CoSTA: Unsupervised Convolutional Neural Network Learning for Spatial Transcriptomics Analysis

Yang Xu[1] and Rachel Patton McCord[2]*

[1] UT-ORNL Graduate School of Genome Science and Technology, University of Tennessee, Knoxville, TN

[2] Biochemistry & Cellular and Molecular Biology Department, University of Tennessee, Knoxville, TN

* Correspondence: rmccord@utk.edu (R.P.M.)

Abstract

The rise of spatial transcriptomics technologies is leading to new insights about how gene regulation happens in a spatial context. Here, we present CoSTA- a novel approach to learn spatial representation from gene expression matrices via convolutional neural network (ConvNet) clustering. We reanalyze published spatial transcriptomics data and demonstrate that our method learns spatial relationships of genes by distinguishing them from noise.

Keywords

Spatial transcriptomics, gene clustering, convolutional neural network

Background

Spatial transcriptomics has recently gained extensive attention from the scientific community. Different technologies have enabled high resolution measurements of how gene regulation is spatially organized across a tissue or thousands of single cells.[1] However, current analysis pipelines often lose spatial information by treating each pixel in an expression matrix as an independent feature. For example, the new seqFISH+ technique can fluorescently detect 10,000 mRNAs in situ at single cell resolution, and there are often groups of cells that have correlated gene expression with their neighbors to make up larger structures. However, the original report analyzed these expression patterns using PCA and hierarchical clustering, treating each cell as an independent feature, rather than preserving spatial positions of cell neighbors.[2] Slide-seq similarly produces high-throughput spatially resolved transcription information, using sequencing rather than fluorescence. Previous analyses of Slide-seq data first identified spatially non-random gene expression, but then looked for genes expressed in similar patterns using overlap analysis rather than preserving spatial features.[3] So far, the existing algorithms that are used for analysis of spatial transcriptomics are based on statistical modeling and primarily propose to distinguish spatially expressing or variable (SE or SV) genes from random spatial expression noise. For example, both SpatialDE and SPARK analysis approaches estimate how significant the spatial pattern of a gene is.[4, 5] SpatialDE further builds in an unsupervised pattern detection algorithm to cluster significant SE genes into different groups which should have certain spatial patterns in collective. SPARK, in contrast, was designed only for finding SE genes. To examine spatial relationships between genes, this method still relies on hierarchical clustering with individual pixels as features. Therefore, even with a distinct power to identify highly significant SE genes, the latter part of the SPARK analysis decouples the expression from its original spatial

context. Thus far, existing spatial transcriptomics analyses involve either multi-step complex feature engineering for spatial quantification or human-imposed rigid or statistical modeling-based screening of candidate SE genes. In this work, we propose an approach inspired by computer vision and image classification to find relationships between spatial expression patterns of different genes while preserving the full spatial context (Fig. 1).

Results

Convolutional neural networks have demonstrated a wide range of applications in computer vision, including image classification and object recognition. A few groups have proposed different approaches to use convolutional neural networks (ConvNet) in unsupervised learning.[6-8] Here, we adopt an unsupervised **Co**nvNet learning strategy for **S**patial **T**ranscriptomics **A**nalysis (CoSTA) (Fig. 1a and see Methods for detailed description). Though there are many unsupervised learning strategies, we chose to apply the workflow of DeepCluster, because it is straightforward and easy to implement.[6] Our CoSTA approach consists of two main parts: clustering by Gaussian mixture model (GMM) and weight updating as commonly performed in training neural networks. Our inputs are sets of images, where each image represents the spatial expression pattern of one gene and all images represent the same biological space. These can also be thought of as a set of gene expression matrices where the matrix records the expression of a given gene at a given position in the space. We first initialize a ConvNet randomly and then forward these gene expression matrices through the ConvNet. The ConvNet for our analyses consists of three convolutional layers, and each convolutional layer is followed by a batch normalization layer and a max pooling layer. Finally, we flatten the matrix output from the last max pooling layer into a vector that captures the spatial features of the gene expression data. The size of this vector will vary depending on the image size from a given spatial transcriptomics technique. We then apply L2-normalization across features and reduce dimensionality by UMAP before we perform GMM clustering of genes. UMAP can preserve global and local structures during dimension reduction and previously showed better performance in image clustering than other dimension-reduction methods, for example Isomap and t-SNE.[7, 9] The purpose of clustering is generating labels, so that we can update the ConvNet like most common supervised neural network training. When the ConvNet is randomly initialized, features extracted by this ConvNet are weak. However, using them to generate labels can still guide the ConvNet to learn more discriminative features. Indeed, Caron et al. showed DeepCluster can learn from weak signal to bootstrap the discriminative power of a ConvNet.[6] Instead of giving each gene an arbitrary cluster label, we assign an auxiliary target distribution as a soft assignment. This approach will emphasize genes with high confidence in the clustering task and discount noisy labels due to random initialization of ConvNet. Doing this can also lead to more stable target values for training the neural network.[8] Finally, we can use the soft assignments we generated from clustering to train the ConvNet. We add a fully connected layer after the ConvNet. This fully connected layer produces probabilities for each gene. Thus, we can optimize the model by minimizing bi-tempered logistic loss based on Bregman Divergences between the generated soft assignments and the probabilities from the fully connected layer.[10] In summary, the CoSTA approach uses a ConvNet clustering architecture which repeats 1) generating features by ConvNet, 2) generating soft assignments by GMM clustering, and 3) using soft assignments to update ConvNet. Once we finish training, we only retain the trained ConvNet for the purpose of feature extraction. Further details about the rationale of this learning architecture can be found in Methods.

To demonstrate the spatial information lost by overlap analysis and why a spatial representation approach such as CoSTA is needed, we present a simplified biologically-inspired example (Fig. 1b). These cartoons represent a feature commonly observed in biological tissue sections: a tightly connected epithelial layer of cells (rectangles) adjacent to a collection of stromal cells (circles). In this example, the spatial expression patterns of three genes are shown. Comparing gene expression patterns by overlap only, we observe that Gene 1 and 2 have the same amount of overlap as Gene 1 and 3 (40%). Thus, an overlap approach to measure gene pattern similarity, like the one used in previous Slide-seq analysis [3], would report that Gene 1 is equally similar to both Gene 2 and Gene 3. However, biologically, it is relevant that Gene 1 and Gene 2 are expressed primarily in the epithelial layer while Gene 3 is expressed in the stroma. This biological difference is not detected by strict overlap, but instead requires a spatial representation that would detect the vertical stripe of epithelial layer expression as a salient pattern. Therefore, we are motivated to use our ConvNet clustering based CoSTA approach to prioritize similar shape more than overlap for biological cases where layers of cells and the overall patterns of groups of cells matter more than independent individual cell identities.[11]

As a first test of CoSTA's ability to detect correlated spatial patterns in the absence of exact overlap, we use the MNIST handwritten digit image data.[12] For example, when the aim is to find which digits have correlated handwritten patterns to the digit 3, CoSTA identifies only other instances of digit 3 as correlated. In contrast, overlap analysis will include all other digits as correlated digits of 3 (Fig. S1). Meanwhile, we notice CoSTA finds a smaller number of correlated digit 3 while overlap analysis returns more correlated digits overall (Fig. S1). As observed again below in biological analyses, this broader but less specific vs. narrower and more specific correlated sets is a general feature of overlap vs. spatial pattern analysis.

Before we apply CoSTA to real spatial transcriptomics data, we simulated 5 synthetic datasets following the simulation method in SPARK.[5] Each dataset is generated based on three real expression patterns from mouse olfactory bulb data replicate 11 (Fig. 1c left panel).[13] We generated 2500 fake spatial expression matrices for each pattern, to mimic data for 7500 total genes, and then simulated noise and variability around the patterns as follows. For each gene, we added non-spatial residual errors onto each spatial coordinate independently based on a normal distribution with mean of zero and variance from 0.2 to 0.6. The variance introduces different levels of noise. We then evaluated whether CoSTA can separate these 3 patterns by assigning right clustering label to each gene. When the model was initialized, the Normalized Mutual Information (NMI) against the true class label ranged from 0.24 to 0.57 (Fig. 1c right panel). As training proceeded, CoSTA learned discriminative features to distinguish the 3 patterns, and CoSTA eventually achieved NMIs from 0.92 to 0.97 against the true class label (Fig. 1c right panel, Table S1). Notably, when the introduced variance was 0.6 and the starting point of NMI was below 0.3, the CoSTA approach using only bi-tempered logistic loss failed this task. We introduced center loss (CL) as an additional loss function to train CoSTA, and CoSTA with center loss was able to separate the 3 patterns and achieved 0.92 NMI against the true class label. To demonstrate that features learned by CoSTA from these synthetic datasets are spatially related, we shuffled these synthetic datasets. Shuffling all the gene matrices exactly the same way keeps the pixelwise overlap information identical while disrupting correlations between neighboring pixels, causing disruption of the spatial pattern. We found that CoSTA cannot distinguish the genes into correct pattern labels as well with shuffled data (NMI = 0.23 to 0.87), demonstrating that CoSTA is detecting spatial features that depend on the positions of neighboring pixels, rather than features that can be captured by a set of single pixels (Fig. S2 and Table S1). We also tested

SpatialDE on these true and shuffled synthetic datasets. SpatialDE performed very well on the true datasets, as expected. However, shuffling the data did not usually change the performance of SpatialDE (Table S1), indicating an important difference between CoSTA and SpatialDE: SpatialDE is more likely to detect patterns of individual pixels while CoSTA emphasizes the spatial positions of these pixels relative to each other and overall shapes of patterns. Overall, the performance of CoSTA with synthetic data demonstrates that CoSTA can learn discriminating spatial features.

To extend the application of CoSTA to real spatial transcriptomics data, we first applied it to reanalyze a MERFISH dataset (see MERFISH Analysis in Methods for complete details).[14] In order to compare with published analyses with the SPARK approach, we focused on the same slice of the mouse hypothalamus (Bregma + 0.11 mm from animal 18).[5] The expression patterns of a set of 155 genes expected to be spatially variable were measured with MERFISH for this slice, along with 5 blank control genes. We first initialized a ConvNet and forwarded the MERFISH spatial gene expression matrices through it to obtain gene feature vectors. Then we clustered the 155 spatially variable genes with the 5 blank genes and with 9 cell type-specific expression patterns defined by the original publication through a combination of MERFISH and scRNA-seq data. We clustered these genes, controls, and cell type patterns into 10 groups and visualized them by UMAP. Without training, SE genes, control genes, and cell types are spread across the 2-dimensional space and boundaries between groups are not distinctively defined (Fig. 2a). Next, we trained the CoSTA model to obtain refined feature vectors. After training, SE genes, control genes and cell types formed distinct groups that have clearer boundaries in the 2D visualization (Fig. 2b) and refined cluster memberships that reproducibly and quantitatively form tighter clusters according to a linear intrinsic dimensionality (LID) estimator (Fig. 2c) [15].

From this MERFISH data, SPARK identified 145 SE genes including one blank control, and SpatialDE found 139 SE genes with one blank control.[5] Because CoSTA is not designed for estimation of spatial relevance but primarily for detection of spatial similarity and spatial relationships between gene expression patterns, we cannot use approaches in SPARK and SpatialDE to call SE genes directly. Therefore, we took advantage of the existence of 9 defined cell type specific expression patterns and tested how genes are retrieved as highly correlated to one of these patterns without retrieving blank controls. CoSTA revealed 133 SE genes that are associated with the different cell type patterns, and none of the blank controls were called associated with a pattern (Table S2). This number is slightly lower than the SE genes identified by SPARK and SpatialDE, but with higher specificity (no blank controls detected). Our result is also more sensitive than Trensceek which only identified 108 SE genes with one blank control.[16] Three genes, *Avpr1a*, *Chat*, and *Nup62cl*, were highlighted by Sun et al., because they were only identified by SPARK.[5] CoSTA is also able to identify the spatial expression patterns of these genes. *Chat* is significantly correlated to the Endothelium pattern. *Avpr1a* is grouped with *Nnat* and *Cd24a* that both have similar spatial pattern to Ependymal, and *Nup62cl* is grouped with *Mbp* and *Opalin* which are correlated to Mature OD (Fig. 2d and Table S2). However, we also note that on visual inspection, the spatial patterns of some of these genes are ambiguous. This is likely why CoSTA associates these genes with other gene patterns, but not directly with the original cell type pattern.

After successfully demonstrating the application of CoSTA to MERFISH data, we next expand our application of CoSTA to Slide-seq data. Slide-seq takes advantage of high-throughput single cell RNA sequencing and barcoding. Therefore, it enables spatial gene expression

measurement for all genes in the genome.[3] As a first demonstration that CoSTA can be applied to this type of high-throughput spatial transcriptomics data, we performed an experiment-mixing test to evaluate whether CoSTA can separate different spatial patterns. Due to the unavailability of a "gold standard" for positive and negative spatial similarity of gene expression, we mixed gene matrices from four different spatial transcriptomics experiments by Slide-seq and tested the ability of CoSTA to deconvolve them.[3] Each overall experiment is performed on an independent brain slice of a different mouse, so the shapes and spatial features of each experimental sample overall consitute a large difference between experiments. Each gene within each experiment will have a somewhat different pattern (and it will be our next goal to distinguish those differences and similarities), but we first tested whether genes within the same experiment could be classified together based on their overall spatial features. We implemented training as above and then clustered the mixed experiment gene matrices into 4 clusters. The confusion matrix shows clustering labels are largely consistent with true experimental labels (Table S3).

We next performed a shuffling test on gene matrices from one Slide-seq experiment, to break correlated patterns of neighboring regions in the way described for the shuffling of synthetic data above. We trained a new model and examined model-reported similarity among expression patterns of ten random genes. If CoSTA successfully learned spatial features that distinguish the expression of these genes, the distances between two gene patterns should change when spatial patterns and relationships between neighboring pixels are disrupted. We randomly selected *Prdx5* as the reference gene and calculated Euclidean distances of 9 other genes with it. We order these ten genes based on their distances to *Prdx5*. Then, we shuffled gene matrices 100 times, passed the shuffled matrices through the trained ConvNet, and recalculate paired distances with *Prdx5* (Fig.3a). We find that in 5 of 9 comparisons, distances decreased upon shuffling, as the distinctive patterns captured by CoSTA were removed by shuffling, converting the matrices into generic, more similar patterns. In 4 of 9 comparisons, distances increased with shuffling, likely indicating that key similarities between the spatial patterns became disrupted during shuffling (Fig. 3b). In contrast, the similarity measured by overlap analysis would not change after shuffling since individual pixels were shuffled identically. This result suggests that the learned features by CoSTA are relevant to the spatial expression pattern.

We next applied CoSTA to reanalyze two spatial transcriptomics datasets measured by Slide-seq.[3] These datasets are derived from two biological conditions: 3 days after brain injury ("3 days") and 2 weeks after brain injury ("2 weeks"). In the first investigation of these two datasets in Slide-seq, Rodriques et al. primarily focused on genes that were spatially correlated with *Vim*, *Ctsd* and *Gfap* at both 3 days and 2 weeks after brain injury.[3] For comparison, we also examined genes correlated with *Vim*, *Ctsd* and *Gfap* from our CoSTA results. One property of our approach is that features of each gene change every epoch when weights are updated. This may result in changes to the nearest neighbors of a gene during model training and can be used to infer how strong and stable the inferred spatial patterns are in a given condition. We measured the overlap between detected *Vim, Ctsd, and Gfap* neighbor genes before and after weight updating across training epochs, and we found neighbors tend to be more stable for the 2-week dataset than for the 3 days dataset (Fig. 3b and Fig. S3). This may indicate that in the acute phase after injury, *Vim*, *Ctsd* and *Gfap* are more variable and less spatially patterned, but these patterns become stronger at 2-week time point after injury.

To screen truly spatially patterned genes out from noise, we use ensemble learning. Briefly, we initialized 5 ConvNets and trained them separately. We then calculated the nearest neighbors

for every gene in the same dataset, at neighbor set sizes of 5, 10, 15, 20, 25, 30, 40, 50, and 100. We use a broad range of neighboring levels because we think different genes may form different sizes of communities. Next, we calculated Jaccard similarities across the 5 CoSTA models and keep genes that have an averaged Jaccard similarity larger than 0.2 at least in one level. Through ensemble learning, we obtain a refined set of SE genes by CoSTA. The majority of the SE genes identified by CoSTA are also called SE by SPARK (Fig. S4a, b). *Vim*, *Ctsd*, and *Gfap* passed this threshold in 2-week data but do not pass the threshold to be considered SE genes for the 3-day dataset. Overall, a smaller proportion of genes were considered SE at 3 days, consistent with the more variable gene neighbors observed for 3-day above. Notably, *Vim*, *Ctsd*, and *Gfap* are also not present in the 3 days SE gene list identified by SPARK, and only *Ctsd* and *Gfap* were identified as SE genes by SPARK in the 2 weeks data. We call genes that pass the threshold are “stable”, and genes that are filtered out as “unstable”. We propose that the percentage of stable vs. unstable genes represents the degree of spatial patterning in the experiment set. We also note that less strongly patterned genes could reflect actively variable biological regulation (such as might happen during acute response to injury), not only technique noise. Unfortunately, we are unable to distinguish a weak spatial pattern from inherent noise, because of lack of “ground truth” for pattern matching. However, we can devise systematically noisy datasets by shuffling true datasets. This test serves as our final control for CoSTA’s ability to distinguish patterns from random noise. We shuffled a whole set of gene matrices from 3 days and 2 weeks and applied CoSTA to these two shuffled datasets. We reason that if spatial expression patterns of all genes were random, CoSTA should not learn any meaningful spatial features. We find that this shuffled dataset has overall lower NMI than its original dataset during training (Fig. S4b; see Methods for details of NMI use). Further, more genes were filtered out in the shuffled 2-week data (Fig. S4c). This demonstrates that CoSTA captures spatial features that are distinct from noise. For true 3-day and shuffled 3-day data, the numbers of genes that pass the threshold do not have an obvious difference (Fig. S4c). This may indicate that the inherent noise within 3-day dataset is so high that it is not very distinct from systematically simulated noise. Indeed, few patterns are visually obvious for example gene matrices from 3 days (Fig. S5a). However, we note that CoSTA on true 3-day data did pull out more SE genes that overlap with SPARK SE genes than did CoSTA with shuffled data (Fig. S4a), indicating that some patterns are consistently detectable and specific in the true data.

We focused our further analysis on the 2-week data. We applied SpatialDE and SPARK to this dataset for comparison to CoSTA. The original Slide-seq publication previously identified 843 genes that are correlated with *Vim*, *Ctsd*, and *Gfap* via overlap analysis.[3] However, our CoSTA, with a rigid threshold, identified many fewer correlated genes (63 with z-scores < -2.325), and only 19 genes matched the original Slide-seq set (Fig. 3c). SPARK first identified 1294 significantly SE genes and then clustered them into 10 groups by hierarchical clustering with individual pixels as features. Our CoSTA correlated gene list only has 5 gene overlaps with genes that are grouped with *Vim*, *Ctsd*, and *Gfap* by SPARK. This further supports that correlated genes identified by CoSTA are different from what is obtained using individual pixel similarity. We also used SpatialDE to find significant SE genes. Surprisingly, the whole dataset passed the SpatialDE test for significant spatial expression. Then, we applied the unsupervised pattern detection algorithm built in SpatialDE to cluster genes into 10 groups. This resulted in a large number of genes grouped with *Vim*, *Ctsd*, and *Gfap*. A majority of our CoSTA set (41 genes) overlaps with genes identified by SpatialDE (Fig. 3c). Though the set of correlated genes identified by CoSTA is much smaller than sets identified by the other 3 methods, we find that these genes are highly

enriched for meaningful biological function. In the original study, Rodriques et al. highlighted that genes correlated with *Vim*, *Ctsd*, and *Gfap* are enriched for functions in immune response, gliogenesis and oligodendrocyte development—all functions that are biologically expected in response to injury.[3] We found that the correlated genes identified by CoSTA have higher enrichment in immune response and gliogenesis than the genes identified by SpatialDE, SPARK and this original Slide-seq report (Fig. 3d). However, none of genes fall into category of oligodendrocyte development. When we visually inspected expression patterns of genes in the category of oligodendrocyte development, their individual and collective patterns do not have similarities to expression patterns of *Vim, Ctsd, and Gfap*. They are either noisy or expressed globally (Fig. S5b). As we noted before, overlap analysis tends to include more correlated genes that have high global expression. Therefore, certain genes would be called correlated simply because they have more overlaps. From results above, we conclude that CoSTA returns a reduced, stringent set of correlated genes while retaining biological significance.

Finally, we compared the types of spatial patterns detected by CoSTA and other previous methods. For each method (CoSTA, SpatialDE, SPARK, and the original Slide-seq overlap approach), we obtain the set of genes classified as spatially correlated with *Vim, Ctsd, and Gfap.* The SpatialDE list was generated by following default analysis procedure of SpatialDE.[4] Because SPARK doesn't have a built-in pattern detection algorithm, we used hierarchical clustering to assign SE genes identified by SPARK into 10 groups, as suggested in SPARK.[5] On the diagonal of Fig. 4a, we show the average expression pattern of the set of correlated genes obtained from CoSTA, SPARK, SpatialDE, and Slide-seq, respectively. Other images show expression patterns of genes unique to the method listed in the row vs. the method listed in the column. For example, the image on the 1st row and 2nd column is the expression pattern of correlated genes identified by CoSTA but not SPARK, and the image on the 2nd row and 1st column is the expression pattern of genes found in SPARK but not CoSTA. We note that CoSTA detects a localized, specific pattern of gene expression (bright in the upper middle) shared within its correlated gene set while the patterns detected by the other methods look similar to the average expression across all genes (Fig S6; thus being less distinctive to this specific correlated set). Using the learned spatial representation, we further clustered all CoSTA-determined SE genes at the 2-week time point into 6 groups. The cluster that contains *Vim*, *Ctsd*, and *Gfap* (cluster 3) is composed of 89 genes expressed in a distinct pattern (Fig. 4b and Table S4). Other clusters also successfully identify distinctive spatial patterns of expression (Fig. 4b and Fig. S6) We also used SpatialDE to cluster SE genes identified by CoSTA into 6 clusters. We found that the two methods share some commonalities in detecting patterns but also have some disagreements (Fig. S6).

Discussion

We have shown that our CoSTA approach can successfully implement deep learning ideas from computer vision to infer spatial gene expression relationships. Identifying spatial patterns from high-throughput spatial transcriptomics data is still challenging, however. We often do not have a clear ground truth answer for what should be detected as a pattern vs. noise and what similarities in patterns are most biologically relevant. Different approaches will have different strengths and weaknesses depending on the types of patterns and relationships to be detected. The very first step in any approach to analyzing spatial transcriptomics data is estimating significant SE genes. To identify SE genes, SpatialDE relies on the assumption that spatial expression of a given gene follows a multivariate normal distribution across spatial coordinates.[4]

However, this assumption leads all genes in a Slide-seq dataset to be identified as SE genes by SpatialDE. This may occur because noisy signals generated by the Slide-seq experiment may also follow or are confounded within the multivariate normal distribution. Therefore, a multivariate normal model will not be able to distinguish spatial patterns from noise in certain types of experimental data. Different from SpatialDE, both SPARK and CoSTA make use of kernels to identify SE genes. SPARK defined 5 periodic and 5 gaussian kernels to cover a range of possible spatial patterns that the authors believe are observed in common biological datasets.[5] Therefore, identifying SE genes involves a statistical evaluation of how well kernels match spatial patterns of interest. This SPARK approach is very valuable if an experimental dataset is accompanied by prior knowledge about relevant spatial patterns. Kernels in CoSTA also serve a similar purpose but are not predefined. Instead, kernels in CoSTA are learned through training a neural network. To identify SE genes, we rely on the idea that a true spatial pattern should be collective, which means a group of genes should share a spatial pattern. Therefore, when we apply kernels learned independently from 5 ConvNets, genes in the same group should have similar responses to these kernels. Conversely, a noisy gene expression pattern would respond to the 5 sets of ConvNet kernels differently, clustered with different groups of genes each time. Indeed, this kernel approach helps identify SE genes in Slide-seq data, and we see agreement between CoSTA with SPARK on the identification of SE genes, but without requiring an *a priori* definition of relevant patterns.

Identification of SE genes is just the beginning of extracting biological meaning from spatial gene expression. Careful analysis of the spatial relationships between genes is also necessary. Often, as in overlap analysis, studying gene relationships is based on vectorizing gene expression patterns and measuring their similarities in a latent space without considering spatial information such as the position of neighboring datapoints. One key motivation for CoSTA, therefore, is to preserve a spatial and shape representation of gene expression patterns. In comparison, SPARK does not have a pattern detection function, but can be combined with hierarchical clustering with pixels as features to assign each gene a pattern label. SpatialDE implements a clustering model based on a spatial Gaussian-process-based (GP) prior.[4] This clustering model is an extension of GMM with the addition of a spatial prior on cluster centroids. Therefore, pattern detection by SpatialDE goes beyond the pixel level. In our method, we define the key goal as learning a spatial representation for each gene. We have demonstrated that features learned by CoSTA are not isolated to individual pixels. Because of use of convolutional layers, spatial features learned by our method represent local patterns and multiple local patterns together form the global pattern for the gene matrix. Finally, vectorizing gene matrices allows us not only to find different spatial patterns within a dataset by clustering but also to study spatial relationships of pairs of genes. Such a pairwise examination, in contrast, is not implemented in SpatialDE.

Again, depending on the biological reality underlying the data, different approaches will have different advantages. The CoSTA approach will have advantages in cases where overall pattern shape is important, while direct overlap calculations may perform better when exact cell to cell correlation is more biologically relevant. The CoSTA approach may also have future applications to datasets in which images of different genes are not from the identical biological section, but instead from neighboring tissue slices, as is common in traditional histology. If a pattern or shape of expression is maintained while exact overlap is lost, CoSTA could still detect such a pattern similarity where an overlap approach would not.

Throughout our analyses, we find that overlap approaches, as well as SPARK and SpatialDE tend to capture more global patterns, grouping together as significantly correlated genes that are more distant in their spatial pattern relationships, while CoSTA captures a narrower and more specific set of genes, more likely to be based on local features of a pattern. This was observed in our analysis of digit image data as well as in applications to Slide-Seq and, to a lesser extent, MERFISH. This difference in outcomes again demonstrates the different advantages and disadvantages of different approaches. CoSTA would likely be more useful in a case where users want to narrow their set of candidate related genes and extract specific expression patterns. We also note throughout the Methods section alterations to parameters of CoSTA that could allow for detection of more general patterns.

Conclusions

In this study, we demonstrated that our deep learning CoSTA approach provides a different angle to spatial transcriptomics analysis by focusing on the shape of expression patterns. CoSTA includes more information about the positions of neighboring pixels than does an overlap or individual pixel correlation approach. CoSTA can be applied to any form of spatial transcriptomics data that are represented in matrix to find genes expressed in similar patterns as well as to evaluate the strength of the spatial patterning of each gene. We find that CoSTA captures more focused groups of spatially related genes while still detecting the biological function information found by other approaches that report larger sets of related genes.

Methods

Resize Gene Images and Normalization

The raw images of Slide-seq consist of over 1,000,000 pixels, which makes computation difficult. Therefore, we first binned 100 pixels into one pixel and resized matrices from different experiments into the same 48X48 image size. This results in a lower resolution, which may obscure small-scale fine details, but large scale global features of expression patterns of genes are preserved. CoSTA can be applied to any spatial transcriptomics dataset at any resolution, as long as the user has sufficient computational resources available. To avoid extreme computational burden, we recommend that users interested in high resolution features zoom into regions of interest and crop images in that region to efficiently apply CoSTA to their data. After binning, we normalized gene matrices as described in Svensson et al.[4] This normalization involves finding the total gene expression counts for each pixel across all gene matrices and then normalizing each pixel of each matrix by the log total counts across all matrices for this pixel. If this normalization is not performed, the expression of a gene could be over or undercounted at certain spatial locations where expression levels were systematically high or low for all genes. Normalization by total counts at each pixel ensures that our approach captures the spatial covariance for each gene beyond this potential artifactual effect. For visualization of expression patterns, we instead use averaged raw count values, and scale values from 0 to 1 divided by the maximum value. Thus, expression images in all figures are in 0 to 1 scale. This allows a more direct visual inspection of the raw data.

CoSTA Architecture

1. ConvNet

The ConvNet stage of CoSTA consists of 3 convolutional layers for Slide-seq and MERFISH analysis. Inputs are sets of spatial gene expression images (matrices) as described above. We first initialize a ConvNet randomly and then forward these gene expression matrices through the ConvNet. All weights in convolutional layers are initialized on a Xavier uniform distribution. Each convolutional layer is activated by a rectified linear unit function and is followed by a batch normalization layer and a max pooling layer to reduce the size of the output. To produce a feature vector for each gene, we flatten the matrix output from the last max pooling layer by concatenating all matrix columns into a single column. One fully connected layer is added to the model after the last max pooling layer with a customized softmax activation to produce outputs as probabilities (See **4. Loss Function**). The fully connected layer is only used during training, when we need gradients to pass backwards through the model. Once trained, this fully connected layer will be discarded, and we use L2-normalized outputs as the spatial representations. Specific parameters used in ConvNet, such as the number and size of filters in each convolutional layer, can be found in python code. We note that different numbers of convolutional layers have been used for different image classification tasks. We recommend that users start with a 3-convolutional-layer network for initial data exploration. However, if a dataset has a larger size of gene matrices, outputs from the 3-convolutional-layer network will be very long vectors. Therefore, users can increase the number of convolutional layers to decrease the dimensions of outputs if needed.

2. UMAP and Clustering

The flattened spatial representation vector output from the three convolutional layers is reduced by UMAP before GMM clustering. We implemented UMAP using the original python source code[9]. We set up “n_neighbors=20” and “min_dist=0”, while using UMAP for dimension reduction. To cluster samples into N clusters, a user can reduce dimensions to N UMAP-dimensions. In this study, we reduce all samples to 30 UMAP-dimensions and cluster all samples into 30 clusters by GMM. While 30 clusters are used here for the model training purpose, once the model is trained, the user can use the final output vector of spatial features to cluster genes into any number of groups desired. To test the influence of the initial choice of number of clusters, we tested 10, 20, and 30, 50, 75, and 100 clusters in 2-week Slide-seq data. Using larger numbers of clusters leads to the identification of fewer SE genes (Fig. S7a). Our model can converge no matter how many clusters are used for training (Fig. S7b). For a purpose of comparison, we called the 15 nearest genes of *Vim*, *Ctsd*, and *Gfap* individually, and total 45 genes in one test as correlated genes were used for comparing effects of the number of clusters. The choice of the number of clusters will influence the scale of correlated expression pattern detected (Fig. S7c). More global pattern differences will be detected using smaller numbers of clusters while finer scale pattern distinctions are detected with larger numbers of clusters (Fig. S7c). Increasing the number of clusters will also bring a disadvantage of larger computational cost and longer training time (Table S5). In this case, 30 clusters show good specificity, and the detected spatial pattern is not further refined with increasing cluster numbers (Fig. S7c). Without ground truth for a dataset, the number of clusters must be chosen based on the scale of patterns desired to be detected for a particular biological application and the results inspected visually.

3. Auxiliary Target Distribution as Soft Assignment

After clustering, we calculate centroids by averaging samples in the same cluster (Eq. 1).

$$Eq.1: c_i = \frac{1}{M_i}\sum_{j=1}^{M_i} x_{ij}$$

Where c_i is the centroid for the i^{th} cluster, M_i is the total number of samples in this cluster, and $x_{i,j}$ is a reduced UMAP vector for the j^{th} sample in the i^{th} cluster.

Then, each sample is assigned probabilities based on Euclidean distances to cluster centroids (Eq. 2).

$$Eq.2: p(y = i|\, x) = \frac{e^{1/d_i}}{\sum_{i=1}^{N} e^{1/d_i}}$$

Where d_i is the Euclidean distance of sample x to the centroid c_i, and N is the total number of clusters.

Next, we transform probabilities of each sample to an auxiliary target distribution using equation 3.

$$Eq.3: q_{ij} = \frac{p_{ij}^2/f}{\sum_{i=1}^{N}(p_{ij}^2/f)}$$

where $f = \sum_{j=1}^{M} p_{ij}$. i denotes the i^{th} cluster and j denotes the j^{th} sample, p_{ij} is probability that the j^{th} sample belongs to the i^{th} that we get through Equation 2. q_{ij} is the auxiliary target probability that the j^{th} sample belongs to the i^{th} cluster. This transformation was proposed by Xie et al, which is raising p_{ij} to the second power and then normalizing by frequency per cluster.[17] The use of power 2 is to highlight samples that have high confidence in the clustering task and discount samples for which the model is uncertain about cluster assignment.

4. Loss Function

To optimize the neural network, we use bi-tempered logistic loss based on Bregman Divergences as the primary loss function. Bi-tempered logistic loss was proposed by Amid et al and showed advantage of making supervised learning robust to noise.[10] To achieve the robustness, they devised tempered softmax function and tempered logistic loss and gave detailed mathematical reasons behind (Eq. 4 and 5). We reason that training CoSTA also faces the problem of unknown noise within the data, because clustering will assign wrong labels to samples. This is even true when clustering is based on the ConvNet that is randomly initialized. Therefore, use of bi-tempered logistic loss is to deal with wrong or uncertain labels generated by clustering. When both t_1 and t_2 are equal to 1, bi-tempered logistic loss is the common KL-divergence loss with softmax activation.

$$Eq.4: L = y_i\left(log_{t_1} y_i - log_{t_1}\hat{y}_i\right) - \frac{1}{2-t_1}\left({y_i}^{2-t_1} - {\hat{y}_i}^{2-t_1}\right)$$

Where $log_{t_1}(x)$ can approximate to $\frac{1}{1-t_1}(x^{1-t_1} - 1)$. y_i is the target value and $\hat{y}_i$ is the predicted value out of the fully connected layer.

$$Eq.5: \hat{y}_i = exp_{t_2}(\hat{\alpha}_i - \lambda_{t_2}(\hat{\boldsymbol{\alpha}}))$$

Where $\hat{\alpha}_i$ is linear activation of output of the fully connected layer for the i^{th} cluster, and $\lambda_{t_2}(\hat{\boldsymbol{\alpha}}) \in \mathbb{R}$ is s.t. $\sum_{j=1}^{k} exp_{t_2}\left(\hat{\alpha}_j - \lambda_{t_2}(\hat{\boldsymbol{\alpha}})\right) = 1$.

Center loss is an optional setting in our model. Center loss was first proposed to assist models to learn discriminative representations in supervised learning.[18] Optimizing models with center loss is equal to minimizing intra-class variation defined by Eq. 6.

$$Eq.6: L_c = \frac{1}{2}\sum_{j=1}^{M_i}\left\|x_i - c_j\right\|^2$$

Where c_i is the centroid of i^{th} cluster, and x_j is the hidden features of j^{th} sample in this cluster.

Because lowering center loss will push samples closer to the cluster center, the learned representations will be more discriminative in the hidden space. Though we did not use center loss to train models for Slide-seq data, we found that adding center loss during training can substantially improve accuracy in Fashion image data (Fig. S8) and the synthetic data with variance as 0.6. If a user has a biological dataset with some degree of known ground truth for comparison, initial data exploration should explore whether combining center loss and bi-tempered logistic loss is more appropriate to capture the known spatial features of the data.

5. Normalized Mutual Information

Unlike supervised learning, we do not have ground truth for training in the CoSTA approach. To monitor how well training proceeds, we use normalized mutual information (NMI) to compare clustering labels before and after weight updating across training epochs. Increase of NMI during training indicates a decreased changing of clustering labels and thus suggests convergence of model. We cannot hold aside a validation set during CoSTA training. Therefore, NMI also serves as a metric of overfitting. Once we do not observe a large jump of NMI in consecutive epochs, we consider that the model has converged.

6.Experiments with Common Image datasets

While experimenting with MNIST handwritten, USPS-digit, and Fashion image datasets that come with true labels, we noticed that the CoSTA approach can learn to predict more true labels than the model that is just initialized and exceeds UMAP+GMM with pixel values as features (Fig. S8). For the Fashion image dataset, CoSTA was greatly improved after we add center loss with bi-tempered logistic loss as a whole loss function. However, the learning ability of CoSTA with these datasets is less than with supervised learning approaches (typically >95% accuracy). The highest accuracy we got is 0.961 (MNIST handwritten), 0.931 (USPS-digit) and 0.686 (Fashion), as measured by NMI between the clustering label and true class label. NMIs achieved with CoSTA applied to the MNIST and Fashion datasets are higher than for all other deep learning clustering methods, and the CoSTA NMI for USPS scores second in the ranking of deep learning approach performance.[7] We also tested whether SpatialDE can identify patterns in these three image datasets. We used the automatic histology pattern detection implemented in SpatialDE to cluster images in MNIST handwritten, USPS-digit, and Fashion into 10 groups, and SpatialDE achieved

0.532 (MNIST handwritten), 0.658 (USPS-digit), and 0.568 (Fashion) NMIs, which are even lower than UMAP+GMM clustering with pixels (Fig. S8).

SE Gene Calling

To call out SE genes, we use an approach of ensemble learning. Simply put, we train 5 CoSTA models independently. We then calculate a set of nearest neighbors for every gene in the same dataset, using neighbor set sizes of 5, 10, 15, 20, 25, 30, 40, 50, and 100. This is because different genes with their neighbors may form a community with different sizes. Using a broad range of neighboring set sizes can enable us to include SE genes that only form a small community with a few genes as well as SE genes that fall into a large gene group. Next, we calculated Jaccard similarities across the 5 ConvNets and keep genes that have averaged Jaccard similarity larger than 0.2 at least in one level of neighbor set sizes: 5, 10, 15, 20, 25, 30, 40, 50, or 100.

Correlated Gene Calling

To find significant correlated genes, we use the learned features from one of 5 CoSTA models to calculate Euclidean distance pairwise between all genes. For example, to get significant correlated genes with *Vim*, we calculated distances of all other SE genes to *Vim* based on the learned features. Then, we used these distances to create a null distribution. Distances that have Z-scores lower than -2.323 ($p<0.01$) are considered significant, and genes that have significant distances would be called out as correlated genes to *Vim*. Because we trained 5 independent models, we obtain 5 sets of correlated genes for each SE gene in the data. Then, we keep correlated genes that show up in at least in 3 models.

MERFISH Analysis

We obtained the MERFISH dataset collected on the mouse preoptic region of the hypothalamus from Dryad[14](https://datadryad.org/stash/dataset/doi:10.5061/dryad.8t8s248), and we used the slice at Bregma + 0.11 mm from animal 18 for analysis as used for SPARK analysis.[5] We reduced the image resolution 10-fold and resized images to 85X85 matrices. Next, we directly applied a customized CoSTA model to the MERFISH dataset. This customized approach has the same general architecture that defines CoSTA, as described above. The customized ConvNet also has three convolutional layers but each convolutional layer has a larger filter, to reduce the overall size of the output. To compare with results from SPARK, we created null distributions for correlated gene calling by permuting images 100 times. Permuted images are forwarded through CoSTA to get permuted spatial features. Then we calculated their Euclidean distances with the spatial features of the true image, and these distances serve as the null distribution. Because the 9 defined cell type expression patterns are known, significantly correlated genes to these 9 expression patterns were called SE genes. For each gene in this MERFISH dataset, including the 5 blank controls, we calculated its Euclidean distances and its 100-time shuffled distances to the 9 expression patterns. If the true Euclidean distance of one gene to one cell type pattern are lower than Z-score -2.323, we call this gene an SE gene that is correlated to the expression pattern typical of this particular cell type. To visualize the training process, we project the feature vectors of each gene onto the first two UMAP dimensions and label each gene according to clusters defined using the whole feature vector. We use a linear intrinsic dimensionality (LID) estimator to quantify the change in cluster distinctness before and after training. This estimator mainly measures a ratio between distance of each datapoint to its the second closest datapoint and distance to its closest datapoint. Ratios are ordered from low to high and it fits a line that crosses the origin. The slope of this line represents the LID of this data in the latent space. Simply put, the

lower LID, the more clustered datapoints are in the latent space. Indeed, among 10 different runs, spatial representations after training show lower LIDs than without training.

Analysis of Slide-seq with SPARK and SpatialDE

Analysis of Slide-seq with SPARK and SpatialDE follows the standard analysis pipelines proposed by these two methods, with default parameters. Code of analysis can be found at the GitHub repository (https://github.com/rpmccordlab/CoSTA).

Figure Legends

Fig. 1

CoSTA model approach and motivation. a, Overall CoSTA pipeline. Inputs are gene matrices from spatial transcriptomic experiments. ConvNet stage forwards images through 3 convolutional layers and then flattens the output into a spatial representation vector. UMAP reduces dimensionality of the spatial representations from the ConvNet stage before these gene representations are used to cluster genes with GMM. Each gene is then assigned cluster probabilities based on distances to cluster centroids, which are transformed to an auxiliary target distribution that can be minimized by reducing bi-tempered logistic loss and/or center loss. Gradients are backpropagated through a fully connected layer to ConvNet. The process is repeated until the model converges, at which point the output from the ConvNet is used as the final spatial representation (red arrow). b, Biologically-inspired example in which overlap does not capture all aspects of spatial pattern similarity. Rectangles represent an epithelial cell layer while ovals represent stromal cells. By overlap comparison, Gene 1 has the same similarity to both Gene 2 and Gene 3 (40% overlap). However, the biologically relevant expression along the epithelial layer is only shared between Gene 1 and Gene 2. Detecting this shape similarity requires learning a spatial representation. c, Performance of CoSTA in synthetic datasets. left panel: the 3 real expression patterns in mouse olfactory bulb data replicate 11; right panel: learning curves of CoSTA in 5 synthetic datasets with different noise levels. NMIs are measured between clustering labels by CoSTA and true class label.

Fig. 2

Analyzing MERFISH data with CoSTA approach. a and b, Visualization of the spatial feature vectors obtained for each gene, blank control, and cell type pattern from MERFISH data in a 2D UMAP layout. a, features extracted from a randomly initialized ConvNet with no training. Each dot is a gene, blank control, or cell type pattern. Colors indicate cluster labels obtained from clustering on the full feature vectors; b, features extracted by trained ConvNet. Each dot is colored with the original clustering labels from *a* to show how some cluster memberships rearrange. c, Local intrinsic dimensionalities of spatial representations by CoSTA without and after training (10 independent runs of CoSTA). d, CoSTA-detected spatial correlations of genes identified as SE only by SPARK. Top row displays known cell type specific expression patterns for 3 cell types. Lower rows display genes with expression patterns identified as significantly correlated to these cell types by CoSTA. *Chat, Avpr1a, and Nup62cl* were detected as SE genes by SPARK but not other approaches. Raw count values for each image are scaled from 0 to 1 to normalize the visual comparison.

Fig. 3

CoSTA Analysis of Slide-seq data. a, Shuffling test to disrupt spatial patterns. Left panel: The first row shows the three original spatial expression patterns of three example genes. Images in the second row are spatial patterns after shuffling (all images shuffled in the same way so that pixel-level overlap is preserved while spatial neighbor relationships are broken). Right panel: Distances between 9 randomly selected genes and *Prdx5*. Genes are ordered based on how close they are to *Prdx5* using spatial features extracted by CoSTA from true gene matrices (left to right: closest to farthest). Shuffled gene matrices are forwarded through CoSTA, and distances between gene pairs are subtracted from the unshuffled distances. Each point represents distance change for one shuffling (100 shufflings total). Red line at 0 indicates no change in distance would be observed using overlap calculations. b, The number of overlapped gene neighbors of *Vim*, *Ctsd*, and *Gfap* before and after each weight updating across all training epochs (30 nearest neighbors considered, see Fig. S3 for different size neighbor sets). Results shown for two experiments: 3 days (blue) or 2 weeks (red) after brain injury. c, Overlap of gene lists correlated with *Vim*, *Ctsd*, and *Gfap* at 2 weeks identified by CoSTA, SPARK, SpatialDE, and overlap analysis ("Slide-seq"). d, GO term enrichment in the correlated gene sets from different approaches for biologically relevant functions identified by the original Slide-seq analysis. Quantified along the axis is the fraction of genes in each method's correlated gene list that are annotated with the given GO term.

Fig. 4

Collective expression patterns detected in Slide-seq data. a, Collective expression pattern of *Vim*, *Gfap*, *Ctsd* and their correlated genes after 2 weeks brain injury defined by different methods. Patterns on the diagonal derived from genes correlated with *Vim*, *Gfap*, *Ctsd* defined by CoSTA, SPARK, SpatialDE, and overlap analysis, respectively. Other images show expression patterns of unique genes identified by 1 approach (row) over another approach (column). For example, the image on the first row and 4th column is the expression pattern of correlated genes found by CoSTA but not Slide-seq overlap analysis, and the image on the 4th row and 1st column presents expression pattern of genes identified by Slide-seq overlap as correlated to the key 3, but not identified by CoSTA. b, Gene clusters of CoSTA SE genes at 2-week time point on 2 UMAP dimensions. Mean expression patterns are presented for selected clusters. Visualization with raw count values that are scaled from 0 to 1.

Supplementary Fig. 1

Comparison of CoSTA and overlap analysis performance in finding correlated digits to digit 3. 1000 images are sampled from the full MNIST dataset, and each digit contains 100 samples. CoSTA (red bars) uniquely calls samples of digit 3 as correlated to digit 3. However, overlap analysis (blue bars) reports that all digits show some overlap with digit 3. CoSTA also reports a smaller number of digit 3 images but overlap analysis report a greater number of correlated digits overall.

Supplementary Fig. 2

Learning curve of CoSTA in true and shuffled synthetic datasets, with different variances. Clustering label generated by CoSTA is against true class label for measurement of NMI.

Supplementary Fig. 3

The number of overlapped neighbors of *Vim*, *Ctsd*, and *Gfap* before and after each weight updating across all epochs, considering either 10 nearest neighbors (left), 20 nearest neighbors (center), or 50 nearest neighbors (right).

Supplementary Fig. 4

The number of SE genes after 3 days and 2 weeks brain injury. a, Overlap of SE genes identified by SPARK, CoSTA learned with true data and CoSTA learned with shuffled data. b, learning curve of CoSTA with true and shuffled data. Y-axis shows NMI calculated between cluster labels at training epoch *t* and cluster labels at previous epoch *t-1*. X-axis shows training epoch *t*. c, Percent of all measured genes that are called SE genes by the 3 approaches.

Supplementary Fig. 5

Expression patterns of *Vim*, *Ctsd*, and *Gfap* 3 days and 2 weeks after brain injury. a, Expression patterns of *Vim*, *Ctsd*, and *Gfap* after 3 days after brain injury. b, Expression patterns of *Vim*, *Ctsd*, *Gfap* and genes involved in oligodendrocyte development (bottom row) 2 weeks after brain injury. Patterns that are visibly similar between *Vim, Gfap,* and *Ctsd* (small red boxes) are not strikingly visible in oligodendrocyte development genes.

Supplementary Fig. 6

Expression patterns of SE genes identified by CoSTA 2 weeks after brain injury. SE genes were clustered into 6 groups by SpatialDE and CoSTA. CoSTA cluster numbers correspond to Figure 4b and the most similar SpatialDE cluster is placed below the corresponding CoSTA cluster when possible. Average expression pattern in 3rd row shows the overall pattern of all genes combined in the 2-week dataset.

Supplementary Fig. 7

Effect of cluster number on CoSTA results with 2-week post injury Slide-seq data. a, SE genes identified by CoSTA with 10-100 clusters. b, CoSTA learning curve with 10-100 clusters. Y-axis shows NMI calculated between cluster labels at training epoch *t* and cluster labels at previous epoch *t-1*. X-axis shows training epoch *t*. c, Mean expression pattern of genes found to be correlated with *Vim*, *Gfap* and *Ctsd* identified by CoSTA with cluster numbers ranging from 10-100. Raw count values are scaled from 0 to 1 for these visualizations.

Supplementary Fig. 8

CoSTA approach applied to clustering USPS, MNIST and Fashion datasets. Left panels: Models were trained for 10 epochs. After each weight updating, we clustered images into 10 clusters and directly compared them to true class labels through NMI. The grey line indicates clustering by UMAP+GMM with pixel values as features. The black line indicates clustering by SpatialDE. The orange line represents learning with combined center loss and bi-tempered logistic loss in Fashion dataset. Right panels: NMIs between clustering at the t^{th} updating and the previous $(t-1)^{th}$ updating.

Supplementary Table 1

Comparison of CoSTA and SpatialDE on 5 true and shuffled synthetic datasets. Adjusted Rand Index and Normalized Mutual Information are used to measure the ability of separating different

spatial patterns. For shuffled data, each gene matrix still keeps its ground truth label but the original spatial pattern is disrupted.

Supplementary Table 2

Clusters of SE genes identified by CoSTA in the MERFISH dataset (cell type patterns are included in clusters).

Supplementary Table 3

Confusion matrix of clustering labels derived from CoSTA results compared to the original known experimental label.

Supplementary Table 4

Genes in each cluster of SE genes detected by CoSTA from the 2-week data, as shown in figure 4b.

Supplementary Table 5

Runtime of CoSTA for 3-day and 2-week Slide-seq data. Runtimes are measured in minutes and under different numbers of clusters being assigned during training.

Abbreviations

ConvNet: convolutional neural network

SE or SV gene: spatial expression or spatial variable gene

CoSTA: unsupervised ConvNet learning strategy for spatial transcriptomics analysis

Declarations

Ethics approval and consent to participate
Not applicable

Consent for publication
Not applicable

Availability of data and materials
The processed Slide-seq datasets were retrieved from https://singlecell.broadinstitute.org/single_cell/study/SCP354/slide-seq-study. We also deposited processed MERFISH and Slide-seq data and scripts for all analyses in this study at the GitHub repository (https://github.com/rpmccordlab/CoSTA) under an Open Source Initiative compliant MIT license. The version of the code used in the manuscript is available at DOI: 10.5281/zenodo.3948711.

Competing interests
The authors declare that they have no competing interests.

Funding
This research was supported in part by NIH NIGMS grant R35GM133557 to R.P.M.

Author Contributions
Y.X. conceived the project, developed the computational approach, and performed all analysis. R.P.M. advised the project, and Y.X. and R.P.M. wrote the manuscript.

Acknowledgments
We thank Tian Hong, Tongye Shen, and Amir Sadovnik for insightful discussion.

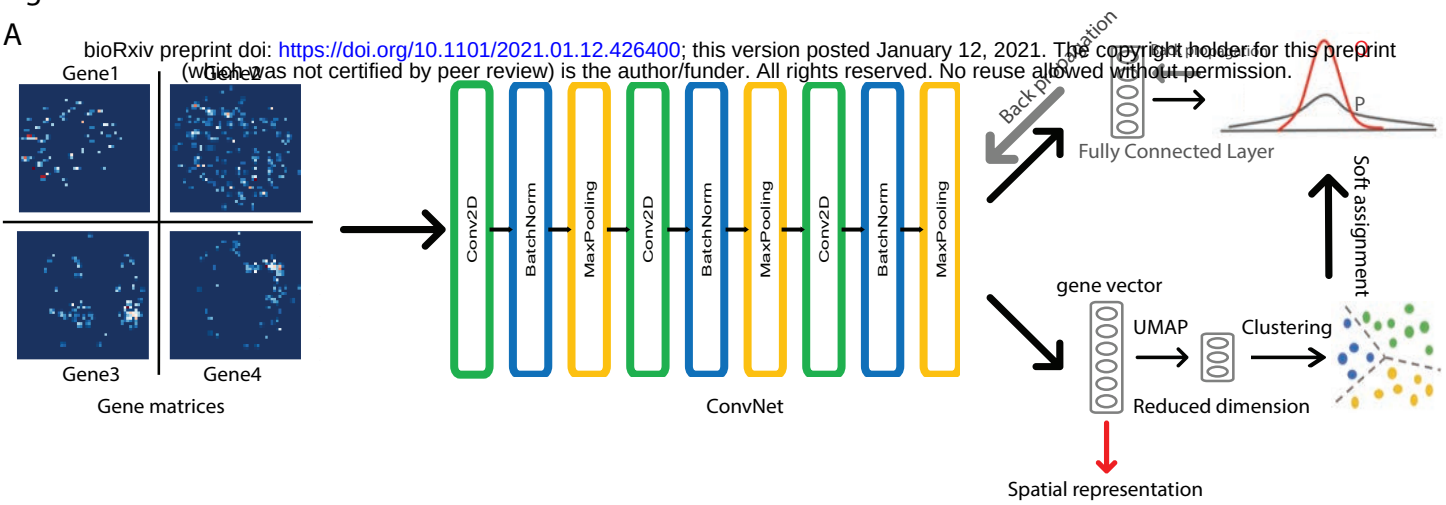
References

1. Burgess DJ: **Spatial transcriptomics coming of age.** *Nature reviews Genetics* 2019, **20:**317.
2. Eng C-HL, Lawson M, Zhu Q, Dries R, Koulena N, Takei Y, Yun J, Cronin C, Karp C, Yuan G-C, Cai L: **Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH.** *Nature* 2019, **568:**235.
3. Rodriques SG, Stickels RR, Goeva A, Martin CA, Murray E, Vanderburg CR, Welch J, Chen LM, Chen F, Macosko EZ: **Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution.** *Science (New York, NY)* 2019, **363:**1463.
4. Valentine S, Sarah AT, Oliver S: **SpatialDE: identification of spatially variable genes.** *Nature Methods* 2018, **15**.
5. Sun S, Zhu J, Zhou X: **Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies.** *Nature Methods* 2020, **17:**193-200.
6. Caron M, Bojanowski P, Joulin A, Douze M: **Deep Clustering for Unsupervised Learning of Visual Features.** 2018.
7. McConville R, Santos-Rodriguez R, Piechocki RJ, Craddock I: **N2D: (Not Too) Deep Clustering via Clustering the Local Manifold of an Autoencoded Embedding.** 2019.
8. Xie J, Girshick R, Farhadi A: **Unsupervised Deep Embedding for Clustering Analysis.** 2015.
9. McInnes L, Healy J, Melville J: **UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.** 2018.
10. Amid E, Warmuth MK, Anil R, Koren T: **Robust Bi-Tempered Logistic Loss Based on Bregman Divergences.** 2019.
11. Addison M, Xu Q, Cayuso J, Wilkinson DG: **Cell Identity Switching Regulated by Retinoic Acid Signaling Maintains Homogeneous Segments in the Hindbrain.** *Developmental Cell* 2018, **45:**606-620.e603.
12. Li D: **The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web].** *IEEE signal processing magazine* 2012, **29:**141-142.
13. Ståhl PL, Salmén F, Vickovic S, Lundmark A, Navarro JF, Magnusson J, Giacomello S, Asp M, Westholm JO, Huss M, et al: **Visualization and analysis of gene expression in tissue sections by spatial transcriptomics.** *Science (American Association for the Advancement of Science)* 2016, **353:**78-82.
14. Moffitt JR, Bambah-Mukku D, Eichhorn SW, Vaughn E, Shekhar K, Perez JD, Rubinstein ND, Hao J, Regev A, Dulac C, Zhuang X: **Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region.** *Science (New York, NY)* 2018, **362**.
15. Facco E, d'Errico M, Rodriguez A, Laio A: **Estimating the intrinsic dimension of datasets by a minimal neighborhood information.** *Scientific reports* 2017, **7:**12140-12148.
16. Edsgärd D, Johnsson P, Sandberg R: **Identification of spatial expression trends in single-cell gene expression data.** *Nature methods* 2018, **15:**339-342.

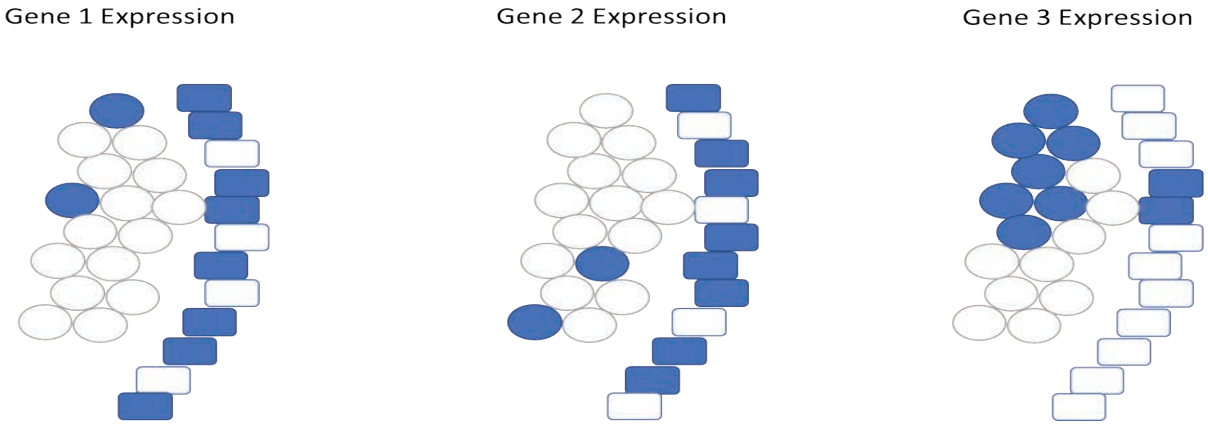
17. Yang J, Parikh D, Batra D: **Joint Unsupervised Learning of Deep Representations and Image Clusters.** 2016.
18. Wen Y, Zhang K, Li Z, Qiao Y: **A Discriminative Feature Learning Approach for Deep Face Recognition.** In *Computer Vision – ECCV 2016*; *2016//; Cham*. Edited by Leibe B, Matas J, Sebe N, Welling M. Springer International Publishing; 2016: 499-515.

Fig. 1

A



B



C

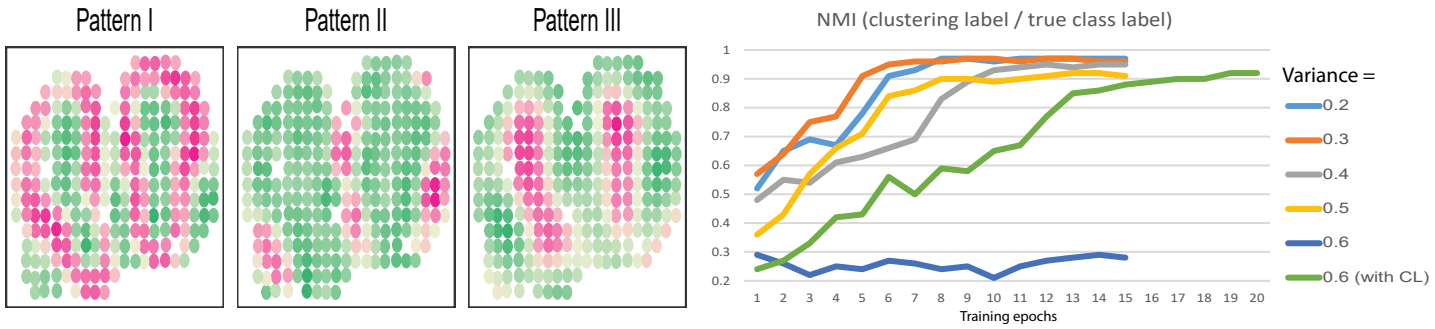
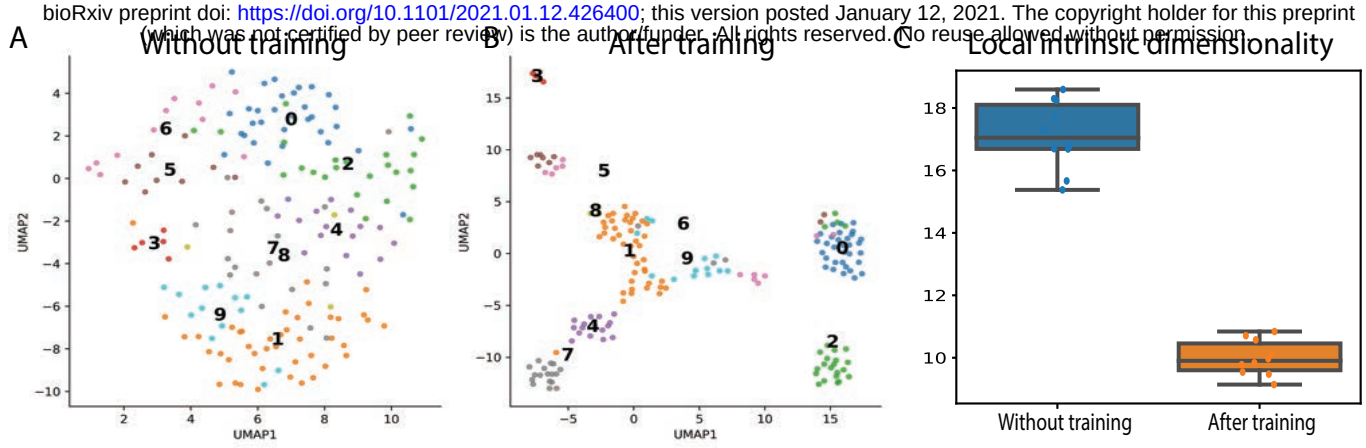


Fig. 2



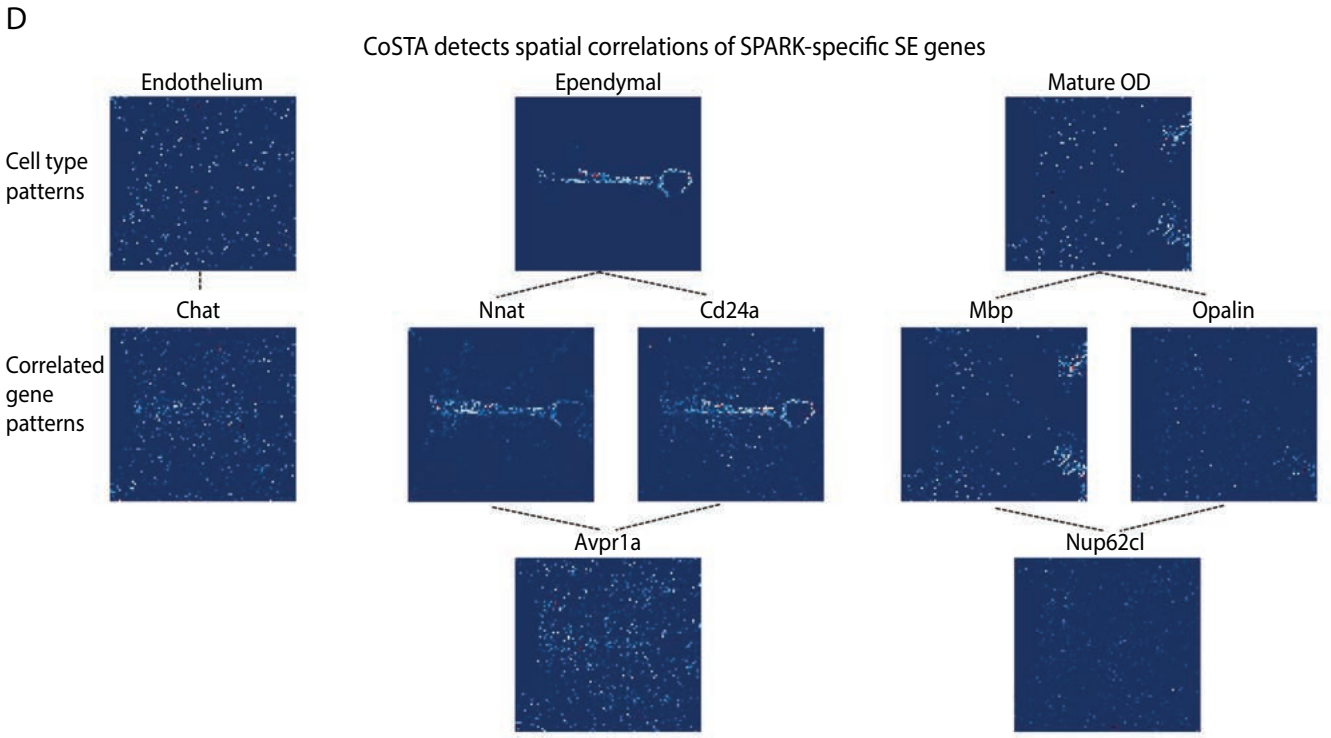


Fig. 3

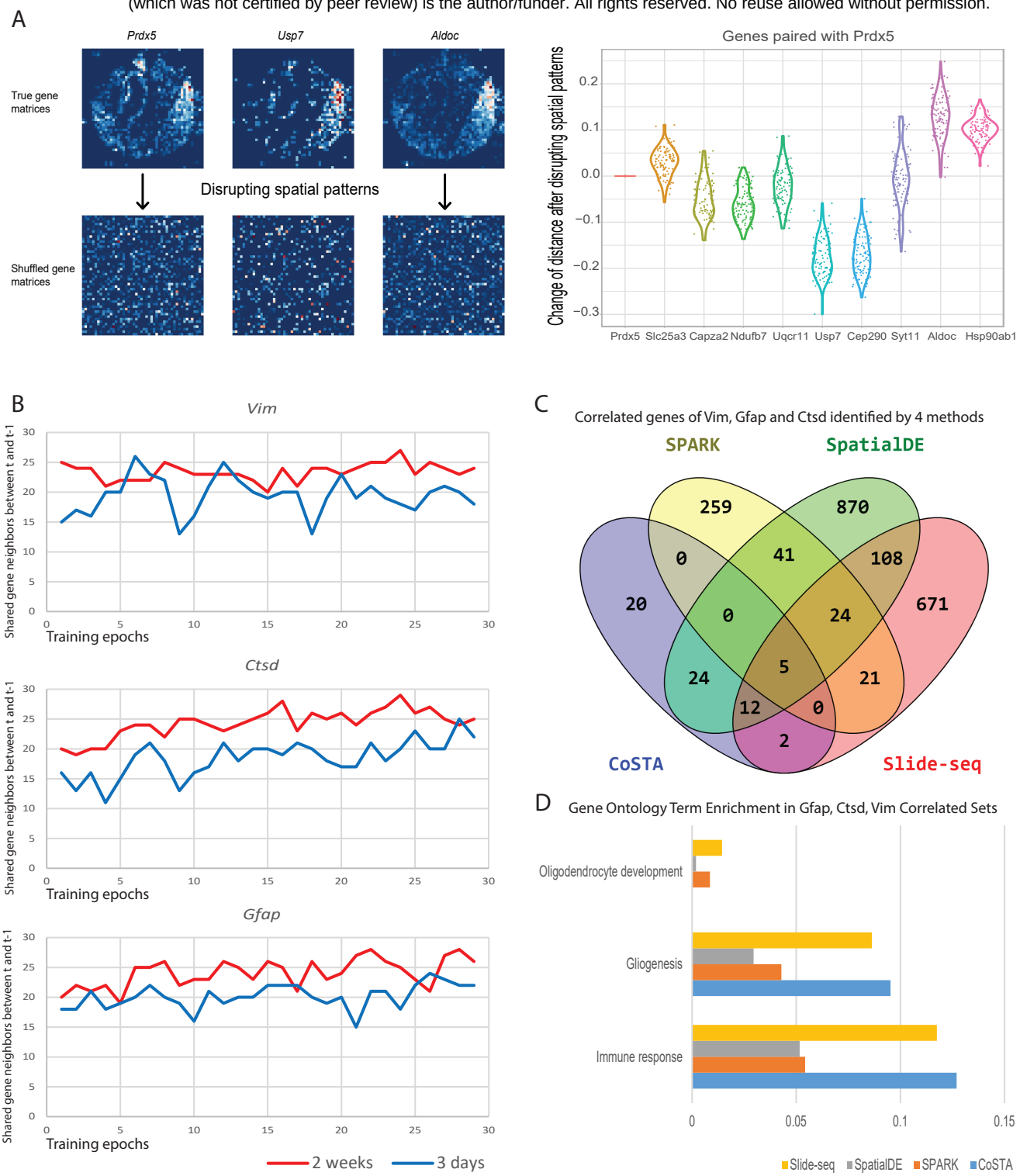
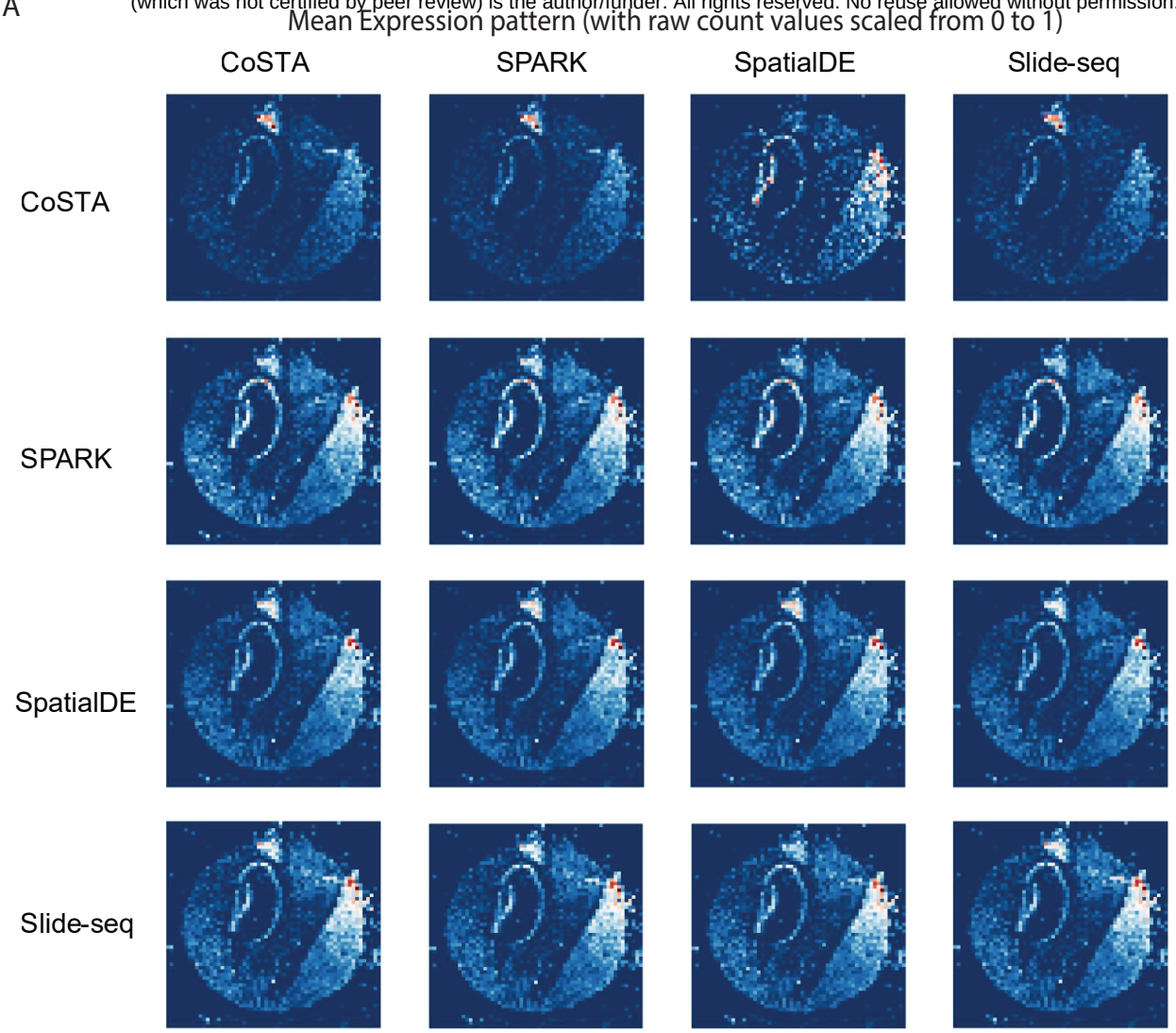


Fig. 4

A



B

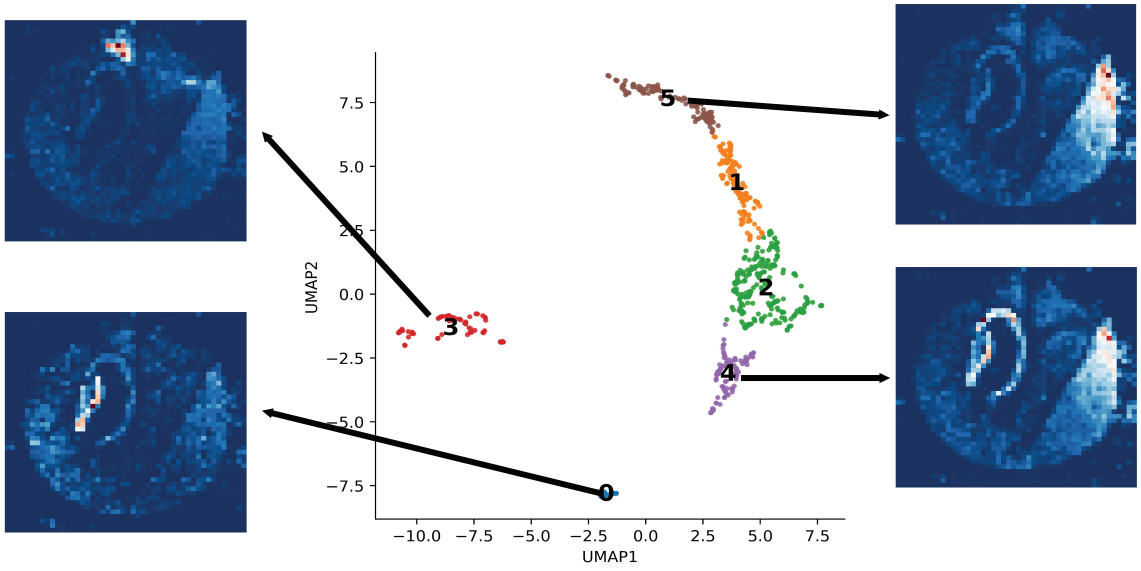


Fig. S1

Overlook of MNIST handwritten digit data

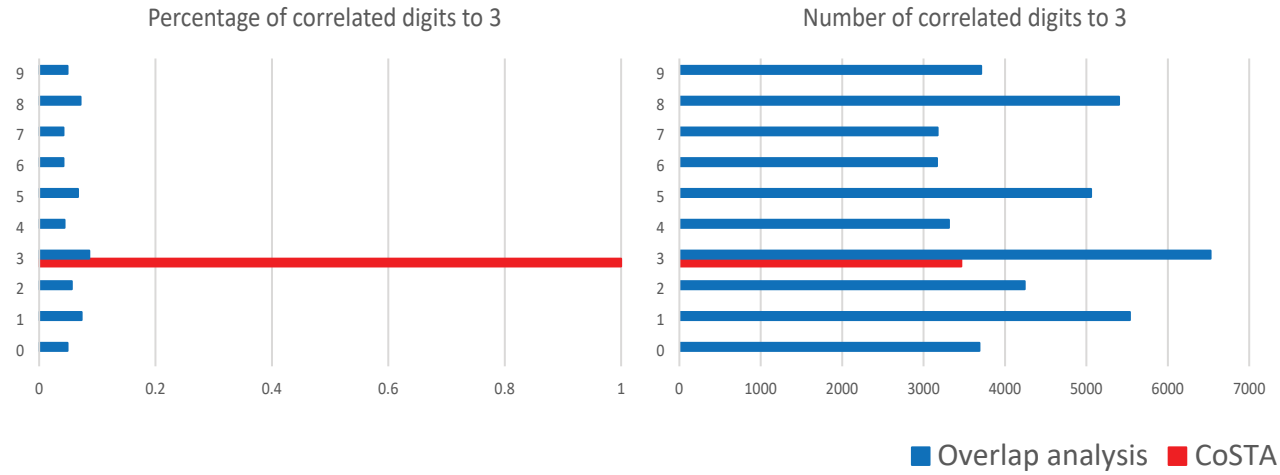


Fig. S2

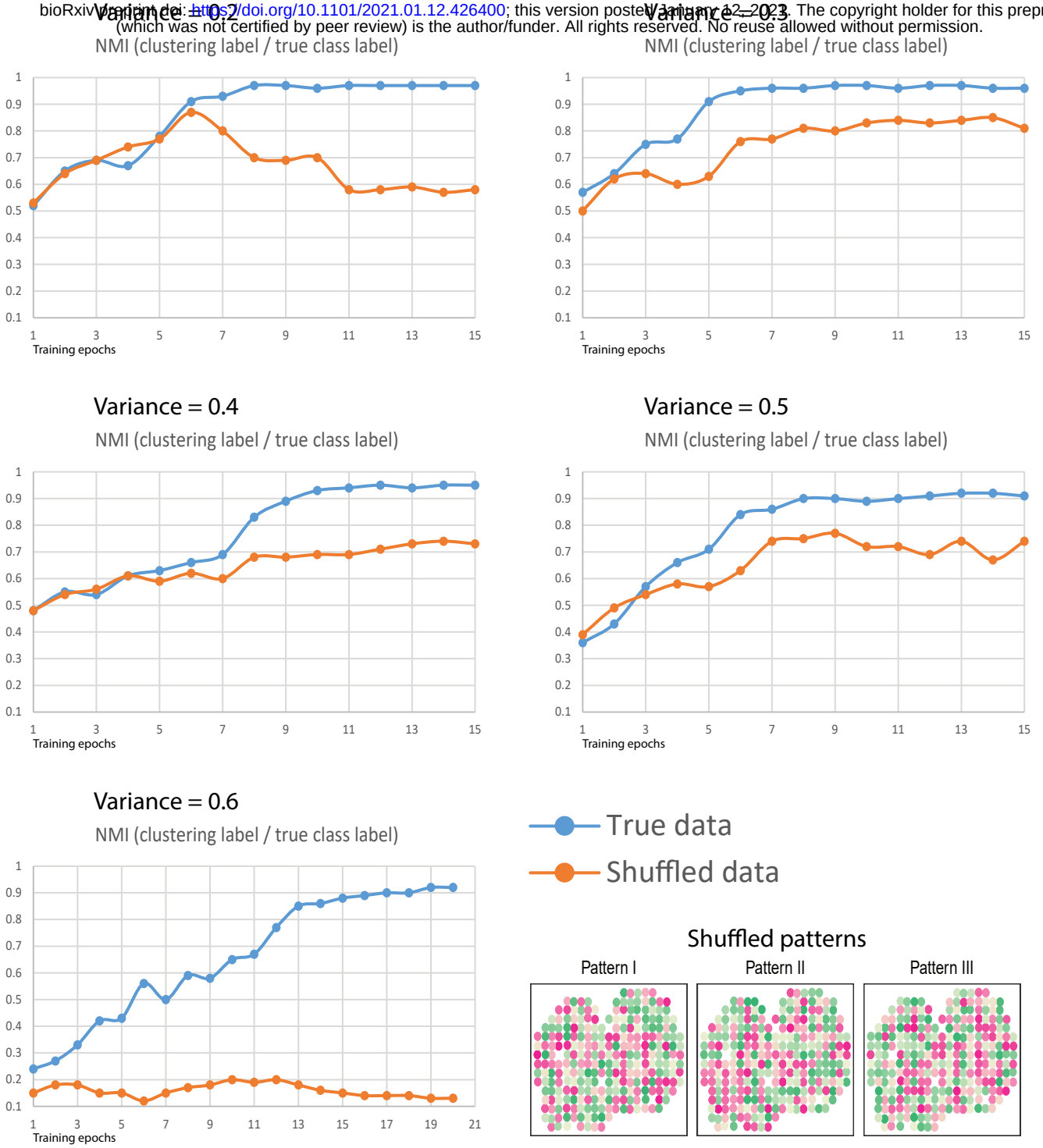


Fig. S3

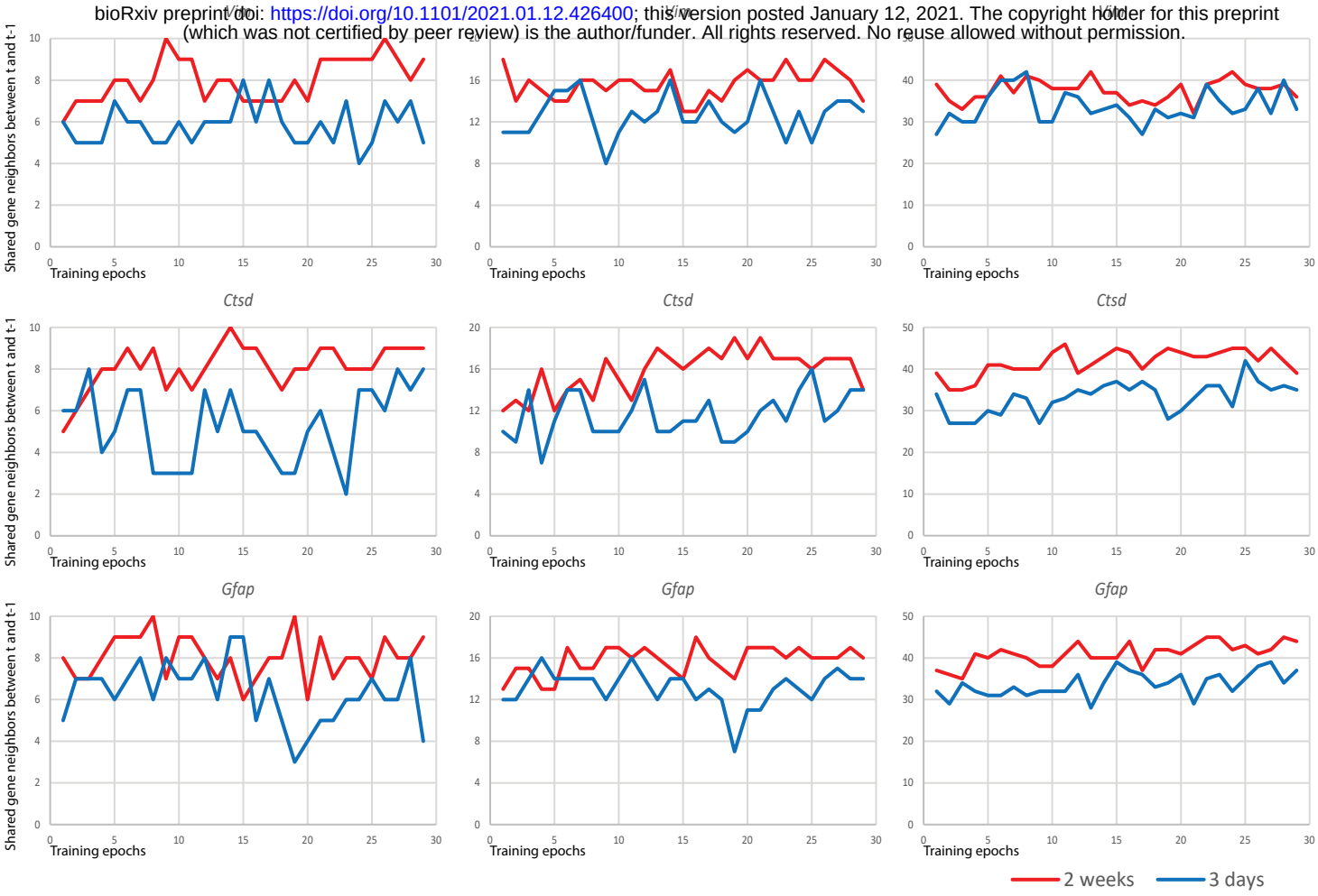
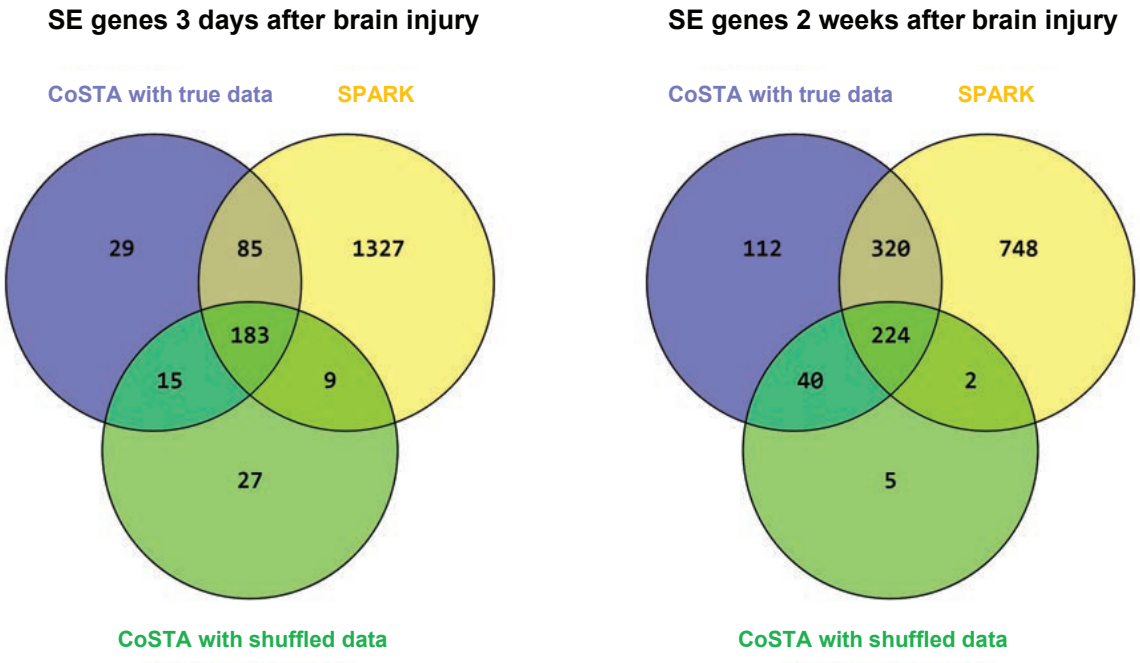
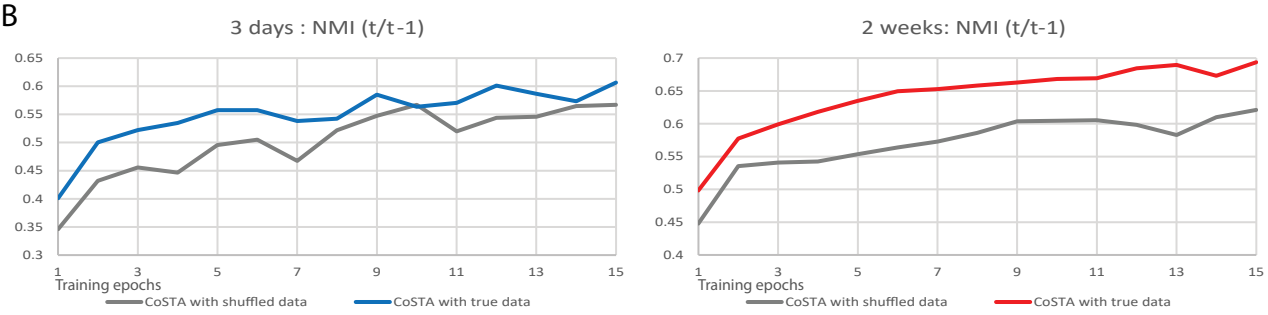


Fig. S4

A



B



C

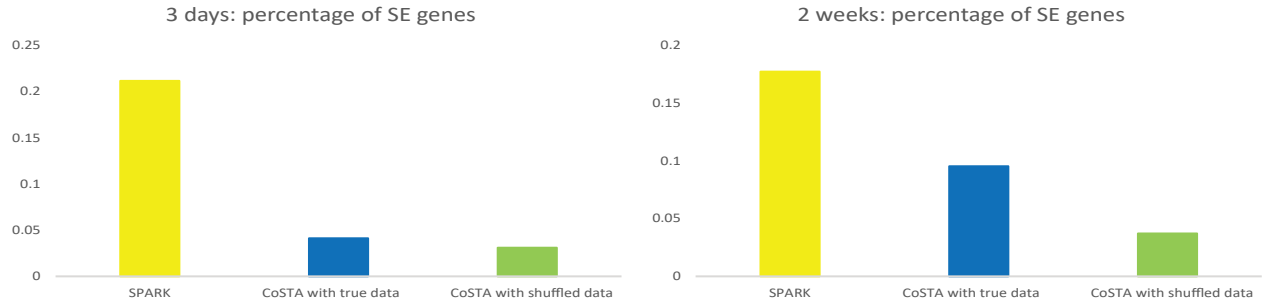
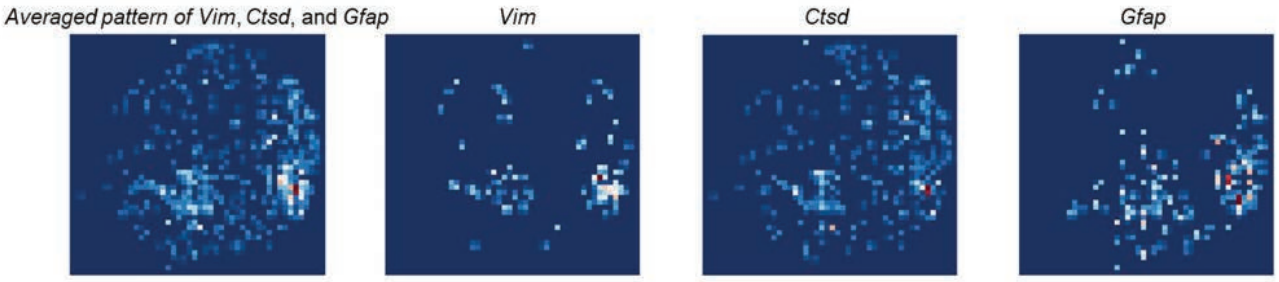


Fig. S5

A

3 days after brain injury



B

2 weeks after brain injury

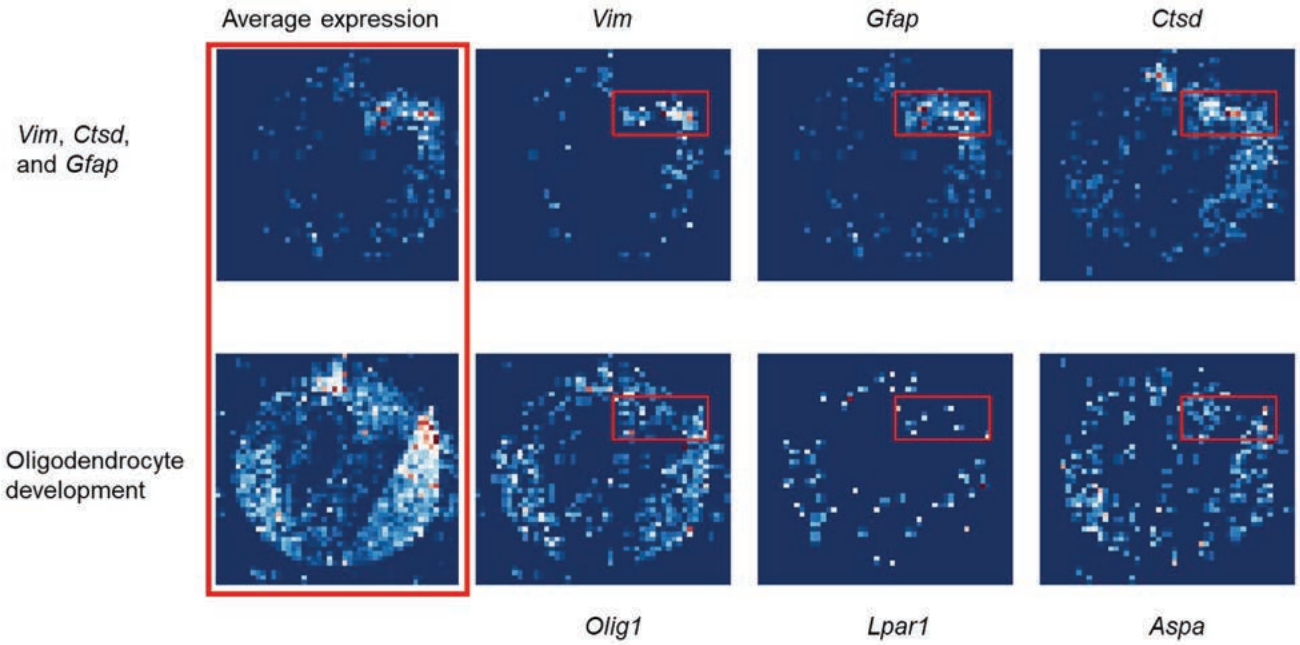


Fig. S6

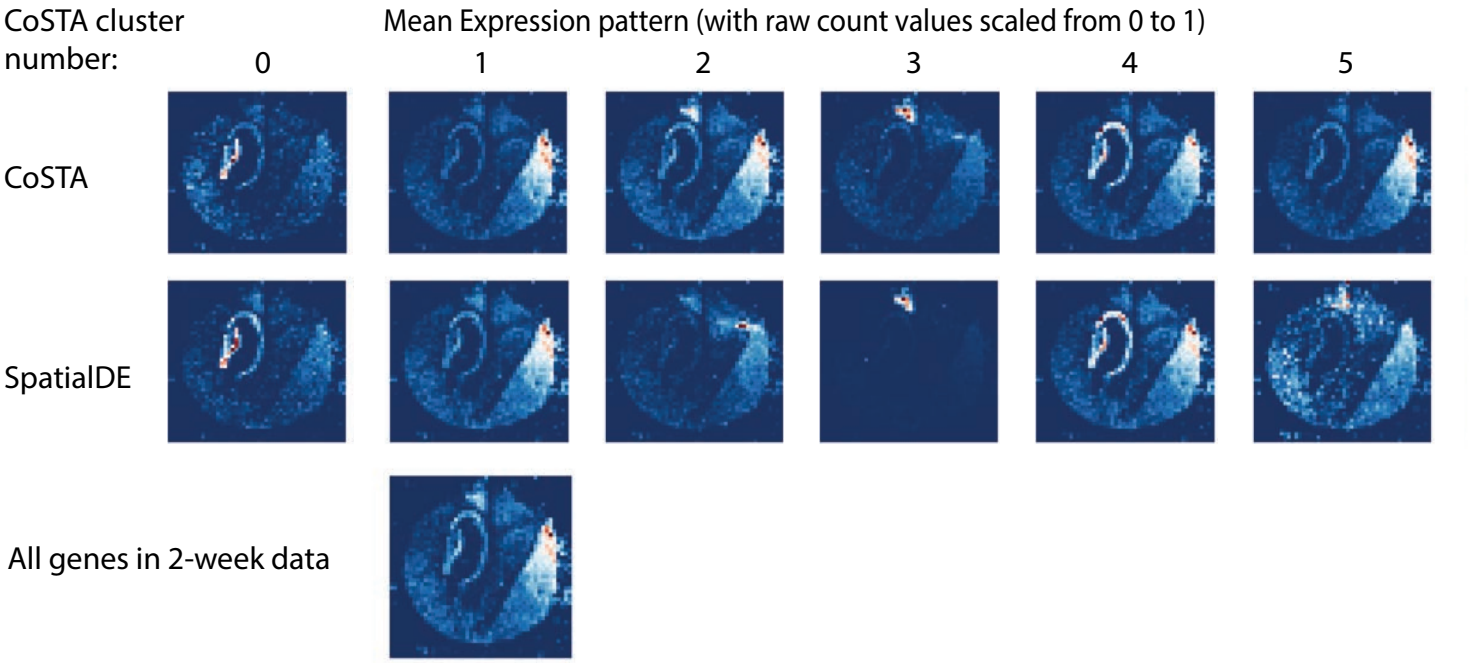
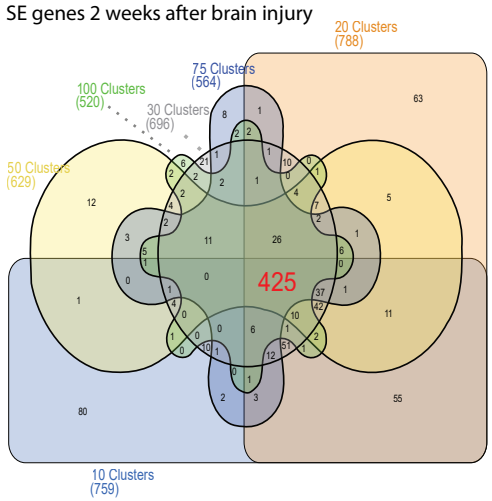
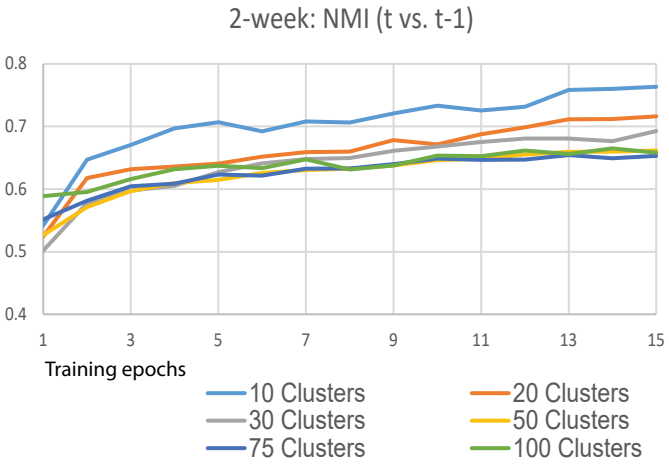


Fig. S7

A



B



C

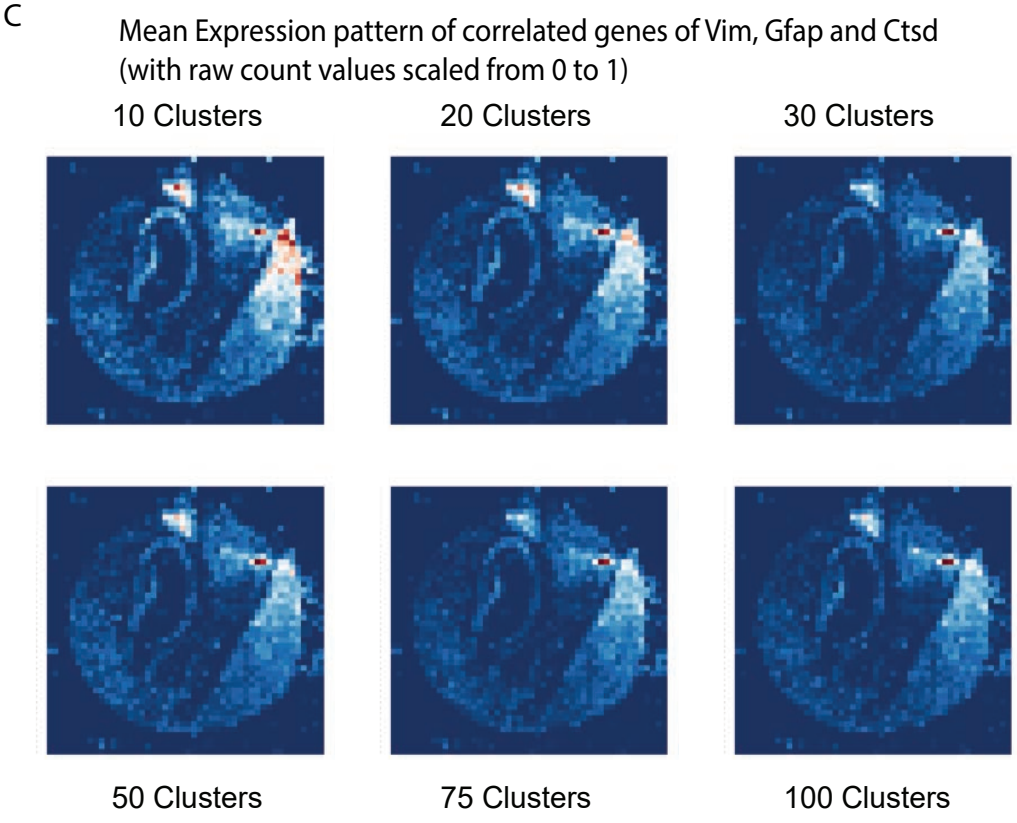
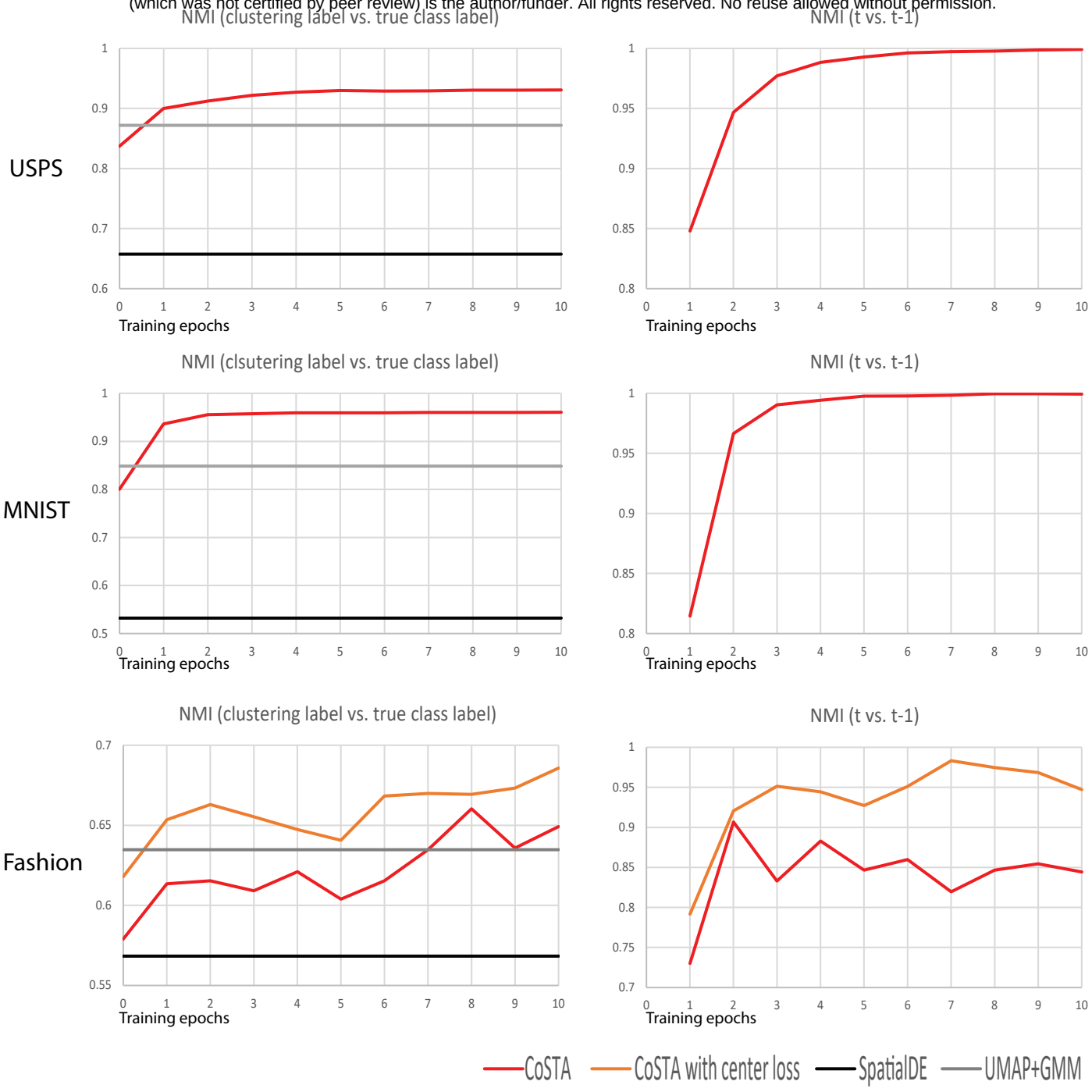


Fig. S8



Supplementary Table 1

NMI	CoSTA		SpatialDE	
noise level (variance)	True data	Shuffled data	True data	Shuffled data
0.2	0.97	0.87	0.99	0.72
0.3	0.97	0.85	1	0.99
0.4	0.95	0.74	0.99	0.98
0.5	0.92	0.74	0.98	0.97
0.6	0.29|0.92	0.23	0.95	0.95

Supplementary Table 2

Gene	Cluster
Endothelial 1	0
Ermn	0
Gabra1	0
Gjc3	0
Igf1r	0
Man1a	0
Ndrg1	0
OD Mature 2	0
Sema3c	0
Sgk1	0
Slco1a4	0
Ttyh2	0
Aldh1l1	1
Amigo2	1
Ar	1
Arhgap36	1
Astrocyte	1
Cbln1	1
Cbln2	1
Cckar	1
Cpne5	1
Creb3l1	1
Crhr2	1
Cspg5	1
Dgkk	1
Excitatory	1
Gabrg1	1
Galr1	1
Glra3	1
Gpr165	1
Htr2c	1
Igf2r	1
Inhibitory	1
Irs4	1
Isl1	1
Kiss1r	1
Onecut2	1
Oprd1	1
Oprk1	1
Oprl1	1
Pak3	1
Pnoc	1
Prlr	1
Rnd3	1
Scg2	1

	Cluster
Slc17a6	1
Sox4	1
Sox6	1
Sox8	1
Syt4	1
Tmem108	1
Adora2a	2
Bdnf	2
Brs3	2
Ccnd2	2
Chat	2
Endothelia	2
Gbx2	2
Gem	2
Grpr	2
Krt90	2
Lpar1	2
Microglia	2
Nts	2
OD Mature	2
Rgs5	2
Rxfp1	2
Selplg	2
Cdkn1a	3
Cenpe	3
Cplx3	3
Cyp19a1	3
Fezf1	3
Fn1	3
Klf4	3
Mbp	3
Ndnf	3
Necab1	3
Ntng1	3
Nup62cl	3
OD Mature	3
Opalin	3
Plin3	3
Ramp3	3
Slc17a8	3
Sp9	3
Sytl4	3
Tacr1	3
Calcr	4
Cxcl14	4
Esr1	4

	Cluster
Gad1	4
Gal	4
Gda	4
Npy2r	4
Penk	4
Rgs2	4
Serpinb1b	4
Th	4
Trhr	4
Coch	5
OD Immat	5
Pcdh11x	5
Pdgfra	5
Traf4	5
Crhbp	6
Cyr61	6
Ebf3	6
Endothelia	6
Fst	6
Gnrh1	6
Lmod1	6
Mki67	6
Myh11	6
OD Immat	6
OD Mature	6
Oxt	6
Pericytes	6
Sst	6
Syt2	6
Tac2	6
Ucn3	6
Adcyap1	7
Aqp4	7
Avpr1a	7
Cckbr	7
Cd24a	7
Ependyma	7
Etv1	7
Fos	7
Mlc1	7
Nnat	7
Nos1	7
Npy1r	7
Omp	7
Pou3f2	7
Sema4d	7

	Cluster
Slc15a3	7
Slc18a2	7
Sln	7
Tac1	7
Tiparp	7
Avpr2	8
Egr2	8
Galr2	8
Pgr	8
Synpr	8
Vgf	8

Supplementary Table 3

	0	**1**	**2**	**3**
0	2266	36	2	6
1	1	5117	115	157
2	0	78	7396	102
3	4	114	91	7085

Clustering label

Experiment label

Supplementary Table 4

Cluster 0
Cdkn1b
Vps13c
Hap1
Wbscr17
Scn3b
Pitpnm2
Gm26917
Marcksl1
Jph1
Vav3
Slc17a6
C1ql2
Sema5a
Nr3c2
Prox1
Rasl10a
Limd2
Plk5
Ahcyl2
Epha7
Tcf7l2
Fam163b
Vamp1
Dock10

Cluster 1		
Gpm6b	Psma7	Dlgap1
Ssb	Prrc2c	Cbx5
Lmo4	Peg3	Prkar1b
Prdx2	Sltm	Scn1b
Rock2	Ldha	Elavl4
Gria3	Soga3	Prkacb
Cmip	Phactr1	Atp2b2
Ppp1r9a	Tuba4a	B2m
Rtf1	Ankrd12	Lpgat1
Vamp2	Srrm2	Ndrg3
Snap47	Myo5a	Sncb
Arpp19	Strbp	Dclk1
Ensa	Klf9	Gnb1
Cfap36	Egr1	Trim37
Ube2k	Ntrk2	Sult4a1
Pdap1	Zfr	mt-Tp
Rsrp1	Map1a	Fam81a
Tuba1b	Luc7l3	Pak1
Rabep1	Kif21a	Kifap3
Acot7	Stmn1	Gabra1
Zc3h13	R3hdm1	Pde1a
Chd3os	Sptbn1	Ddx24
Syngr1	Fam171b	
Mapt	Eef1a2	
Eid1	Atp6v0e2	
Nrn1	6330403K07Rik	
Rgs4	Ptprn	
Zranb2	Mphosph8	
Nars	2210016L21Rik	
Zfand5	Rbm25	
Eif5b	Zfp365	
Mapk10	3110035E14Rik	
Atp5o	Rangap1	
Ntm	Ttyh1	
Prkcb	S100b	
Zmynd11	Sf3b1	
Psmc1	Plcb1	

Cluster 2
Dnaja1
Matr3
Dnm1
Snap25
Rab6a
Cox4i1
Arpp21
Gria2
Ywhag

Cluster 2		
Basp1	Pcmt1	
Fam168a	Gda	
Oxct1	Ncl	
Rufy3	Rph3a	
Apba2	Klc1	
Ndufb9	Mdh2	
Cdk5r1	Atp1a2	
Ghitm	Srpk2	
Ndufa10	Tmem50a	
Ppig	Sars	
Atp5c1	Nefm	
Aplp1	Rrp1	
Ndrg2	Son	
Scd2	Pgm2l1	
Srsf11	Nrxn2	
Tagln3	Cplx1	
Sbno1	Eif3c	
Uqcc2	Ctsb	

Cluster 3	
Trf	Car14
Xpnpep3	Pbxip1
Ppp1r1b	Hemk1
Sox5	Igfbp7
Gcnt2	Ier3
Tmem98	Cldn11
Cryab	C1qb
Phactr2	Stk39
Otx2	Cd63
Acaa2	Ezr
Atox1	Aldh2
Ifi27	Cox8b
Qdpr	C1qc
Haus2	Rbp1
Ccnd2	Vat1l
Slc31a1	Slc16a2
Crhbp	Vim
Gpr88	Gfap
Hist1h2bc	Tgfb2
Cnp	Lyz2
Itpk1	Lars2
Col1a2	Spint2
Igfbp2	Ctnnal1
Etfb	Npc2
Tnfaip8	Dcn
Ranbp9	Trpm3
Slco1c1	Calb2
Plin3	Mag
Elovl7	Cdkn1c
Krt12	Gng5
Ctsd	Sh3d19
Pcp4l1	Sostdc1
Ctss	Vamp8
Tesk1	Pvrl3
Kl	
Kcnj13	
Mgp	
S100a1	
Tbc1d9	
Pltp	
Cd59a	
Calml4	
Ccdc66	
Cab39l	
Gas6	
Nwd2	
Clic6	
Mal	
Nnat	
Igf2	
Folr1	
Fxyd1	
1500015O10Rik	
Slc16a6	
Prlr	

Cluster 4	
Cpne6	Chgb
Grin2a	Cfl1
Mycbp2	Npm1
Arpc1a	Nfib
Enc1	Ywhaz
Xist	Ncdn
Wasf1	Cacna1e
Gnaq	Ppp3r1
Hpca	Atp1a3
Zeb2	Btbd9
Epha4	Cttnbp2
Neurod6	Ap2a2
Olfm1	Ppfia2
Camk2b	Kif5c
Gria1	Ndufb7
Herc1	Ppp3cb
Grin2b	Ank3
Syn2	Brinp1
Auts2	Camkk1
Ubxn4	Mapk1
Nptx1	Capza2
Erc2	Gabra5
Thra	Nell2
Zbtb20	Bdnf
Kalrn	Nbea
Bcl11b	Napa
Cnih2	Ywhah
Celf2	Dynll1
Gng2	Ncam1
Camta1	Abr
Sh3bgrl3	Ptk2b
Rab2a	Rbfox1
Lppr2	Fam131a
Arf3	Rnf112
Syne1	Neurod2
Epha5	Nptxr
Cpne4	Ak5
Trim2	Snca
Ddx5	Cadm2
Synj1	
Sepw1	
Ogfrl1	
Pnmal2	
Zbtb18	
Nrxn1	
Tubb2a	
Rap1gds1	
2010300C02Rik	
Wipf3	
Arpc5	
Stxbp6	
Schip1	
Sirt3	
Pfn1	
Mrfap1	

Cluster 5		
Dlgap4	Itpr1	Smap1
Atl1	Hlf	Elavl3
Dgkz	Hpcal4	Igfbp6
Ktn1	Nlk	Prpf40b
Prkcz	Phactr3	Nudt4
Ndfip2	Tbl1x	Ppap2b
Rora	Flywch1	Tsnax
Ccl27a	Fam107a	Fgf12
Cdc5l	Oxr1	Luc7l
Fnbp1l	Rapgef4	Ube2r2
Plcb4	Btbd10	Serpine2
Pik3r1	Emc4	Lamp5
Pmm1	Slc1a3	Fabp3
Chga	Nos1ap	Kif3a
Cdk11b	Car10	Zfp91
D430041D	Pcdh7	Pfkm
Sv2b	Satb1	Ndufaf2
Ccdc186	Clstn3	Camk2g
Arhgap32	Srp72	Ldb2
Phf3	Stx1a	
Psmd2	Akap8l	
Mgll	Rabl6	
Pin1	Nap1l2	
Meis2	Necap1	
Rims2	Rufy2	
Srrm3	Slc39a10	
Golga4	Nktr	
Snw1	Kcnb1	
Nemf	Pcdh9	
Thoc2	Gabra3	
Clasp2	Chd5	
Rims1	Kcna2	
R3hdm2	Gpbp1	
Tubb4b	Bend6	
Htatsf1	Sgtb	
Ttc9b	Scrn1	
Glrx2	Cfdp1	
Cabp1	Homer1	
Nr3c1	Cobl	
Ddx1	Lingo1	
Stau2	Asap1	
Map9	Pak1ip1	
Foxp1	Dnajc21	
Wdr26	1700025G04Rik	
Sept11	A830010M20Rik	
Pacsin1	Add1	
Scn1a	Lin7a	
Cacng2	Tia1	
Ankrd11	Usp7	
Cxxc5	Gm10419	
Zfp148	Esf1	
Cep290	Epb4.1l3	
Smarcc1	Apc	
Rap2a	Camkk2	
Pip5k1c	Rcan2	

Supplementary Table 5

Slide-seq	**3-day**	**2-week**
# of genes	7576	7294
size of image	48X48	48X48
runtime (in min)	11.5	12.5

(running with 30 clusters)

(running with assigning different clusters)

2-week	10 clusters	20 clusters	30 clusters	50 clusters	75 clusters	100 clusters
runtime (in min)	8	9.5	12.5	14.5	21	29.5

CPU	Intel i9-9880H
Memory	64GB
GPU	NVIDIA Quadro T2000