRNA-seq (sp-scRNA-seq) technologies provide the potential to comprehensively profile the gene expression pattern in the tissue context. However, the development of computational methods does not catch up with the fast advances of technologies and fails to fully fulfill their potential. In this study, we develop a deep learning approach for clustering sp-scRNA-seq data, named DSSC (**D**eep **S**patial-constrained **S**ingle-cell **C**lustering). In this model, we integrate the spatial information of cells into the clustering process by two steps: 1) the spatial information is encoded by using a graphical neural network model; 2) cell-to-cell constraints are built based on the spatially expression pattern of the marker genes and added in the model to guide the clustering process. Then, a deep embedding clustering is performed on the bottle-neck layer of autoencoder by Kullback-Leibler (KL) divergence along with the learning of feature representation. DSSC is the first model which can utilize the information from both the spatial coordinates and the marker genes to guide the cell/spot clustering. Extensive experiments on both simulated and real datasets demonstrate that DSSC boosts clustering performance significantly compared to the state-of-art methods. It has a robust performance over different datasets with various cell-type/tissue organization and/or cell-type/tissue spatial dependency. We conclude that DSSC is a promising tool for clustering sp-scRNA-seq data.

Introduction

Single-cell RNA-sequencing (scRNA-seq) is a powerful, systematic biological tool that allows for transcriptomic analysis of cell heterogeneity and profiles thousands of cells at high resolution to ultimately reveal unidentified cellular subpopulations (Moncada et al. 2020). Despite this, conventional scRNA-seq alone leaves the tissue landscape undefined as cells are dissociated from their respective tissues and suspended in solution (Longo et al. 2021), neglecting and underappreciating the spatial complexity of cells and their relations to functions (Liao et al. 2021). Furthermore, cellular organization and intercellular communication networks for novel types identified by scRNA-seq remain uncharacterized unless ligand-receptor relationships are established (Skelly et al. 2018; Wang et al. 2019; Efremova et al. 2020). As cellular spatial distributions are deeply intertwined with gene expression and cell functions (Zhuang 2021), retaining this information is pivotal to further understand the collective dynamics of biological activities. Spatially resolved single-cell transcriptomics (sp-scRNA-seq) provides an exciting opportunity to map RNA molecules in their tissue locations, allowing for comprehensive profiling of cell heterogeneity (Liao et al. 2021).

Basically, the technologies to profile the spatial-resolved single-cell transcriptomics (or targeted genes) can be divided into two types: 1) hybridization-based (or called image-based) approaches, such as MERFISH, smFISH, and osmFISH. These technologies profile the physical location attributes of cells by single-molecule fluorescence in situ hybridization (Codeluppi et al. 2018; Miller et al. 2021). Pioneering studies in spatial genomics sought to explore fluorescence in situ hybridization (FISH) and digital imaging microscopy to allow for the detection of single RNA molecules in single cells (Femino et al. 1998). Thereafter, various FISH probes were developed for single-cell transcript profiling, allowing for higher accuracy and sensitivity when quantifying RNA molecules at the single-molecule level such as single-molecule in situ hybridization (smFISH) (Femino et al. 1998; Lubeck and Cai 2012; Kwon 2013; Shah et al. 2016). As some smFISH methods are multiplexed by barcoding (Femino et al. 1998; Lubeck and Cai 2012), limitations such as optical crowding and transcript length hinder marker gene targeting and cell-type mapping (Femino et al. 1998; Shah et al. 2016). Codeluppi et al. developed a non-barcoded and unamplified cyclic-ouroboros smFISH (osmFISH) method, optimized for brain tissue, to overcome the limitations of other smFISH methods (Codeluppi et al. 2018). This method demonstrates the ability to process and map large tissue areas and allows for the construction of data-driven reference atlases of human tissue. 2) Sequencing-based approaches, such as 10x Visium, and Slide-seq. A joint robust dissection of scRNA-seq data with spatially resolved single-cell transcriptomics captures a detailed illustration of the concerted cell-cell interactions within the tissue architecture. These technologies provide spatially resolved, untargeted transcriptomic profiling at the pixel level, with a pixel size of 10-100μm (Larsson et al. 2021). Using Visium as an example, it employs spatially barcoded mRNA-binding oligonucleotides grouped in spots (larger than one cell) on the tissue slides. The mRNA from the specialized tissue will bind to the oligos. Then, based on the collected mRNA, a cDNA library with spatial barcodes will be built, preserving the spatial information of spots. In this way, both the gene expression level and the cells/spots spatial organization in the tissue can be measured. The two types of technologies have their own advantages and disadvantages. Briefly, Imaging-based technologies can reach the single-cell resolution, but they can only profile a limited number of targeted genes/proteins; on the other hand, some sequencing-based technologies can profile the whole transcriptomes, but they cannot reach the single-cell resolution.

Clustering analysis is an essential step in most single-cell studies and has been studied extensively. Based on the clustering results, researchers can explore the biological activities in

cell type or subtype level, which could not be reached by studying bulk data (Shapiro et al. 2013; Kolodziejczyk et al. 2015; Kiselev et al. 2019). It has been demonstrated that some cell types, such as the neurons, have high spatial dependency and heterogeneity (Codeluppi et al. 2018). Specifically, tissues are an ensemble of cell types that interactively give rise to a specific function. It has been shown that endothelial cells in the brain are located under certain spatial patterns (Xia et al. 2019; Stoltzfus et al. 2020). Furthermore, within cells of the same type, high spatial self-affinity was measured in ependymal cells and spatial self-evasion was observed in inhibitory neurons such as microglia and astrocytes (Codeluppi et al. 2018). Cell neighbors identified by spatio-temporal organization within tissues in complex organs (e.g., the brain) provides important context to make inferences regarding cell interactions and behaviors. As such, highly accurate and sensitive mapping of tissue sections is important to reveal spatially dependent cells and can be used to understand the convolutions of cell heterogeneity. The set of neighboring cells from the spatial transcriptomics studies may provide valuable information for cell-type annotation. In other cases, such knowledge can lead to the identification of new cell types based on their neighborhood profiles. However, this entails that computational resources to analyze transcriptomic data are appropriately equipped with mechanisms to integrate the spatial features. However, traditional methods, such as Seurat (Butler et al. 2018) and SC3 (Kiselev et al. 2017), cannot utilize valuable spatial information in the clustering analysis.

Some tools have been developed for spatially transcriptomic data. Giotto is a computational method specifically designed for spatial transcriptomic data analysis that performs cell-type enrichment analysis, spatially coherent gene detection, cell neighborhood, and interaction analyses, and spatial pattern recognition (Dries et al. 2021). Unlike other computational methods that are geared towards scRNA-seq analysis but utilize spatial information to identify cell types (Stuart et al. 2019), marker genes (Svensson et al. 2018), or domain patterns (Zhu et al. 2018), Giotto is purely centered towards spatial data analysis but is capable of integrating scRNA-seq data to enhance spatial-cell type enrichment analysis. In the clustering analysis, Giotto employs graphic clustering algorithms, such as Louvain (Blondel et al. 2008), to find cell communities. BayesSpace is a Bayesian statistical method that enhances spatial transcriptomic resolution and performs clustering analysis by modeling dimensionally reduced representation of the single-cell count matrix and grouping neighboring spots to the same cluster via spatial prior (Zhao et al. 2021). BayesSpace draws a distinction in use of a t-distributed error model to identify spatial clusters and employs a Markov chain Monte Carlo to estimate model parameters. However, BayesSpace has a limited scope of application as it is majorly designed for

decomposing the data with low resolution from the sequencing-based technologies, such as the 10x Visium. Besides, some other methods, such as SpaGCN (Hu et al. 2021) and stLearn (Pham et al. 2020), employ deep neural networks, such as CNN and GCN, to analyze the sp-scRNA-seq data. These tools can also integrate the information from the H&E images to enhance the cell clustering.

It is widely demonstrated that in many tissues, especially in the brain, many marker genes have exhibited strong spatial expression dependencies (Guillozet-Bongaarts et al. 2014; Zeisel et al. 2015; Maynard et al. 2021). Therefore, the information from the markers can be used as the prior knowledge to guide the sp-scRNA-seq analyses, especially for the clustering analysis. However, none of the methods mentioned above can incorporate the marker gene information in the clustering process.

In this article, we propose a novel clustering approach for sp-scRNA-seq data, DSSC (**D**eep **S**patial-constrained **S**ingle-cell **C**lustering). DSSC integrates the prior information from both the physical organization of cells and the expression of the spatial dependent marker genes into the clustering process by a denoising graphical autoencoder with cell-to-cell constraints. Our extensive experiments indicated that DSSC outperforms the state-of-the-art methods in both simulated and real datasets, revealing that it is a promising tool for spatial-resolved single-cell data clustering.

Results

Simulation experiments

DSSC is developed for clustering spatial-resolved single-cell data by integrating the prior knowledge from cell/spot location and marker genes. The overall architecture of the DSSC model is shown in **Figure 1**. In the simulation experiments, we test the performance of DSSC on the data in different cell-type spatial organizations and dependencies. We simulated the scRNA-seq data by Splatter and placed them in the spatial locations from two real datasets from 1) osmFISH data (Figure 2a); 2) sample 151673 from spatialLIBD data (Figure 2b); We adjust the cell-type spatial dependencies by perturbing the spatial coordinates of 10%, 15%, and 20% of total cells (see details in the method section). Constraints are built based on the true labels with 5% perturbations. We compare DSSC with seven existing clustering methods including SpaGCN, stLearn, Seurat, Giotto, BayesSpace, *k*-means + PCA, and SC3. We compare both the clustering performance (measured by AC, NMI, and ARI) and the predicted label's spatial

heterogeneity (denoted as PLSH, measured by KNN ACC and Moran's I) of these methods. The results of simulation experiments are shown in Figure 4. Generally, we find that the spatial-based clustering methods (DSSC, SpaGCN, stLearn, BayesSpace, and Giotto) have higher clustering performance and PLSH than the traditional scRNA-seq clustering methods (Seurat, SC3, and *k*-means). Cell-type spatial-dependency is negatively correlated with the performance of the spatial-based clustering methods, but it has no influence on the performance of the traditional clustering methods. BayesSpace cannot encode the spatial coordinates of the osmFISH data, so the clustering performance and PLSH of it are much higher in the spatial organization 2 (Figure 2b) than in spatial organization 1 (Figure 2a). Although DSSC outperforms the competing methods in both spatial organizations, its advantage is much higher in spatial organization 1 than in spatial organization 2. In summary, these results reveal that DSSC's performance is not affected by the sequencing technologies and cell type spatial organizations, while other methods may prefer the sequencing-based technologies (such as the 10x Visium). Besides, DSSC can keep a superior performance over the competing methods under low, medium, and high cell-type dependencies (Fig a and b). Therefore, these experiments demonstrate the robustness of DSSC's performance. The statistical tests of the clustering performance between DSSC and the competing methods are shown in Table S1,S2, and S3.

Real datasets

We then test the performance of DSSC in three studies including 25 real datasets with 1 dataset from osmFISH (mouse cortex), 12 datasets from spatialLIBD (human cortex), and 12 datasets from 10x Genomics (Mouse brain, denoted as 10xMBAD). In all datasets, we compare DSSC with seven competing methods as described above. For the data from spatialLIBD and 10xMBAD, we use the markers from the original paper of spatialLIBD (Pardo et al. 2022). Since osmFISH data only has 33 genes, we only use the genes with the top Moran's I.

OsmFISH dataset

The results of the osmFISH dataset are shown in Figure 3. Since the latent dimension of SpaGCN is larger than the feature dimension of this data, we exclude SpaGCN from the competing methods for this experiment. BayesSpace cannot recognize the neighbors from the hybridization technologies, so the spatial information is not used by it for this dataset. The marker genes used here for DSSC *are Rorb* and *Syt6* (Figure 3c). As expected, the expression of these genes have high spatial dependency. We find that DSSC can identify the layer

structures in the cortex (Figure 3a). These layers are not clearly profiled by the competing methods (Figure 3b). Besides, DSSC outperforms the competing methods in both clustering performance and PLSH (Figure 3b). Some spatial-based methods, such as Giotto and stLearn, have very high KNN accuracy, but their clustering performance is much lower than DSSC. A potential reason for this result is that the spatial information overwhelms the clustering signal from the gene expression during the clustering process, resulting in the high spatial dependence but low clustering performance.

SpatialLIBD dataset

We then test all the methods on the spatialLIBD datasets (Figure 4). The marker genes used in this dataset are *PCP4* and *MOBP* (Figure 4c) for layer 5 and WM respectively from the paper of spatialLIBD. These genes show strong spatial dependencies. So, they can be used to guide the clustering process. Figure 4a shows that DSSC is the only method that can identify 5 layers in the sample 151673. Some other spatial-based methods, such as SpaGCN, and BayesSpace, cluster some cells in clumps, not in layers. Figure 4b shows that DSSC outperforms all the competing methods in the 12 spatialLIBD samples in both clustering performance and PLSH. Spatial-based methods have overall better performance than the traditional scRNA-seq clustering methods, revealing the benefits from using the spatial information. BayesSpace has the second-best performance in this dataset since it can recognize the spatial neighbors for each cell in this dataset. The statistical tests of the clustering performance between DSSC and the competing methods are shown in Table S4.

10xMBAD dataset

We then apply DSSC on the 10xMBAD dataset (Figure 5). Since this dataset has no true labels, we use silhouette score (SS) to evaluate the clustering performance. We find that all the methods have similar predicted labels' spatial heterogeneity on this dataset (Figure 5a). DSSC, BayesSpace, and SpaGCN have higher SS than other methods. To further prove the accuracy of clustering of DSSC, we identify the cluster of thalamus in a wild-type (WT) sample and an Alzheimer's Disease (AD) sample by a marker gene *Tcf7l2* (Figure 5b)(Lipiec et al. 2020) and then perform a different expression analysis (DE) between the two groups of cells. We select thalamus since it has been widely demonstrated to be associated with the memory and cognition loss during AD (Pardilla-Delgado et al. 2021; van de Mortel et al. 2021). BayesSpace and SpaGCN fail to identify the region of thalamus in the corresponding WT and AD samples (Figure 5c). The DE results are shown in Figure 5d. Many genes that overexpress in the AD

group are proved by previous studies. For example, *Olfm1* has been shown as a potential neuroprotective agent in Alzheimer's disease (Takahama et al. 2014); *Cst3* has contributions in increasing the neuronal vulnerability and impaired neuronal ability to prevent neurodegeneration (Kaur and Levy 2012); *Syn2* is related to the onset and progression of Alzheimer's disease (Kumar and Reddy 2020). As a result, in the pathway analysis of the KEGG geneset from the DE results (Figure 5e), the Alzheimer's disease pathway is significantly enriched in the thalamus of the AD sample. Another significant pathway, olfactory transduction, is also shown to be associated with AD from the previous studies (Zou et al. 2016). Spliceosome is also demonstrated to be altered in the Alzheimer transcriptomes (Koch 2018), which is significantly down-regulated in the AD sample. These downstream analyses further consolidate the clustering results of DSSC. The statistical tests of the clustering performance (SS) between DSSC and the competing methods are shown in Table S5.

Model test

We test three parameters in DSSC: 1) the number of constraints (ML and CL respectively); 2) the parameter that controls the clustering loss (gamma); 3) the number of neighbors in the *k*NN graph for GAT layers on the 12 spatialLIBD datasets (Figure 6a). We find that when the constraint number is 0 (no constraints) or 6000 (too many constraints), the performance of DSSC becomes unstable. A suitable number of constraints (here we suggest setting the constraint number around the cell number) will not only improve the clustering performance but also makes the model more stable. Compared to the model without clustering loss (gamma=0), DSSC's performance is improved when gamma is 0.01. However, a too high gamma (>1) will seriously impact the model's performance. When the numbers of neighbors are higher than 10, DSSC's performance is not sensitive to them. However, a model without considering neighbors (K=0) has much lower performance revealing the contributions from using the spatial information in clustering analysis. The results of the statistical tests of the parameter tuning experiments are in Table S6, S7, and S8. We then test DSSC on the simulated datasets with incremental numbers of cells (Figure 6b). We find that DSSC has a linearly ascending running time with the increased cell numbers. Thus, it can be easily used for analyzing large datasets. All experiments here are performed on the NVIDIA Tesla P100 with 16Gb memory.

Discussion

In this paper, we have developed a deep learning approach, DSSC, for clustering sp-scRNA-seq data. DSSC utilizes a denoising graphical autoencoder to learn a nonlinear representation

of data. Spatial information is integrated into the clustering approach by two ways: 1) constraints from marker genes; and 2) GAT encoders. To our knowledge, DSSC is the first model that can encoder the information from both spatial coordinates and marker genes for guiding the clustering. More broadly, DSSC is a flexible model in which its reconstruction loss function can be switched depending on the data structure. The available reconstruction loss includes ZINB loss, NB loss, and MSE loss to deal with various scenarios. In this study, DSSC has been tested on both simulated and real datasets. The aim of our experiments is to test the robustness of DSSC's clustering performance over the data with different cell type spatial organization and cell type spatial dependency. The evaluation has been conducted regarding two aspects, clustering performance, and space heterogeneity. Our results show that DSSC outperforms the state-of-art methods over different datasets.

Recently, a new general-purpose density estimator has been introduced by employing a symmetrical and paired generative adversarial network (GAN) architecture (Liu et al. 2021b). Adopting this GAN architecture, a new method scDEC enables simultaneous learning of latent features and cell clustering and shows its superiority over competing methods in scATAC-seq analysis (Liu et al. 2021a). If spatial information could be accommodated in this GAN architecture, we may expect similar promising improvement in analysis of sp-scRNA-seq data. We leave such exploration to future work.

One limit of the current model is its compatibility with the datasets with low spatial dependency. DSSC employs the spatial information of cells to boost the clustering performance, while not all tissue types have a high spatial dependency. Besides, for approaches like 10x Visium, our model is dependent on the assumption that all the cells in one spot are in the same cell type. In the future investigation, this issue can be solved by doing the decomposition of spots. The latent representation of DSSC can be used for many downstream analyses, such as the cell-to-cell communication and trajectory analysis.

Methods

Denoising autoencoder

The autoencoder is a neural network for learning a nonlinear representation of data (Hinton and Salakhutdinov 2006). It receives corrupted data with artificial noises and reconstructs the original data (Vincent et al. 2008). It is able to learn a robust latent representation for noisy data.

We use the denoising autoencoder for the highly noisy count data of cells. Let's denote the preprocessed counts data as X and the corrupted data as X_c, formally:

$$X_c = X + \sigma * n$$

where n is the artificial noise in standard Gaussian distribution (with mean=0 and variance=1), and σ controls the weights of n. We set σ as 0.1.

Next, we use an autoencoder to reduce the dimension of count data. Encoders (*E*) are graphical attention networks (GAT) layers and decoders (*D*) are fully connected neural networks. Denoting the latent space as Z and the learnable weights of encoder as w, the encoder can be shown as $Z = E_w(X_c)$. The GAT layers in E can be formalized as:

$$X_i = \begin{cases} ELU(BatchNorm(GAT_i^{(K)}(X_c, A))) & if\ i = 1 \\ ELU(BatchNorm(GAT_i^{(K)}(ELU(X_{i-1}), A))) & if 1 < i < L \\ GAT_i^{(K)}(ELU(X_{i-1}), A) & if\ i = L \end{cases}$$

Where X_i is the output of the i*th* layer. $GAT_i^{(K)}$ is the ith GAT layer with K heads. *L* is the total layers of encoder. A is the adjacent matrix of a kNN graph G built based on the spatial coordinates of cells. Specifically, the distance between two cells i and j is measured by Euclidean distance:

$$M_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

where x and y indicate the coordinates of cells i and j in a two-dimensional physical space. Then A_{ij} $(i, j \in 1, 2, 3, \ldots, N)$ is built by:

$$A_{ij} = \begin{cases} 1, & if\ i\ is\ the\ K\ nearest\ neighbor\ of\ j\ on\ the\ physical\ space \\ 0, & otherwise \end{cases}$$

A is then normalized by $\tilde{A} = \bar{A} \cdot A \cdot \bar{A}$, where $\tilde{A}$ is the normalized graph, $\bar{A}$ is $diag(power(\sum_j^N A_j, -0.5))$ and $(\cdot)$ means dot product. Then $\tilde{A}$ is used as the input for the GAT encoder. In this study, we set the number of heads as 3. The decoder is $X' = D_{w'}(Z)$, where w' are the learnable weights for the decoder and X' is the reconstructed counts from the decoder. The ELu activation function (Nair and Hinton 2010) and batch normalization are used for all the

hidden layers in the encoder and decoder except the bottleneck layer. In the default setting, we use two layers of encoder and decoder. The default bottleneck layer is set as 32.

We employ a zero-inflated negative binomial (ZINB) model in the reconstruction loss function to characterize the zero-inflated and over-dispersed count data (Tian et al. 2019). Note, the raw count data, not the normalized data, is used in the ZINB model (Lopez et al. 2018; Eraslan et al. 2019; Tian et al. 2019). Let X_{ij} be the count for cell i and gene j in the raw count matrix. The NB distributions are parameterized by μ_{ij} and θ_{ij} as means and dispersions respectively. Formally:

$$NB\left(X_{ij}\middle|\mu_{ij},\theta_{ij}\right)=\frac{\Gamma\left(X_{ij}+\theta_{ij}\right)}{X_{ij}!\,\Gamma\left(\theta_{ij}\right)}\left(\frac{\theta_{ij}}{\theta_{ij}+\mu_{ij}}\right)^{\vartheta_{ij}}\left(\frac{\theta_{ij}}{\theta_{ij}+\mu_{ij}}\right)^{X_{ij}}$$

Then, ZINB distribution is parameterized by the negative binomial and an additional coefficient π_{ij} for the probability of dropout events (zero mass):

$$ZINB\left(X_{ij}\middle|\mu_{ij},\theta_{ij},\pi_{ij}\right)=\pi_{ij}\delta_0\left(X_{ij}\right)+(1-\pi_{ij})NB(X_{ij}|\mu_{ij},\theta_{ij})$$

The loss function of ZINB-based autoencoder for the count data is defined as:

$$L_{ZINB}=\sum_{ij}-\log\left(ZINB(X_{ij}\,|\mu_{ij},\theta_{ij},\pi_{ij})\right)$$

We use independent fully connected layers to estimate these parameters in ZINB loss functions. We add three independent fully connected layers M, Θ, and Π after the last hidden layer of the decoder which outputs the reconstructed matrix X'. The parameter layers are defined as:

$$M=diag(s_i)\ \times\ \exp\left(w_\mu X'\right);$$
$$\Theta=\exp\left(w_\theta X'\right);$$
$$\Pi=\exp\left(w_\pi X'\right);$$

where M, Θ, and Π are the matrix of estimated mean, dispersion, and drop-out probability for the ZINB loss of count data. w_μ, w_θ, and w_π are the learnable weights for them, respectively. The size factor s_i for the cell i was calculated in the preprocessing step.

The sizes of layers are set to (128, 32) for the GAT encoder and (32, 128) for the fully connected decoder.

Deep embedded clustering

Our model has two learning stages, a pretraining stage and a clustering stage. In the pretraining stage, we only train the autoencoder without considering the clustering loss and the constraint loss (see details below). Then, in the clustering stage, we simultaneously optimize the autoencoder and the clustering results. We perform unsupervised clustering on the latent space of the autoencoder (Xie et al. 2016). Our autoencoder transfers the input matrix to a low dimensional space *Z*. The clustering loss is defined as the Kullback-Leibler (KL) divergence between the soft label distribution *Q'* and the derived target distribution *P'*:

$$L_{Clustering} = KL(P' \parallel Q') = \sum_i \sum_k p'_{ik} \log \frac{p'_{ik}}{q'_{ik}}$$

where the soft label q'_{ik} measures the similarity between z_i and cluster center μ_k by Student's t-kernel (Maaten and Hinton 2008). The cluster center μ_k is initialized by applying a *k*-means on the bottleneck layer from the pretraining stage, and then updated per batch in the clustering stage. Formally, q'_{ik} is defined as:

$$q'_{ik} = \frac{(1 + \| z_i - \mu_k \|^2)^{-1}}{\sum_{k'} (1 + \| z_i - \mu_{k'} \|^2)^{-1}}$$

The target distribution P' which emphasizes the more certain assignments is derived from *Q'*. Formally p'_{ik} is defined as:

$$p'_{ik} = \frac{q'^2_{ik} / \sum_i q'_{ik}}{\sum_{k'} (q'^2_{ik'} / \sum_i q'_{ik'})}$$

During the training process, Q' and clustering loss are calculated per batch and P' is updated per epoch. This clustering loss will improve the initial estimate (from *k*-means) in each iteration by learning from the high-confident cell assignments, which in turn helps to improve the low-confident ones (Xie et al. 2016).

Autoencoder with pairwise constraints

Based on the autoencoder architecture, we add pairwise constraints of cells (Tian et al. 2021) on the latent space according to the expression of the marker genes. Similar to scDCC (Tian et al. 2021), we employ the must-link constraints which pull two cells to have similar soft labels if they have similar expression patterns of one or more marker genes, and cannot-link constraints which encourage two cells to have different soft labels if they have different expression patterns of one or more marker genes.

Constraints are built by six steps, considering both the spatial coordinates and the gene expression of the cells: 1) select the marker genes from literatures; 2) for each marker, say gene A, smooth the expression of A by averaging the normalized count data of the k (k is defined according to the technology, we set it as 6 in this study) spatial neighbors of each cells; 3) define the cells with the top 5% (cutoff1) expression of A as high, otherwise as low; 4) collect the cells as the confident cells if more than half (cutoff2) of its neighbors (and itself) have the high smoothed expression of A; 5) repeat step 2-4 for all the marker genes; 6) since each marker gene represents a cell type (or a layer in cortex), we connect two confident cells by a must-link if they are selected by the markers for the same cell type (or layer); otherwise, we connect two confident cells by a cannot-link if they are selected by the markers for different cell types (or layers). It is noted that there is a tradeoff between the coverage and the reliability of constraints. A higher cutoff will decrease the coverage of constraints but also reduce the false positive links. We denote the constraints sampled here as the pool of constraints.

The must-link and cannot-link constraints loss are defined as:

$$L_{ml} = \sum_{(i,j)\in ML} log \sum q_i \times q_j$$

$$L_{cl} = \sum_{(i,j)\in CL} \log (1 - \sum q_i \times q_j)$$

Where q is the soft labels described in the clustering section above. Must-links and cannot-links are used for training the model alternately and are updated (resampled) during the training. The number of constraints can be set according to the cell numbers. For example, for a dataset with 4000 cells, we sample 4000 must-links and cannot-links, respectively.

Combining the pairwise constraint loss, reconstruction loss, and clustering loss, the total loss of the DSSC is:

$$L = L_{ZINB} + \gamma * L_{Clustering} + \beta * L_{ml} + \lambda * L_{cl}$$

Where γ, β, and λ are the coefficients for the clustering loss, must-link loss, and cannot-link loss respectively. In the experiments of this study, γ is set to 0.01, β and λ are set to 0.1 and 1 respectively (see parameter tuning in the result section).

Model implementation

This model is implemented in Python3 using PyTorch (Paszke et al. 2017). Adam with AMSGrad variant (Kingma and Ba 2014; Reddi et al. 2018) with an initial learning rate = 0.001 is used for the pretraining stage and the clustering stage. The kNN graph is calculated by the "kneighbors_graph" function from the scikit-learn package. The top 2000 HVGs are selected to train the model. We pretrain the autoencoders for 200 epochs before entering the clustering stage. In the beginning of the clustering stage, we initialize *K* centroids by the *k*-means algorithm. During the clustering stage, reconstruction loss and clustering loss are optimized first. Then, constraint losses are optimized with reconstruction loss. ML and CL losses are optimized alternately. The centroids are also continuously updated by the learning process. Before each epoch, constraints are randomly sampled from the constraint pools. The soft label distribution *Q'* is calculated in each batch and the derived target distribution *P'* is updated after each epoch. The convergence threshold for the clustering stage is that less than 0.1% of labels are changed per epoch. The marker genes used in this study are from the original paper of the spatialLIBD datasets (Maynard et al. 2021), including *PCP4*, *MOBP*, *FABP7*, *AQP4, CARTPT, KRT17* and so forth. More markers can be added if necessary. It is noted that we test the Moran's I and check the expression pattern of each marker before using it (See supplementary notes for details). If a marker has very low spatial dependency in a dataset, we exclude it for building constraints. For the osmFISH dataset with only 33 genes, we just use the genes with the highest spatial dependency (Moran' I) as the markers. All experiments of DSSC in this study are conducted on NVIDIA Tesla P100 with 16Gb memory.

Marker and gene selection

Before running the autoencoder model, we use Moran's I statistic (Moran 1950; Miller et al. 2021) to measure the gene spatial heterogeneity. I_k^{gene} stands for the Moran's I of gene k, which is defined as:

$$I_k^{gene} = \frac{N}{\sum_{i=1}^{N}\sum_{j=1}^{N} A} \cdot \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

Where $\underline{x}$ is the mean value of the normalized counts of the gene K over all cells. A is the kNN graph from spatial information of cells. Marker genes with low Moran' I will not be used to build constraints. It is noted that gene filtering has a tiny influence on the performance of the osmFish dataset since it only has 33 genes. These genes are all selected by the researchers so all of them are important for all or a part of cells in the tissue. In our experiments, because of the low feature number, we only select 30 HVGs out of 33 genes. On the other hand, the sequencing-based methods profile the whole transcriptome (>20000 genes). Many genes are not informative for clustering and even mislead the clustering. So, feature selection is essential for these datasets. In our experiments, we select the top 2000 highly variable genes (HVGs) for training DSSC. An optional feature selection approach is to use the genes with the top Moran's I.

Evaluation metrics for clustering performance

Adjusted Rand Index (ARI) (Hubert and Arabie 1985), Normalized Mutual Information (NMI)(Alexander and Joydeep 2003), and Clustering Accuracy (AC) are used as metrics to evaluate the performance of different methods.

Adjusted Rand Index measures the agreements between two sets U and G. Assuming a is the number of pairs of two cells in the same group in both U and G; b is the number of pairs of two cells in different groups in both U and G; c is the number of pairs of two cells in the same group in U but in different groups in G; and d is the number of pairs of two cells in different groups in U, but in the same group in G. The ARI is defined as:

$$ARI = \frac{\binom{n}{2}(a+d) - [(a+b)(a+c)+(c+d)(b+d)]}{\binom{n}{2}^2 - [(a+b)(a+c)+(c+d)(b+d)]}$$

Let U = {U1, U2, …, C_{tu}} and G = {G1, G2, …, G_{tg}} be the predicted and ground truth labels on a dataset with n cells. NMI is defined as:

$$NMI = \frac{I(U,G)}{\max\{H(U), H(V)\}}$$

Where I(U,G) represents the mutual information between U and G and is defined as:

$$I(U,G) = \sum_{p=1}^{tu} \sum_{q=1}^{tg} |U_p \cap G_q| \log \frac{n|U_P \cap G_q|}{|U_p| \times |G_q|}$$

And H(U) and H(G) are the entropies:

$$H(U) = -\sum_{p=1}^{tu} |U_p| \log \frac{|U_p|}{n}$$

$$H(G) = -\sum_{p=1}^{tg} |G_p| \log \frac{|G_p|}{n}$$

AC is defined as the best matching between predicted and true clusters, which is given as:

$$AC = \max_m \sum_{i=1}^{n} 1\frac{\{\widehat{l_i} = m(l_i)\}}{n}$$

Where $\widehat{l_i}$ are the true labels and l_i are the predicted labels from clustering algorithms. n is the number of cells and m is the number of all possible one-to-one mapping between $\widehat{l_i}$ and l_i. The best mapping is found by the Hungarian algorithm (Kuhn 1955).

The silhouette score (SS) is used to measure the clustering performance without labels. It compares how similar a cell is to its own cluster compared to other clusters. The silhouette score ranges from −1 to +1, where a high value indicates a better clustering. Let's denote the silhouette score of cell i as S_i, so we have:

$$S_i = \begin{cases} 1 - \frac{a_i}{b_i} & if\ a_i < b_i \\ 0 & if\ a_i = b_i \\ \frac{b_i}{a_i} - 1 & if\ a_i > b_i \end{cases}$$

Where a_i stands for how well a cell i is assigned to its cluster based on the distance between this cell and all other cells in its cluster; b_i stands for the smallest mean distance of the cell i to the cells in any other clusters. Then we use the mean value of S_i over all the cells as the SS for a dataset.

Evaluation metrics for spatial heterogeneity and concentration

kNN accuracy measures the consistency of the labels between each cell and its spatial neighbors. It is defined as:

$$A_{KNN} = \frac{\sum_{i=1}^{N} y_i = \hat{y_i}}{N}$$

Where y_i is the predicted label of cell i by clustering algorithms and $\hat{y}$ is the major label of its neighbors (K=20) on the physical space. We also employ a variant of Moran's I (Moran 1950) to measure the cell type spatial concentration. Let I^{label} be the I score for the predicted labels $(y_1, y_2, y_3, \ldots, y_N)$ defined as:

$$I^{label} = \frac{N}{\sum_{i=1}^{N}\sum_{j=1}^{N} A} \cdot \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij} B_{ij}}{N}$$

Where B_{ij} of cell i and j is defined as:

$$B_{ij} = \begin{cases} 1, if\ y_i = y_j \\ 0, otherwise \end{cases}$$

, and A is the kNN graph (with k=20) from spatial information of cells. The I^{label} measures the degree that the physically neighboring cells have the same label. Both metrics are ranged from 0 to 1.

Data simulation

In order to test the model's performance to integrate spatial information for clustering, we simulate the single-cell RNA-seq data by Splatter package in R (Zappia et al. 2017). The parameters for scRNA-seq data simulation are estimated from a real scRNA-seq dataset (https://support.10xgenomics.com/spatial-gene-expression/datasets) and the parameter of clustering signal (de.scale) is fixed as 0.4. Besides simulating the count data, we place each cell on a 2D space with a coordinate (x,y). The physical space and coordinates are extracted from two real datasets (osmFISH and 151507 from spatialLIBD). The regions (domains) on the physical space in the real datasets are provided by the authors. Specifically, let's denote the spot number in a layer k (from true label) as n_k and the total layer number as K. During the simulation, for a layer k, we use splatter to simulate n_k cells and randomly assign these cells to the spatial coordinates of the spots in this layer. We do this for all K layers. So, the cell number in the simulated datasets should be the same as the spot number in the real dataset. Then, we perturb the spatial coordinate of 10%, 15%, and 20% of cells to control the cell type spatial dependency. We also use the spatial coordinates from two datasets (osmFISH (Codeluppi et al.

2018) and spatialLIBD 151507 (Maynard et al. 2021)) to simulate different spatial organizations. Therefore, our simulation experiments can test the robustness of DSSC's performance in the data with different cell type spatial dependencies and cell type spatial organizations. To simulate the constraints from markers, we randomly connect 3000 cells in the same cell type (from the true label) as the must-links. We then perturb the cells in 5% must-links to simulate the real accuracy (about 95%). Similarly, we randomly connect 3000 cells in the different cell types as the cannot-links.

Real datasets

We use data from three studies including 25 sp-scRNA-seq datasets in this study. The first dataset was measured by the osmFISH technology (Codeluppi et al. 2018), and the other two datasets were sequenced by the 10x Visium technology and provided by spatialLIBD (Pardo et al. 2022) and 10x Genomics website, respectively. Specifically, the osmFISH dataset of the somatosensory cortex was downloaded from the website of Linnarsson lab (http://linnarssonlab.org/osmFISH/). This dataset contains 33 genes and 4839 cells. We did not implement the feature selection for this dataset as the low dimension of features. All 10x Visium datasets are read by the 'Load10x_Spatial' function and preprocessed by the 'SCTransform' function by Seurat in R. The 10x mouse brain Alzheimer's disease dataset is downloaded from the website (https://www.10xgenomics.com/resources/datasets). This dataset contains 12 sp-scRNA-seq data with 6 wild-type samples and 6 CRND8 APP-overexpressing transgenic (Alzheimer's Disease, AD) samples. The mice brains were sampled in 2.5, 5.7, and 13.2 month of age. Per phenotype per time-point has two replicates resulting in 12 samples in total. The spatialLIBD dataset is downloaded from R package "spatialLIBD" (Pardo et al. 2022). This dataset contains 12 spatial-resolved RNA-seq datasets which can be grouped into three spatial organizations. Specifically, sample 151507-151510 have the similar spatial organization, sample 151669-151672 have the similar spatial organization, and sample 151673-151676 have the similar spatial organization.

Count data preprocessing

The raw count data is preprocessed and normalized by the Python package SCANPY (Wolf et al. 2018). Specifically, the genes with no count are filtered out. The counts of a cell are normalized by a size factor s_i, which is calculated as dividing the library size of that cell by the median of the library size of all cells. In this way, all cells will have the same library size and become comparable. Then, the counts are logarithm transformed and scaled to have unit

variances and zero means. The treated count data is used in our denoising autoencoder model. However, we use the raw count matrix to calculate the ZINB loss (Lopez et al. 2018; Eraslan et al. 2019).

Competing methods

For consistency, we use DSSC's data preprocessing and feature selection approaches for all the competing methods. Our competing methods include *k*-means (with PCA) (https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html), Seurat (https://github.com/satijalab/seurat) (Butler et al. 2018), SC3 (https://github.com/hemberg-lab/SC3) (Kiselev et al. 2017), BayesSpace (https://github.com/edward130603/BayesSpace) (Zhao et al. 2021), Giotto (https://rubd.github.io/Giotto_site/) (Dries et al. 2021), SpaGCN (https://github.com/jianhuupenn/SpaGCN) and stlearn (https://github.com/BiomedicalMachineLearning/stLearn). For Seurat and Giotto, we adjusted the resolution in the Louvain algorithm for a better K estimation (same or close to the real K). All other parameters in all the competing methods are kept in the default setting or following to the settings in the official pipelines. It is noted that the latent dimension of SpaGCN is higher than the feature dimension of osmFISH data. So SpaGCN cannot be used to analyze osmFISH data. For consistency, H&E images are not used for all the methods.

Statistical test

The differences between the clustering performance of DSSC and the competing methods are tested by the one-sided paired t-test.

Software availability

Source code of DSSC is available at GitHub (https://github.com/xianglin226/DSSC) and as Supplemental Code.

Competing interest statement

This research includes no competing interests.

Acknowledgments

This work was supported by grant R15HG012087 (Z.W.) from the National Institutes of Health and grant 1659472 (Z.W.) from the National Science Foundation.

Contributions

Z.W. conceived and supervised the project. X.L. designed the method and conducted the experiments. X.L., L.G., A.A. and N.W. wrote the manuscript. Z.W. revised the manuscript. X.L, L.G, N.W. and A.A. conducted the experiments of the competing methods. All authors contributed to and approved the manuscript.

Reference

Alexander S, Joydeep G. 2003. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J Mach Learn Res* **3**: 583-617.

Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* **2008**: P10008.

Butler A, Hoffman P, Smibert P, Papalexi E, Satija R. 2018. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature biotechnology* **36**: 411-420.

Codeluppi S, Borm LE, Zeisel A, La Manno G, van Lunteren JA, Svensson CI, Linnarsson S. 2018. Spatial organization of the somatosensory cortex revealed by osmFISH. *Nature methods* **15**: 932-935.

Dries R, Zhu Q, Dong R, Eng C-HL, Li H, Liu K, Fu Y, Zhao T, Sarkar A, Bao F. 2021. Giotto: a toolbox for integrative analysis and visualization of spatial expression data. *Genome biology* **22**: 1-31.

Efremova M, Vento-Tormo M, Teichmann SA, Vento-Tormo R. 2020. CellPhoneDB: inferring cell–cell communication from combined expression of multi-subunit ligand–receptor complexes. *Nature protocols* **15**: 1484-1506.

Eraslan G, Simon LM, Mircea M, Mueller NS, Theis FJ. 2019. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature communications* **10**: 1-14.

Femino AM, Fay FS, Fogarty K, Singer RH. 1998. Visualization of single RNA transcripts in situ. *Science* **280**: 585-590.

Guillozet-Bongaarts A, Hyde T, Dalley R, Hawrylycz M, Henry A, Hof P, Hohmann J, Jones A, Kuan C, Royall J. 2014. Altered gene expression in the dorsolateral prefrontal cortex of individuals with schizophrenia. *Molecular psychiatry* **19**: 478-485.

Hinton GE, Salakhutdinov RR. 2006. Reducing the dimensionality of data with neural networks. *science* **313**: 504-507.

Hu J, Li X, Coleman K, Schroeder A, Ma N, Irwin DJ, Lee EB, Shinohara RT, Li M. 2021. SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nature methods* **18**: 1342-1351.

Hubert L, Arabie P. 1985. Comparing partitions. *Journal of classification* **2**: 193-218.

Kaur G, Levy E. 2012. Cystatin C in Alzheimer's disease. *Frontiers in molecular neuroscience* **5**: 79.

Kingma DP, Ba J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:14126980*.

Kiselev VY, Andrews TS, Hemberg M. 2019. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nature Reviews Genetics* **20**: 273-282.

Kiselev VY, Kirschner K, Schaub MT, Andrews T, Yiu A, Chandra T, Natarajan KN, Reik W, Barahona M, Green AR. 2017. SC3: consensus clustering of single-cell RNA-seq data. *Nature methods* **14**: 483-486.

Koch L. 2018. Altered splicing in Alzheimer transcriptomes. *Nature Reviews Genetics* **19**: 738-739.

Kolodziejczyk AA, Kim JK, Svensson V, Marioni JC, Teichmann SA. 2015. The technology and biology of single-cell RNA sequencing. *Molecular cell* **58**: 610-620.

Kuhn HW. 1955. The Hungarian method for the assignment problem. *Naval research logistics quarterly* **2**: 83-97.

Kumar S, Reddy PH. 2020. The role of synaptic microRNAs in Alzheimer's disease. *Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease* **1866**: 165937.

Kwon S. 2013. Single-molecule fluorescence in situ hybridization: quantitative imaging of single RNA molecules. *BMB reports* **46**: 65.

Larsson L, Frisén J, Lundeberg J. 2021. Spatially resolved transcriptomics adds a new dimension to genomics. *Nature Methods* **18**: 15-18.

Liao J, Lu X, Shao X, Zhu L, Fan X. 2021. Uncovering an organ's molecular architecture at single-cell resolution by spatially resolved transcriptomics. *Trends in Biotechnology* **39**: 43-58.

Lipiec MA, Bem J, Kozinski K, Chakraborty C, Urban-Ciecko J, Zajkowski T, Dabrowski M, Szewczyk LM, Toval A, Ferran JL. 2020. TCF7L2 regulates postmitotic differentiation programmes and excitability patterns in the thalamus. *Development* **147**: dev190181.

Liu Q, Chen S, Jiang R, Wong WH. 2021a. Simultaneous deep generative modelling and clustering of single-cell genomic data. *Nature machine intelligence* **3**: 536-544.

Liu Q, Xu J, Jiang R, Wong WH. 2021b. Density estimation using deep generative neural networks. *Proceedings of the National Academy of Sciences* **118**: e2101344118.

Longo SK, Guo MG, Ji AL, Khavari PA. 2021. Integrating single-cell and spatial transcriptomics to elucidate intercellular tissue dynamics. *Nature Reviews Genetics*: 1-18.

Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. 2018. Deep generative modeling for single-cell transcriptomics. *Nature methods* **15**: 1053-1058.
Lubeck E, Cai L. 2012. Single-cell systems biology by super-resolution imaging and combinatorial labeling. *Nature methods* **9**: 743-748.
Maaten Lvd, Hinton G. 2008. Visualizing data using t-SNE. *Journal of machine learning research* **9**: 2579-2605.
Maynard KR, Collado-Torres L, Weber LM, Uytingco C, Barry BK, Williams SR, Catallini JL, Tran MN, Besich Z, Tippani M. 2021. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nature neuroscience* **24**: 425-436.
Miller BF, Bambah-Mukku D, Dulac C, Zhuang X, Fan J. 2021. Characterizing spatial gene expression heterogeneity in spatially resolved single-cell transcriptomics data with nonuniform cellular densities. *Genome Research*: gr. 271288.271120.
Moncada R, Barkley D, Wagner F, Chiodin M, Devlin JC, Baron M, Hajdu CH, Simeone DM, Yanai I. 2020. Integrating microarray-based spatial transcriptomics and single-cell RNA-seq reveals tissue architecture in pancreatic ductal adenocarcinomas. *Nature biotechnology* **38**: 333-342.
Moran PA. 1950. A test for the serial independence of residuals. *Biometrika* **37**: 178-181.
Nair V, Hinton GE. 2010. Rectified linear units improve restricted boltzmann machines. In *ICML*.
Pardilla-Delgado E, Torrico-Teave H, Sanchez JS, Ramirez-Gomez LA, Baena A, Bocanegra Y, Vila-Castelar C, Fox-Fuller JT, Guzman-Velez E, Martinez J. 2021. Associations between subregional thalamic volume and brain pathology in autosomal dominant Alzheimer's disease. *Brain communications* **3**: fcab101.
Pardo B, Spangler A, Weber LM, Page SC, Hicks SC, Jaffe AE, Martinowich K, Maynard KR, Collado-Torres L. 2022. spatialLIBD: an R/Bioconductor package to visualize spatially-resolved transcriptomics data. *BMC Genomics* **23**: 1-5.
Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A. 2017. Automatic differentiation in pytorch.
Pham D, Tan X, Xu J, Grice LF, Lam PY, Raghubar A, Vukovic J, Ruitenberg MJ, Nguyen Q. 2020. stLearn: integrating spatial location, tissue morphology and gene expression to find cell types, cell-cell interactions and spatial trajectories within undissociated tissues. *BioRxiv*.
Reddi S, Kale S, Kumar S. 2018. On the convergence of adam and be-365 yond. In *International conference on learning representations Retrieved from*, Vol 366.
Shah S, Lubeck E, Zhou W, Cai L. 2016. In situ transcription profiling of single cells reveals spatial organization of cells in the mouse hippocampus. *Neuron* **92**: 342-357.
Shapiro E, Biezuner T, Linnarsson S. 2013. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nature Reviews Genetics* **14**: 618-630.
Skelly DA, Squiers GT, McLellan MA, Bolisetty MT, Robson P, Rosenthal NA, Pinto AR. 2018. Single-cell transcriptional profiling reveals cellular diversity and intercommunication in the mouse heart. *Cell reports* **22**: 600-610.
Stoltzfus CR, Filipek J, Gern BH, Olin BE, Leal JM, Wu Y, Lyons-Cohen MR, Huang JY, Paz-Stoltzfus CL, Plumlee CR. 2020. CytoMAP: a spatial analysis toolbox

reveals features of myeloid cell organization in lymphoid tissues. *Cell reports* **31**: 107523.
Stuart T, Butler A, Hoffman P, Hafemeister C, Papalexi E, Mauck III WM, Hao Y, Stoeckius M, Smibert P, Satija R. 2019. Comprehensive integration of single-cell data. *Cell* **177**: 1888-1902. e1821.
Svensson V, Teichmann SA, Stegle O. 2018. SpatialDE: identification of spatially variable genes. *Nature methods* **15**: 343-346.
Takahama S, Nakaya N, Tomarev SI. 2014. Olfactomedin 1 may suppress APP cleavage through its interaction with BACE1. *Investigative Ophthalmology & Visual Science* **55**: 2959-2959.
Tian T, Wan J, Song Q, Wei Z. 2019. Clustering single-cell RNA-seq data with a model-based deep learning approach. *Nature Machine Intelligence* **1**: 191-198.
Tian T, Zhang J, Lin X, Wei Z, Hakonarson H. 2021. Model-based deep embedding for constrained clustering analysis of single cell RNA-seq data. *Nature communications* **12**: 1-12.
van de Mortel LA, Thomas RM, van Wingen GA, Initiative AsDN. 2021. Grey matter loss at different stages of cognitive decline: a role for the thalamus in developing AlzheimerБ—≥s disease. *Journal of Alzheimer's Disease* **83**: 705-720.
Vincent P, Larochelle H, Bengio Y, Manzagol P-A. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pp. 1096-1103.
Wang S, Karikomi M, MacLean AL, Nie Q. 2019. Cell lineage and communication network inference via optimization for single-cell transcriptomics. *Nucleic acids research* **47**: e66-e66.
Wolf FA, Angerer P, Theis FJ. 2018. SCANPY: large-scale single-cell gene expression data analysis. *Genome biology* **19**: 15.
Xia C, Fan J, Emanuel G, Hao J, Zhuang X. 2019. Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression. *Proceedings of the National Academy of Sciences* **116**: 19490-19499.
Xie J, Girshick R, Farhadi A. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pp. 478-487.
Zappia L, Phipson B, Oshlack A. 2017. Splatter: simulation of single-cell RNA sequencing data. *Genome biology* **18**: 1-15.
Zeisel A, Munoz-Manchado AB, Codeluppi S, Lonnerberg P, La Manno G, Jureus A, Marques S, Munguba H, He L, Betsholtz C. 2015. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**: 1138-1142.
Zhao E, Stone MR, Ren X, Guenthoer J, Smythe KS, Pulliam T, Williams SR, Uytingco CR, Taylor SE, Nghiem P. 2021. Spatial transcriptomics at subspot resolution with BayesSpace. *Nature Biotechnology*: 1-10.
Zhu Q, Shah S, Dries R, Cai L, Yuan G-C. 2018. Identification of spatially associated subpopulations by combining scRNAseq and sequential fluorescence in situ hybridization data. *Nature biotechnology* **36**: 1183-1190.
Zhuang X. 2021. Spatially resolved single-cell genomics and transcriptomics by imaging. *Nature methods* **18**: 18-22.

Zou Y-m, Lu D, Liu L-p, Zhang H-h, Zhou Y-y. 2016. Olfactory dysfunction in Alzheimer's disease. *Neuropsychiatric disease and treatment* **12**: 869.

Figure Legends

Figure 1. DSSC model architecture. The inputs of DSSC are the gene expression matrix and the cell coordinates. The outputs of DSSC are the low-dimension latent space (32D) and the predicted labels. Briefly, DSSC learns a low-dimensional representation of the gene expression matrix while simultaneously leveraging the prior knowledge from the spatial coordinates of cells/spots and the marker genes. Clustering is performed on latent space. Constraint loss, reconstruction loss, and clustering loss are optimized simultaneously. ML loss and CL loss are optimized alternately. Notations: BN stands for the batch normalization; ELU stands for the ELU activation; ML indicates the must-links constraints; CL indicates the cannot-link constraints; ZINB means the zero-inflated negative binominal.

Figure 2. Simulation results from the spatial organization 1 (A, from osmFISH data) and 2 (B, from spatialLIBD sample 151507). True labels with 10%, 15%, and 20% perturbed coordinates are shown on the physical spaces (left). The corresponding clustering results are shown in the bar plots (right).

Figure 3. Results of osmFISH dataset. A. predicted labels; B. clustering performance; and C. marker genes used for DSSC.

Figure 4. Results of spatialLIBD datasets. A. visualization of the predicted label for sample 151673; B. the clustering performance of the 12 samples; and C. the marker gene used in this experiment.

Figure 5. Results of 10xMBAD datasets. A. clustering performance (without true labels); B. a cartoon of brain showing the position of thalamus (from www.flintrehab.com) and the expression of a marker gene, Tcf7l2, for thalamus in a WT and an AD sample; C. predicted labels for a wild type sample and an Alzheimer's disease sample from DSSC, BayesSpace, and SpaGCN; the black arrows indicate the thalamus regions; D. volcano plot from the differential expression analysis (DE) between the cells in thalamus from the wild type and the Alzheimer's disease samples; E. KEGG pathway analysis from the DE results in panel D. The pathway of Alzheimer's disease is highlighted by the red box.

Figure 6. Parameter tuning on the 12 spatialLIBD datasets (A) and running time test on the simulated data with incremental cell numbers (B).

Encoder(GAT)

Decoder(Dense)

Normalized count

Input 1
N x M

Feature Selection

(Optional)

Noise

GAT

BN

ELU

128

GAT

32

Dense

BN

ELU

128

Dense

EXP

ZINB loss

Dropout

Dispersion

Mean

KNN Graph
N x N

Clustering loss

ML
loss

Maker 1

CL
loss

Maker 2

.....

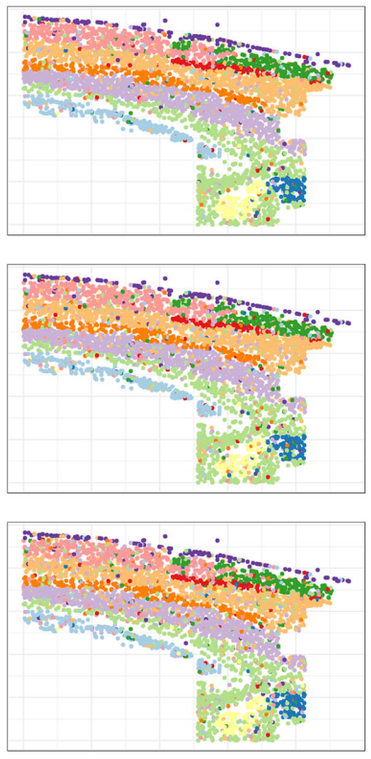
Spatial coordinates

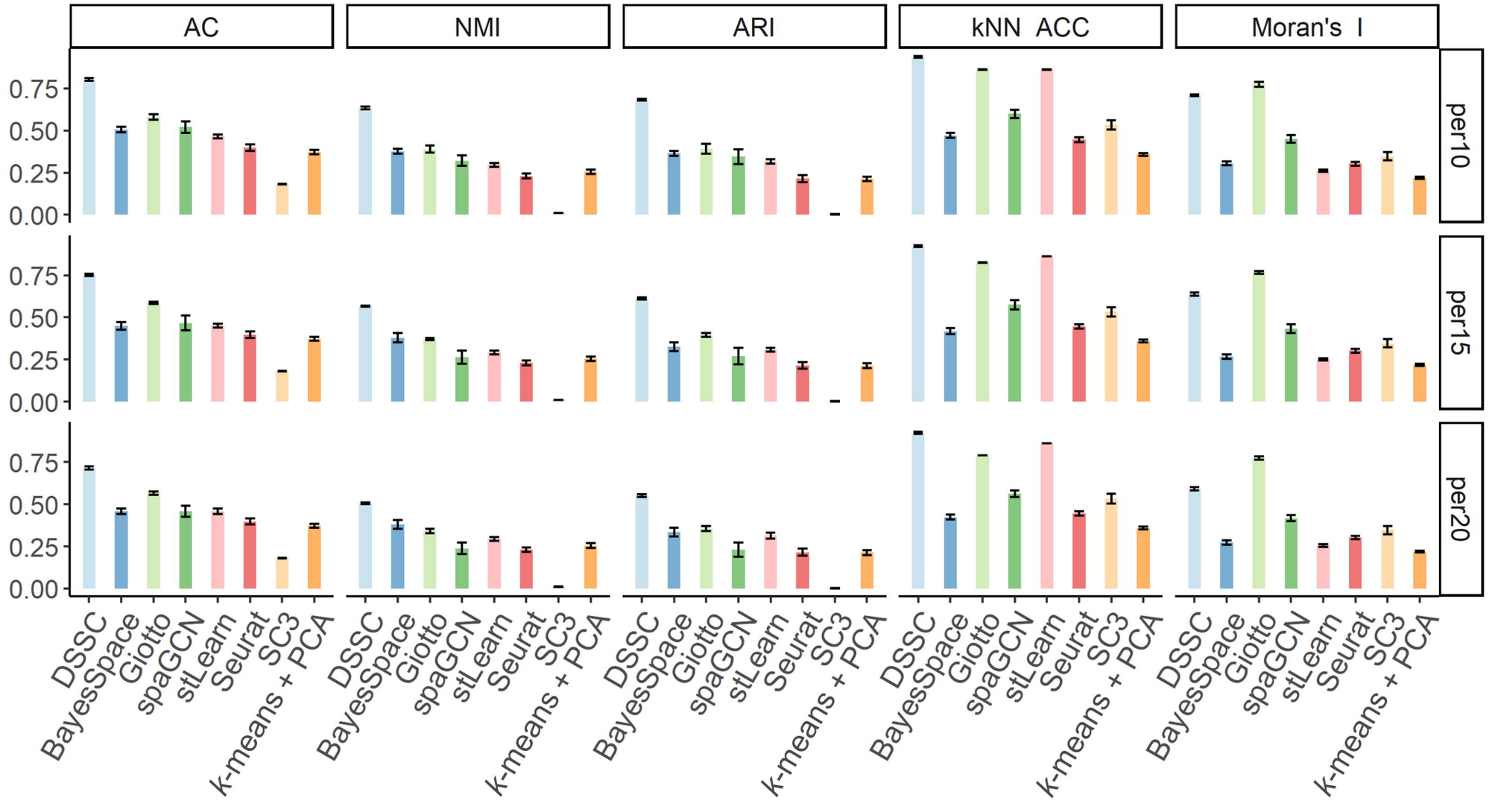
Input 2

Spatial-based
smoothing

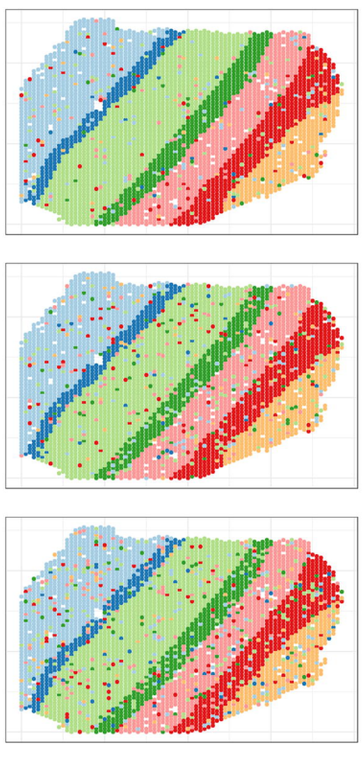
**Total Loss =
ZINB Loss +
Clustering Loss +
ML Loss +
CL Loss**

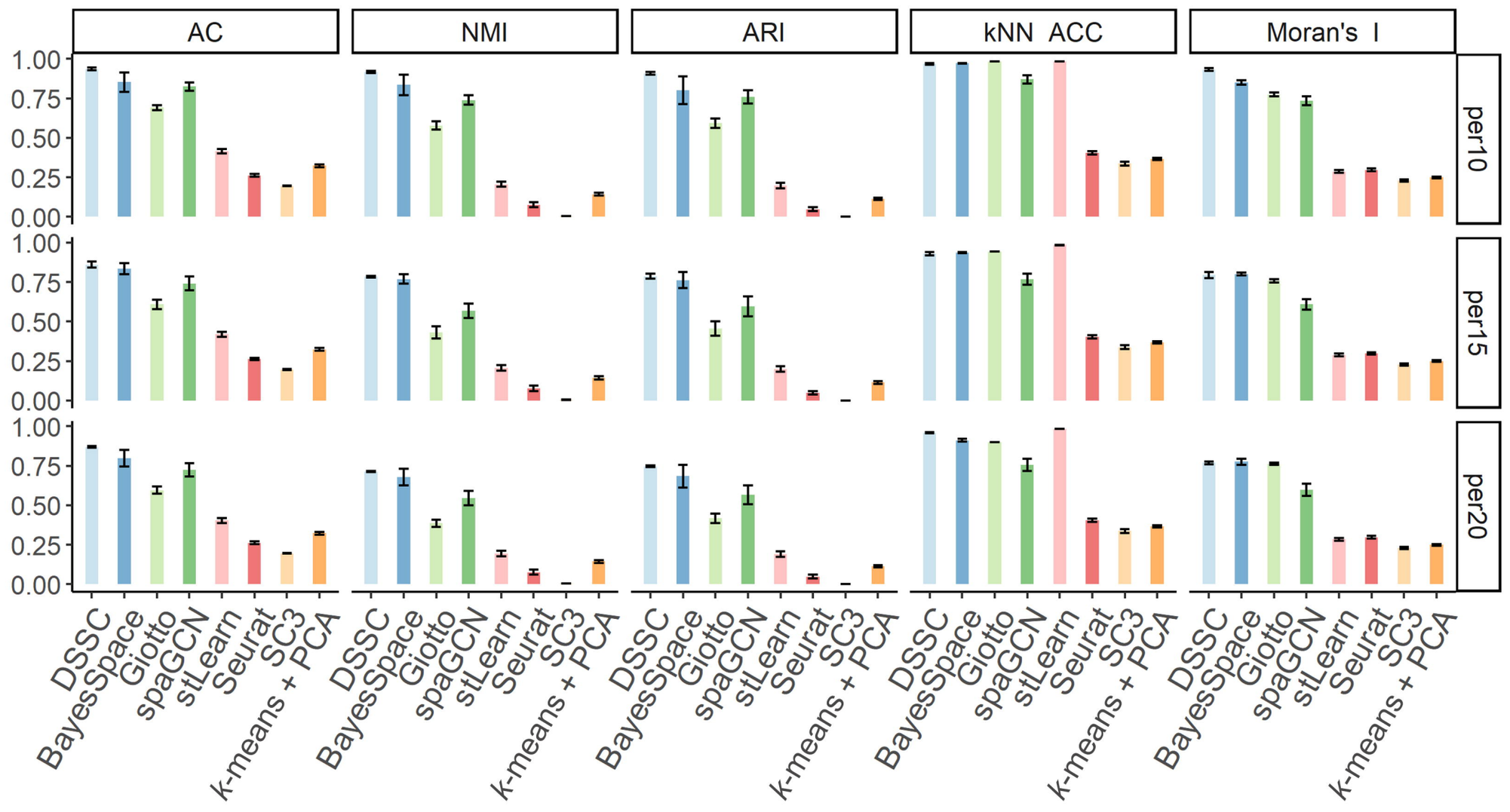
A



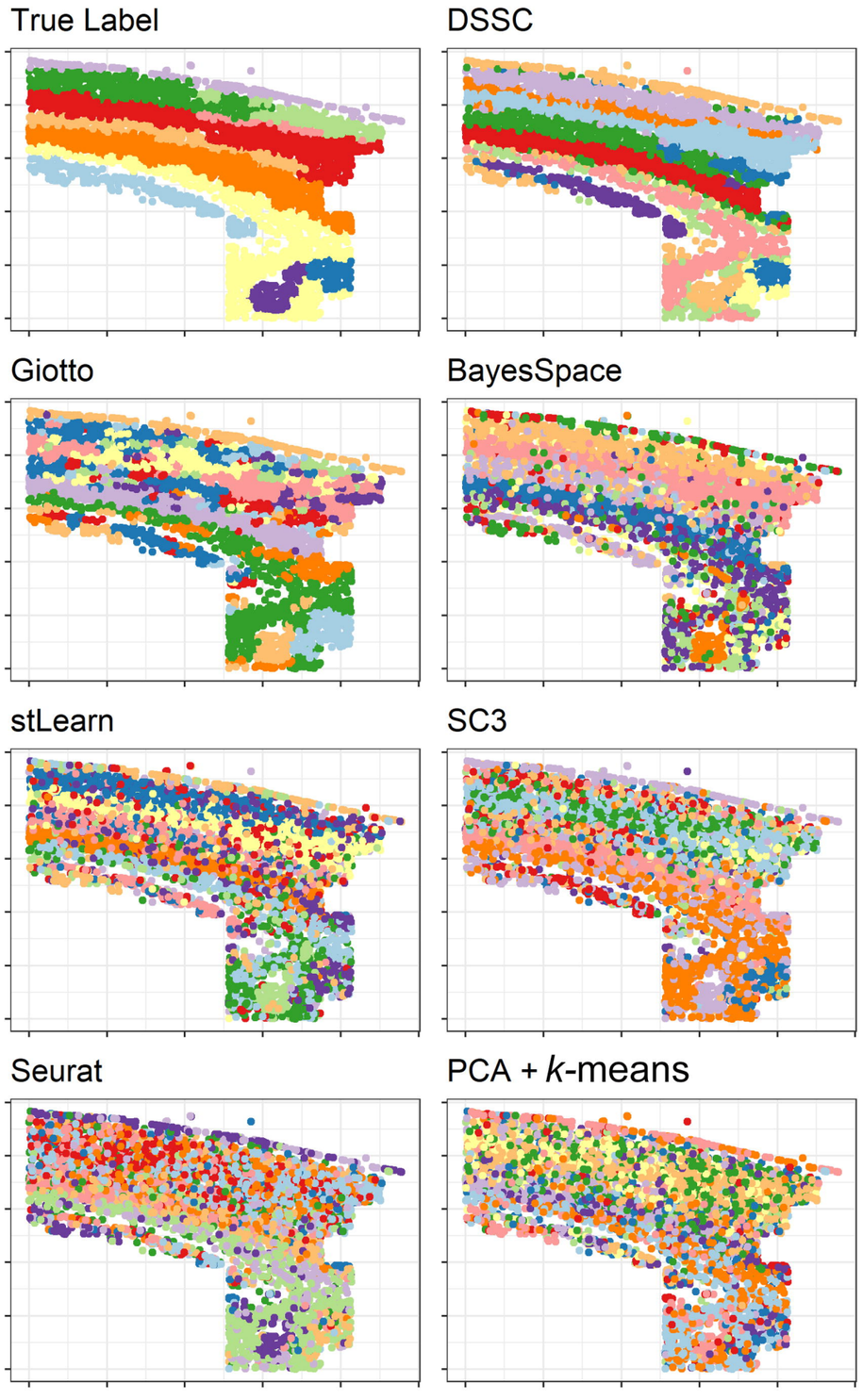

AC
NMI
ARI
kNN ACC
Moran's I
per10
per15
per20
0.00
0.25
0.50
0.75
DSSC
BayesSpace
Giotto
spaGCN
stLearn
Seurat
SC3
k-means + PCA

B



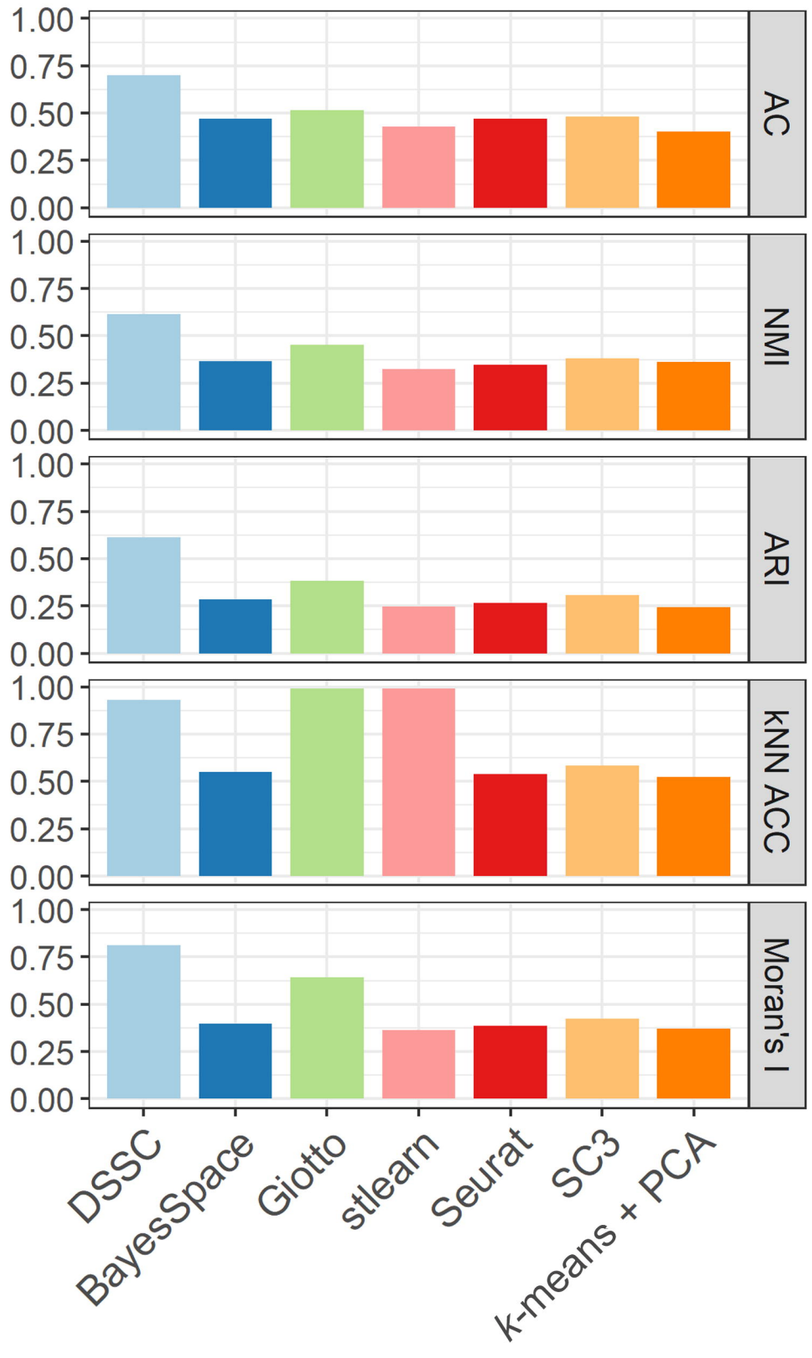

AC
NMI
ARI
kNN ACC
Moran's I
per10
per15
per20
0.00
0.25
0.50
0.75
1.00
DSSC
BayesSpace
Giotto
spaGCN
stLearn
Seurat
SC3
k-means + PCA

A

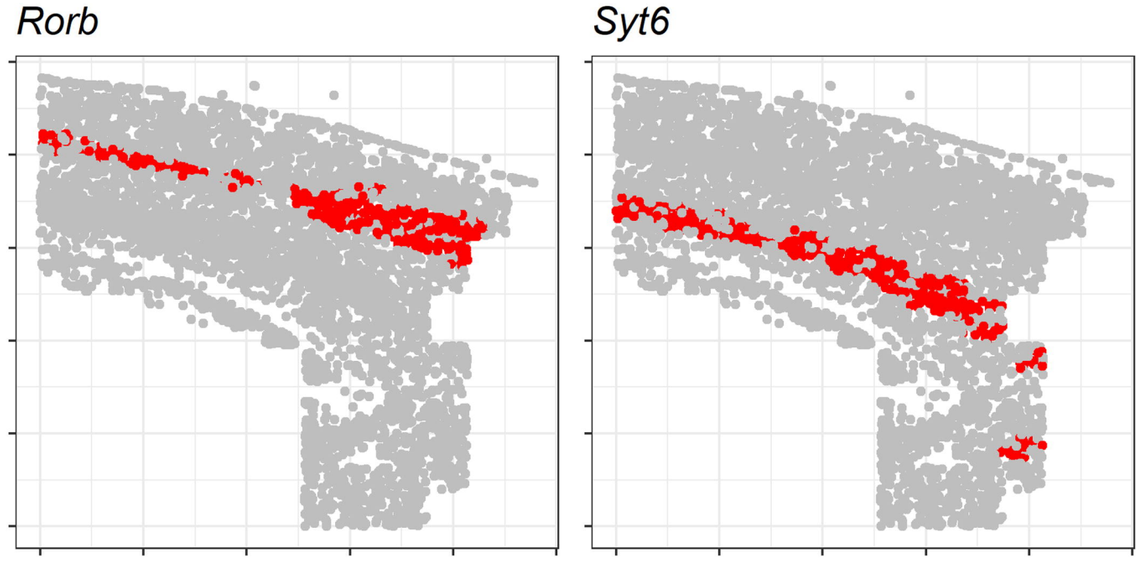


Hippocampus
Internal Capsule Caudoputamen
Layer 2-3 lateral
Layer 2-3 medial
Layer 3-4
Layer 4
Layer 5
Layer 6
Pia Layer 1
Ventricle
White matter

B



C



A

True Label

DSSC

SpaGCN

Giotto

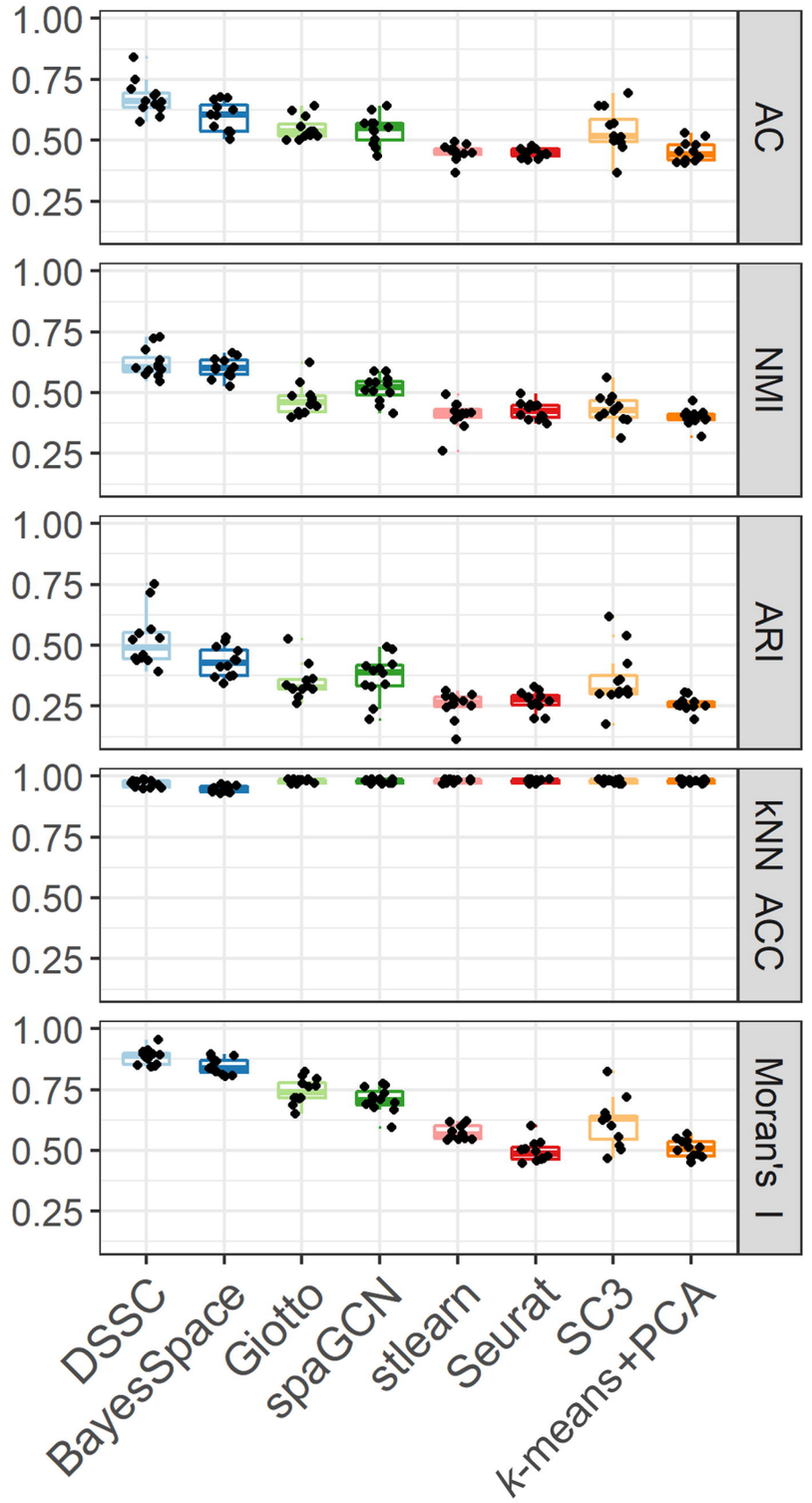
BayesSpace

stLearn

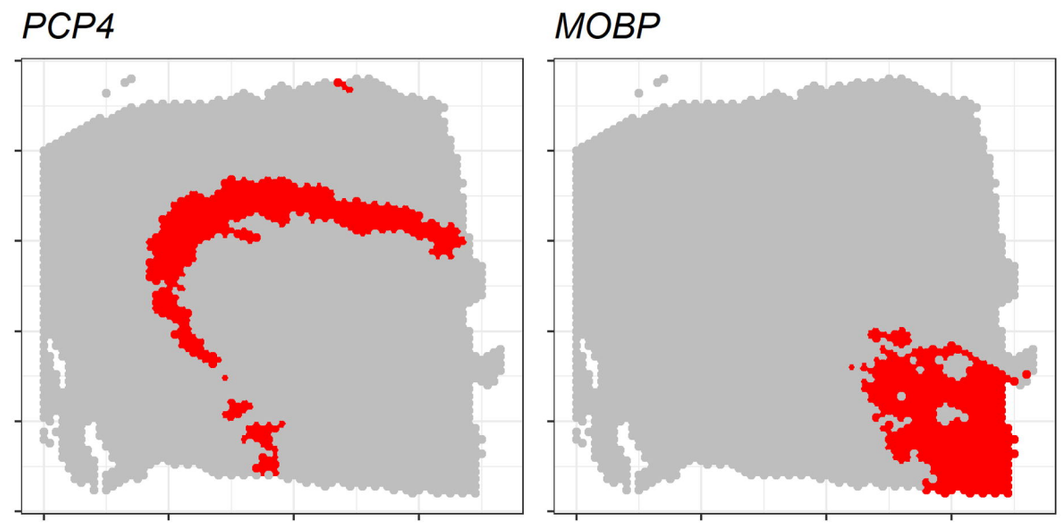
SC3

Seurat

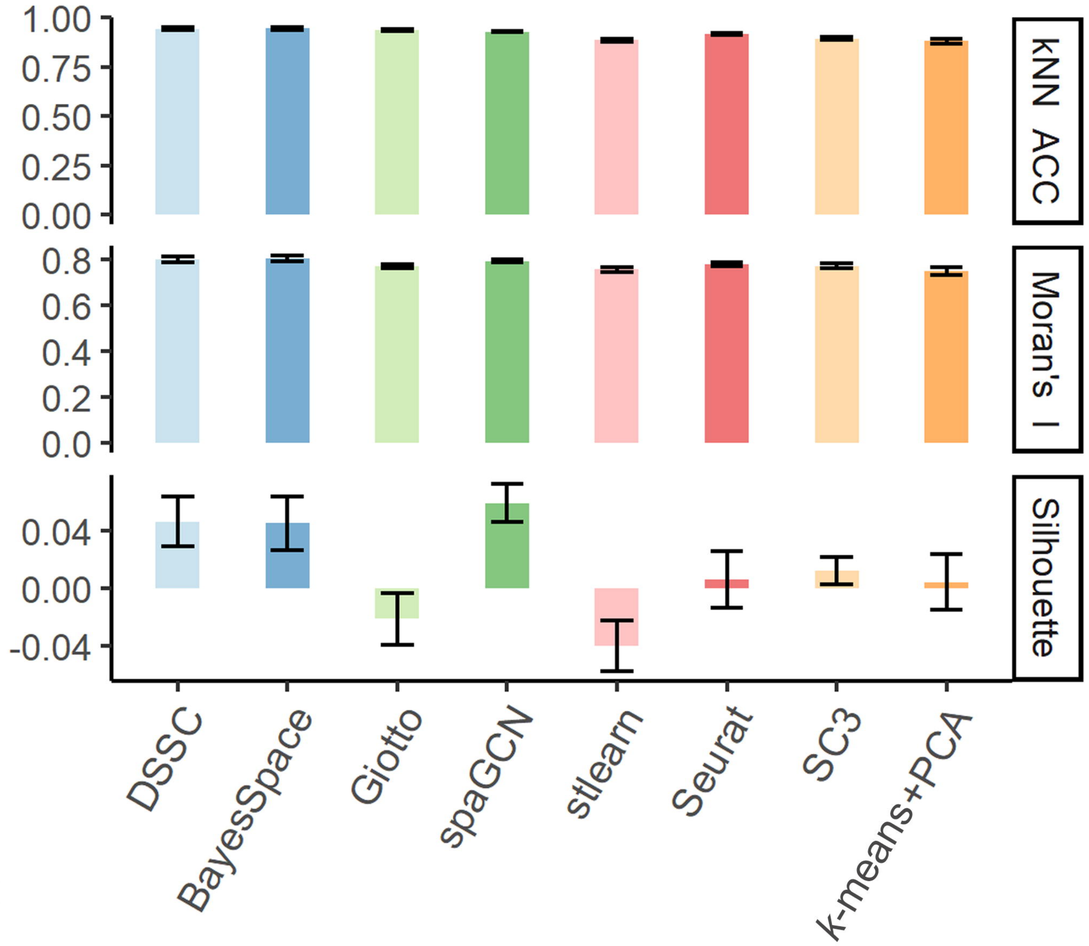
B



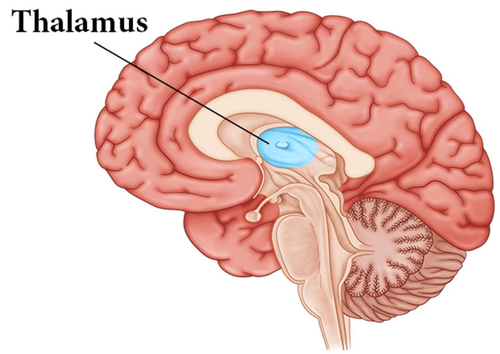
C

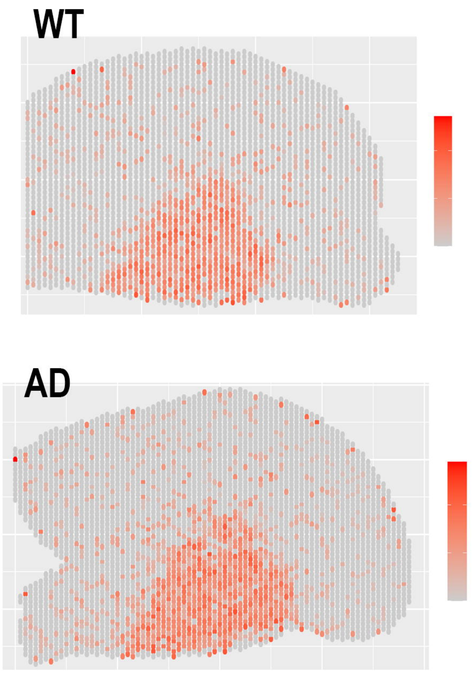


A

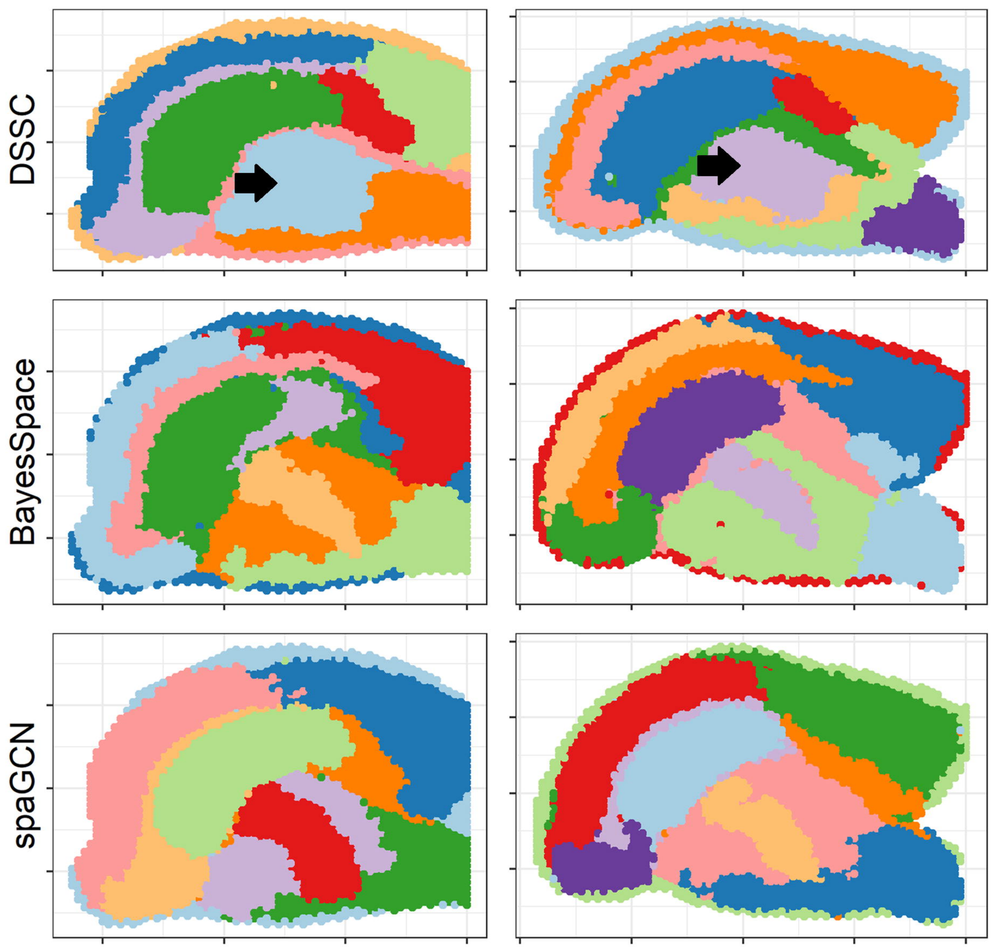


B

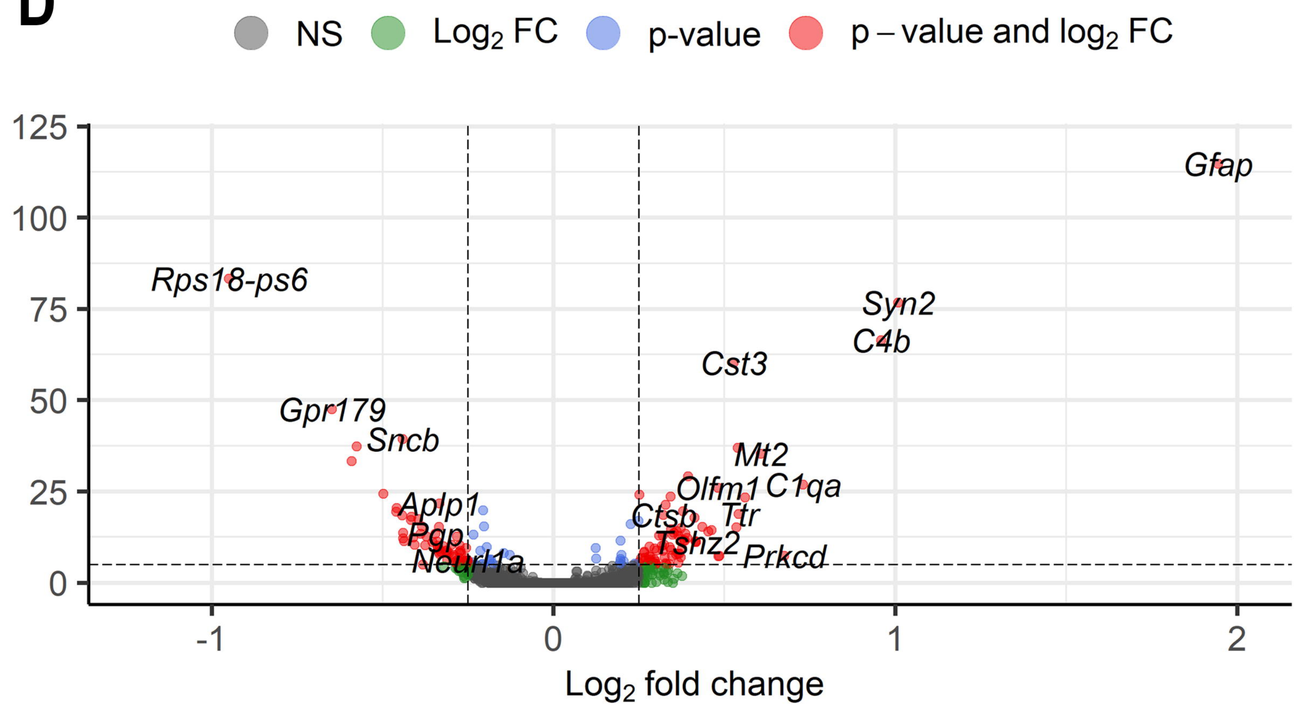




C



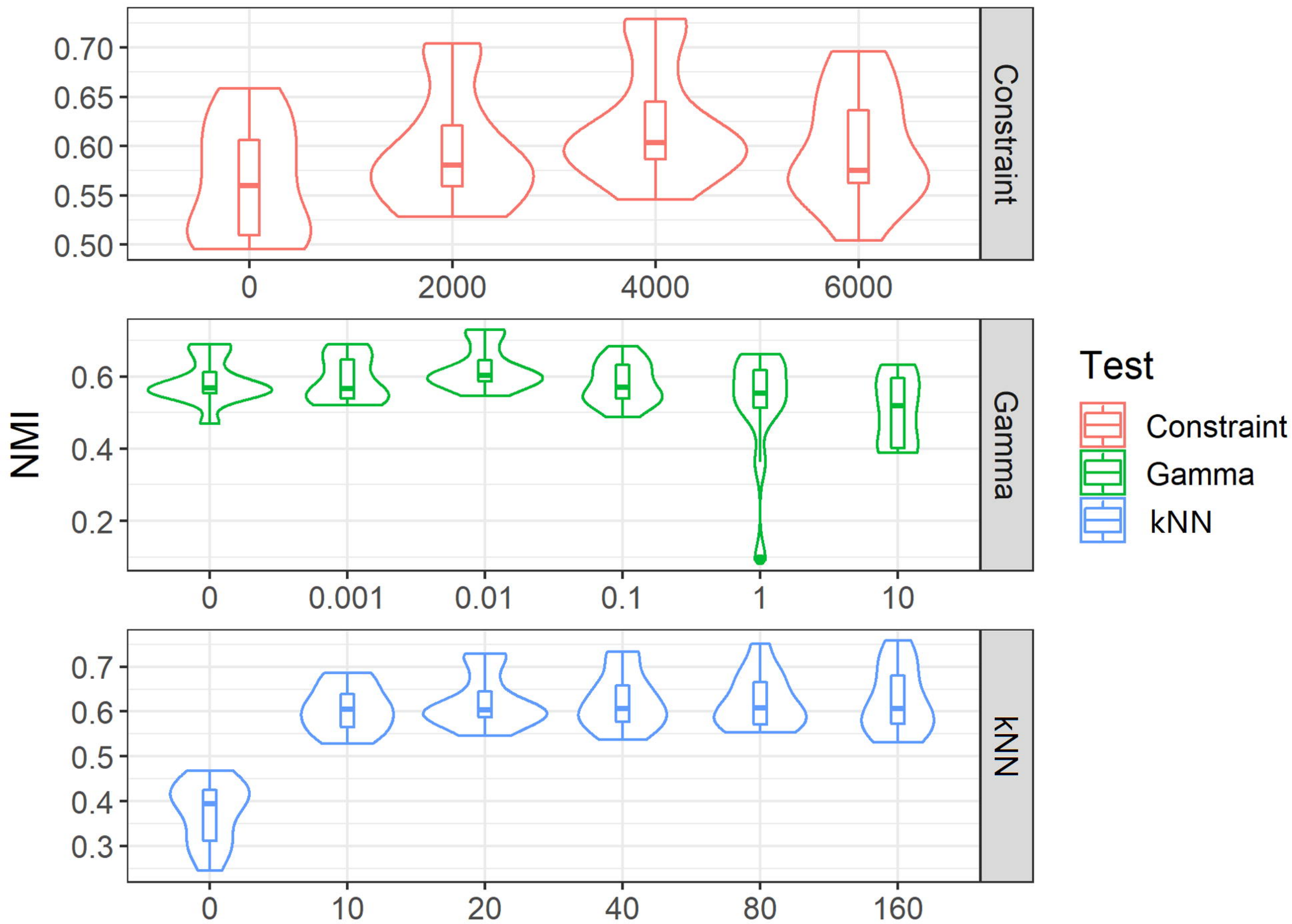
D



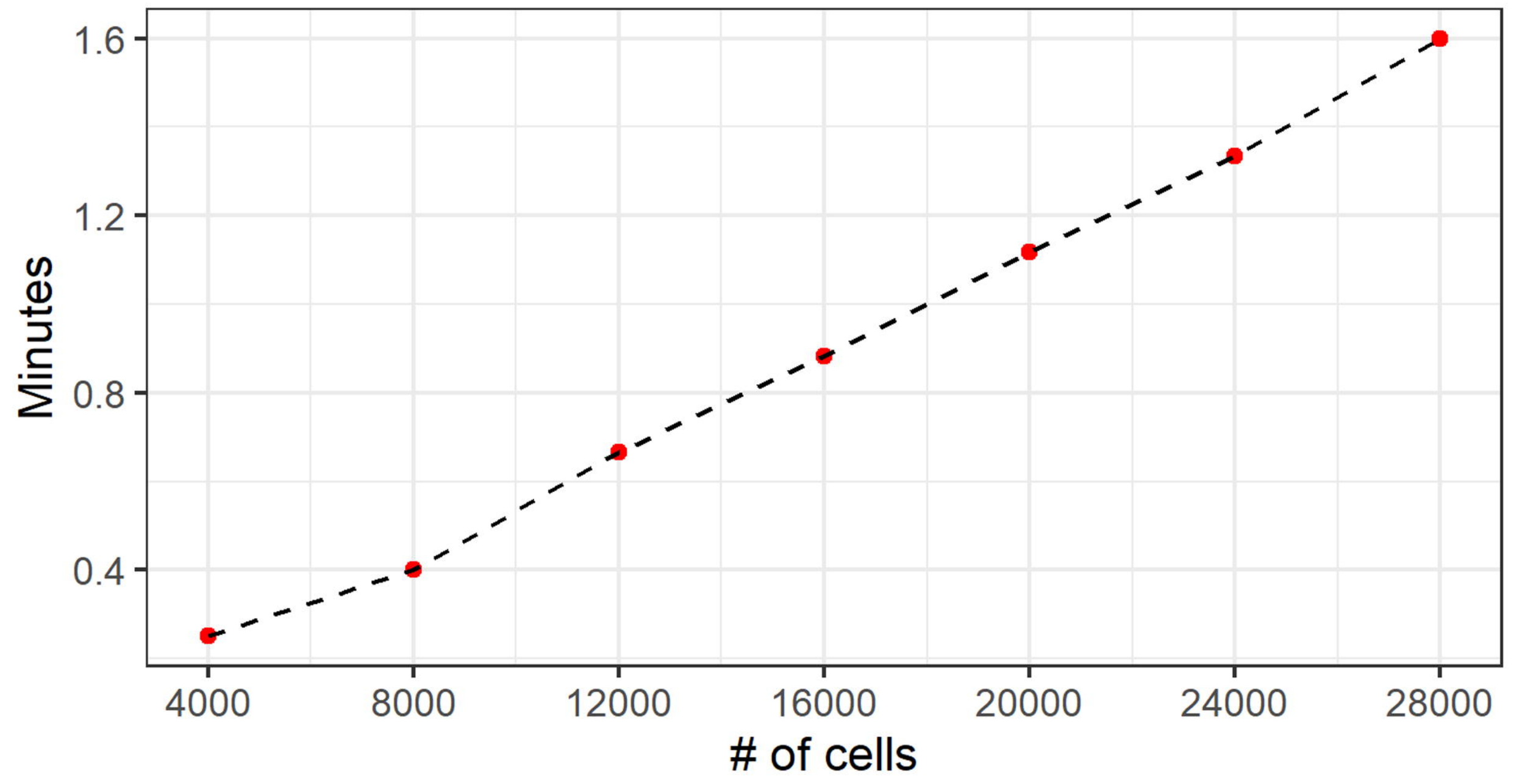
E

Pathway	Gene ranks	NES	pval	padj
KEGG_COMPLEMENT_AND_COAGULATION_CASCADES		2.16	1.2e-07	2.2e-05
KEGG_OLFACTORY_TRANSDUCTION		1.50	4.7e-04	2.9e-02
KEGG_SYSTEMIC_LUPUS_ERYTHEMATOSUS		1.87	4.0e-04	2.9e-02
KEGG_ALLOGRAFT_REJECTION		1.80	1.4e-03	4.8e-02
KEGG_ALZHEIMERS_DISEASE		1.56	1.6e-03	4.8e-02
KEGG_ANTIGEN_PROCESSING_AND_PRESENTATION		1.68	1.2e-03	4.8e-02

A


Constraint
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
NMI
Gamma
0.6
0.4
0.2
0
0.001
0.01
0.1
1
10
kNN
0.7
0.6
0.5
0.4
0.3
0
10
20
40
80
160
Test
Constraint
Gamma
kNN

B


Minutes
1.6
1.2
0.8
0.4
4000
8000
12000
16000
20000
24000
28000
# of cells