Belayer: Modeling discrete and continuous spatial variation in gene expression from spatially resolved transcriptomics

Cong Ma*, Uthsav Chitra*, Shirley Zhang, and Benjamin J. Raphael†

Department of Computer Science, Princeton University

Abstract

Spatially resolved transcriptomics (SRT) technologies measure gene expression at known locations in a tissue slice, enabling the identification of spatially varying genes or cell types. Current approaches for these tasks assume either that gene expression varies continuously across a tissue or that a slice contains a small number of regions with distinct cellular composition. We propose a model for SRT data that includes both continuous and discrete spatial variation in expression, and an algorithm, Belayer, to estimate the parameters of this model from layered tissues. Belayer models gene expression as a piecewise linear function of the relative depth of a tissue layer with possible discontinuities at layer boundaries. We use conformal maps to model relative depth and derive a dynamic programming algorithm to infer layer boundaries and gene expression functions. Belayer accurately identifies tissue layers and infers biologically meaningful spatially varying genes in SRT data from brain and skin tissue samples.

Keywords: *spatially resolved transcriptomics, spatial variation, gene expression, layered tissues, segmented regression, conformal maps*

*These authors contributed equally and author order was decided by a coin toss.

†Correspondence: braphael@princeton.edu

1 Introduction

Spatially resolved transcriptomics (SRT) technologies simultaneously measure both gene expression and spatial location of cells in a two-dimensional tissue slice [12, 72, 75]. Examples of SRT technologies include in-situ hybridization (ISH) techniques such as MERFISH [22] and seqFISH+ [33], which are based on imaging fluorescence probes, and sequencing techniques such as Slide-seq [76], Slide-seqV2 [80] and the 10X Genomics Visium spatial transcriptomics platform [1, 79], which sequence barcoded mRNA molecules whose barcodes record both the spatial locations of the molecules and unique molecular identifiers (UMI) for the mRNA molecule. SRT technologies have enabled a wide range of biological analyses, including analyses on the spatial organization of different tissues [62, 11, 59, 15, 43, 32, 8, 19, 65, 30, 81, 48, 18] and inter-cellular communication [95, 20, 28, 54].

Computational analysis of SRT data presents multiple challenges. First, most SRT technologies produce high-dimensional data; for example, sequencing-based technologies such as Slide-Seq [76] and spatial transcriptomics [79] measure the expression of 10,000 to 20,000 human genes across thousands or tens of thousands of spatial locations. Second, gene expression measurements from many current SRT technologies are highly sparse, with many genes not measured even if they are expressed. For example, current sequencing-based SRT technologies have very low UMI counts per spot; e.g., in the 10X Genomics Visium platform [1] each spot has a median of about 5,000 UMI counts, while Slide-seqV2 [80] reports median UMI counts of approximately 500. Finally, jointly modeling both gene expression and the spatial location of cells in a tissue requires an appropriate model of spatial variation in gene expression, and is thus more involved than applying existing models developed for bulk or single-cell gene expression data.

Most current computational models for analyzing spatial variation in gene expression in SRT data make one of two distinct modeling assumptions. The first modeling assumption is that gene expression is determined by discrete cell types; specifically, there are a small number of clusters with distinct cell type compositions, and gene expression at a spot depends only on the cluster label, i.e. the cell type composition present at the spot. This modeling assumption is made by methods for identifying cell type clusters [99, 28, 74]. These methods implicitly address data sparsity by sharing information across nearby spots, e.g., through the use of models such as hidden Markov random fields (HMRFs). Under this modeling assumption, large differences in gene expression between clusters are allowed, which corresponds to discrete shifts in cell type composition within a tissue slice. However, methods that make this modeling assumption also assume that gene expression is constant within each cluster, and thus do not account for continuous spatial variation of gene expression within a cluster.

The second modeling assumption is that gene expression varies continuously across a tissue slice. This assumption is usually made by methods that identify spatially varying genes [83, 82, 29, 100] or that construct low-dimensional representations of measured cells [89, 86, 23]. For example, SpatialDE [83] and SPARK [82] model gene expression with a Gaussian Process in which the covariance of expression between a pair of spots decreases as their spatial distance increases. This modeling assumption is justified by the biological observation that gene expression is affected by spatial cellular environments and inter-cellular communication [53, 10, 37]. However, most methods that make this modeling assumption do not account for large discrete changes in cell type composition across the tissue and the consequent discontinuous change in expression. One exception are factor analysis approaches, such as [89, 86, 90, 23], which model gene expression as a discrete sum of continuous factors estimated from the data. These approaches can in principle model discontinuous changes in gene expression if the factors have disjoint support. However in practice these methods do not explicitly model such discontinuities. Moreover the factors learned by these methods are sometimes difficult to interpret as they are not guaranteed to correspond to distinct cell types.

We introduce a method called Belayer for analyzing SRT data using a *global* model of tissue organization and gene expression that combines both discrete and continuous spatial variation. Specifically, we define *layered tissues*, a global model of tissue organization for tissues that consist of consecutive layers of cell

types. Layered tissues are common in many organs, e.g., human skin consists of three layers [49], the cerebral cortex consists of six layers [61], and the retina has ten distinct layers of neurons [60]. In the simplest case, a layered tissue has a one-dimensional spatial structure, and we model the expression of a gene as a *piecewise continuous* function of the *depth* of the tissue layers. *Piecewise* functions allow for discontinuities in expression where there are sharp changes in cell type composition in space, such as between tissue layers, while *continuous* functions model gradients of gene expression within a tissue layer, e.g. [66, 39]. To reduce overfitting with sparse SRT data, we specifically model gene expression using *piecewise linear* functions, which are specified by a small number of parameters. The inference of piecewise linear gene expression functions in 1D is related to changepoint detection [7] and segmented regression [2, 14, 93], well-studied problems in time-series analysis whose maximum likelihood solutions can be computed using dynamic programming. We extend the classical dynamic programming algorithm for segmented regression to jointly infer piecewise linear gene expression functions for all genes simultaneously. We also demonstrate that dimensionality reduction using generalized PCA (GLM-PCA) [87] preserves the one-dimensional structure of a layered tissue, thus formalizing the ad hoc dimensionality reduction steps often made when analyzing SRT data. Next, we generalize this approach to two-dimensional layered tissues by using tools from complex analysis, namely conformal maps — complex analytic functions that locally preserve angles between curves [68] — to transform a general 2D layered tissue into a layered tissue with a one-dimensional structure. We extend our dynamic programming algorithm to more general 2D layered tissues. For tissues whose layer boundaries are lines, our DP algorithm is similar to the classical Nussinov algorithm for RNA secondary structure [69].

We implement our algorithms in a method called Belayer and we apply Belayer to simulated SRT data and to three real SRT datasets, including 10X Visium data from the human dorsolateral prefrontal cortex [62] and a mouse skin wound [34] and Slide-SeqV2 data from the mouse somatosensory cortex [80]. We demonstrate that Belayer achieves higher accuracy in clustering tissue layers compared to state-of-the-art SRT clustering methods. Moreover, we also demonstrate that the piecewise linear gene expression functions learned by Belayer enable the identification of spatially varying genes, and have higher accuracy in identifying tissue-specific marker genes compared to commonly-used methods for the identification of spatially varying genes.

2 Results

2.1 Belayer Algorithm

We introduce Belayer, an algorithm for inferring spatial patterns of gene expression in spatially resolved transcriptomics (SRT) data from *layered* tissue slices. Belayer has three defining characteristics (Figure 1). **(1)** The expression of each gene is modeled as a *piecewise linear* function of the *relative depth* within each tissue layer. **(2)** *Conformal maps*, a tool from complex analysis, are used to transform the geometry of each tissue layer to that of a vertical strip and obtain the relative depth of curved tissue layers. **(3)** A dynamic programming algorithm learns tissue layers and piecewise linear gene expression functions. We describe these characteristics in more detail below.

Belayer models the expression of each gene as a piecewise linear function of the spatial location. Specifically, suppose a two-dimensional tissue slice T consists of L layers $R_1, \ldots, R_L$ with a boundary curve Γ_ℓ between each pair $(R_\ell, R_{\ell+1})$ of adjacent layers. In the simplest case, the tissue slice T has "axis-aligned" layer boundaries Γ_ℓ, that is layer boundaries Γ_ℓ that are parallel to the y-axis (Figure 7B). In this case, we model the normalized expression $f_g(x, y)$ of each gene $g = 1, \ldots, G$ at spatial location $(x, y) \in T$ as a

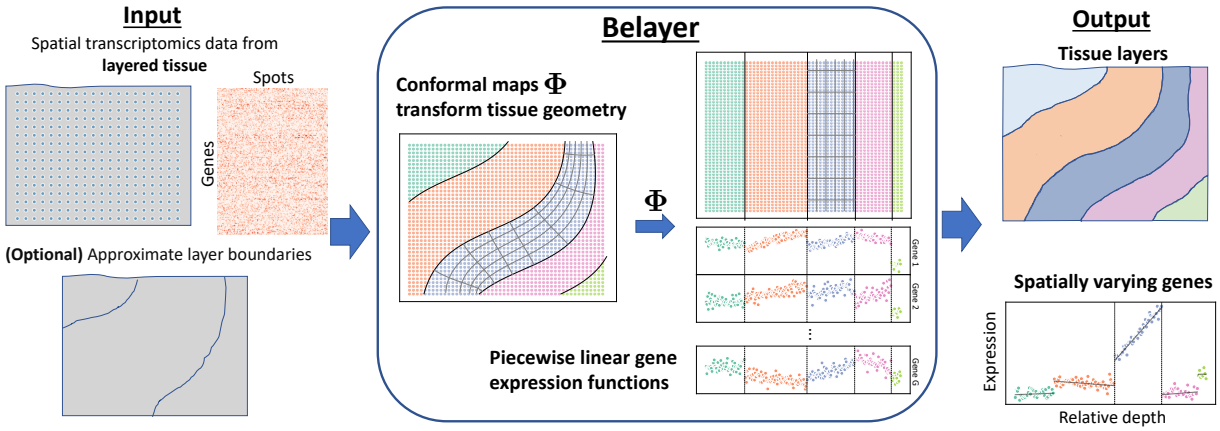

Figure 1: **(Left)** The input to Belayer is spatially resolved transcriptomics data from a layered tissue slice, and optionally approximate layer boundaries for the tissue slice. **(Middle)** Belayer uses conformal maps $\Phi = (\Phi_1, \ldots, \Phi_L)$ to transform the geometry of each tissue layer to that of a vertical strip. Belayer then models the expression of each gene in the transformed geometry as a piecewise linear function of the relative depth of the tissue layers. **(Right)** The outputs of Belayer are (1) the tissue layers, and (2) the piecewise linear gene expression functions. The latter can be used to identify spatially varying genes, such as those with large layer-specific slopes (expression gradients).

piecewise linear function of the x-coordinate:

$$f_g(x, y) = \sum_{\ell=1}^{L} \left(\alpha_{g,\ell} + \beta_{g,\ell} \cdot x\right) \cdot 1_{(x,y)\in R_\ell}, \tag{1}$$

where $\alpha_{g,\ell}$ and $\beta_{g,\ell}$ are the y-intercept and slope, respectively, of the expression of gene g in layer R_ℓ. The linear functions $a_{g,\ell} + \beta_{g,\ell} \cdot x$ model smooth gradients of expression within each layer R_ℓ, while the pieces $1_{(x,y)\in R_\ell}$ allow for discontinuities in expression at the layer boundaries Γ_ℓ. For tissues with axis-aligned layer boundaries Γ_ℓ, the x-coordinate is the depth of position (x, y) in tissue layer R_ℓ (Figure 7B).

For a tissue slice T with arbitrary layer boundaries Γ_ℓ, we generalize our model so that the expression $f_g(x, y)$ of gene g is a piecewise linear function of a quantity that we call the *relative* depth $\Phi_\ell(x, y)$ of position (x, y) in layer R_ℓ:

$$f_g(\Phi(x, y)) = \sum_{\ell=1}^{L} \left(\alpha_{g,\ell} + \beta_{g,\ell} \cdot \Phi_\ell(x, y)\right) \cdot 1_{(x,y)\in R_\ell}. \tag{2}$$

We model the relative depth $\Phi_\ell(x, y)$ in each layer R_ℓ with a *conformal map*. Conformal maps are functions from the complex plane to the complex plane that locally preserve angles between curves. Conformal maps are often used in engineering and physics applications to solve differential equations with complicated boundary conditions by transforming the domain of the differential equation to a simpler geometric structure [68, 5].

Given an $N \times G$ transcript count matrix $\mathbf{A} = [a_{i,g}]$, where $a_{i,g}$ is the count of gene g in spot i, and spatial coordinate matrix $\mathbf{S} = [\mathbf{s}_i]$ where $\mathbf{s}_i = (x_i, y_i)$ are the coordinates of spot i, Belayer aims to estimate layers $\widehat{R}_\ell$, conformal maps $\widehat{\Phi}_\ell$, and piecewise linear functions $\widehat{f}_g(x, y)$ that maximize the likelihood of the

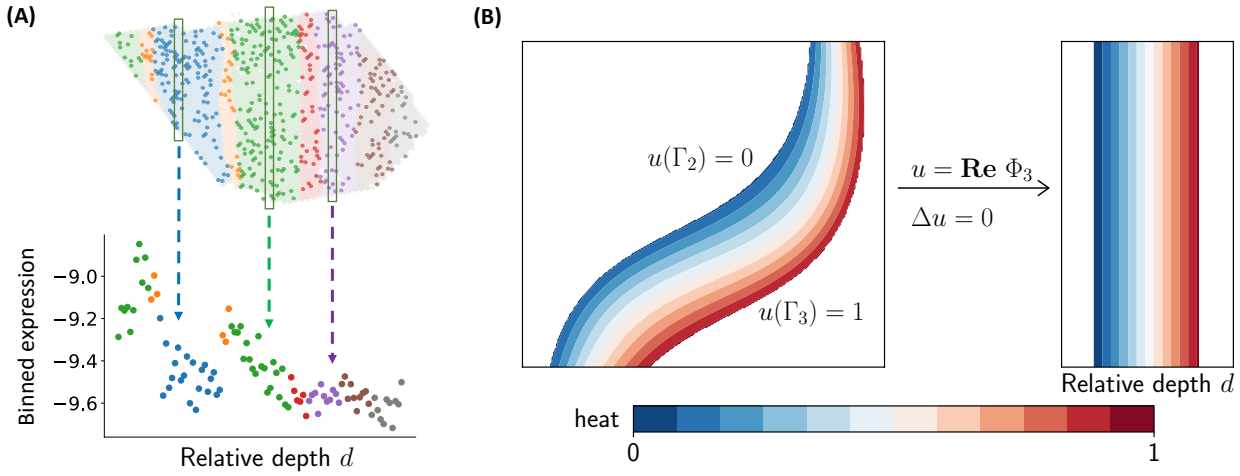

Figure 2: **(A)** (Top) Spatially resolved transcriptomics data from a layered tissue slice with non-zero UMIs indicated in darker color. (Bottom) Visualization of 1D expression values obtained from binning and normalization of UMI counts from 2D spots (x, y) with similar relative depths $\Phi_\ell(x, y)$. **(B)** The real part $u = \mathbf{Re}\,\Phi_\ell$ of a conformal map Φ_ℓ maps a curved layer to a vertical strip aligned by x-coordinate. u is a harmonic function that solves the heat equation $\Delta u = 0$. The relative depth d_i is the heat at spot i when the heat $u = 0$ is fixed at on one boundary and the heat $u = 1$ is fixed on the other boundary of the region. The color shows contours of the heat u in the layered tissue and the vertical strip.

observed SRT data $(\mathbf{A}, \mathbf{S})$:

$$\underset{\substack{\text{layers } R_1,\dots,R_L \subseteq T \\ \text{conformal maps } \Phi=(\Phi_1,\dots,\Phi_L) \\ \text{piecewise linear } f_1,\dots,f_G}}{\arg\max} \sum_{g=1}^{G} \left(\sum_{i=1}^{N} \log P\Big(a_{i,g} \mid f_g\big(\Phi(x_i, y_i)\big)\Big) \right). \tag{3}$$

The layers $\widehat{R}_\ell$ estimated by Belayer correspond to anatomical structures or other spatial partitions of the tissue slice T with distinct gene expression patterns. The estimated piecewise linear functions $\widehat{f}_g(\widehat{\Phi}(x, y))$ describe the spatial expression patterns of each gene g, and can be used to identify genes with potentially interesting spatial expression patterns, including genes with large (absolute) layer-specific slopes $|\beta_{g,\ell}|$ or genes g with large discontinuities at the boundaries $\widehat{\Gamma}_\ell$ of the estimated layers $\widehat{R}_\ell$. In order to easily visualize these spatial expression patterns, we combine expression values $a_{i,g}$ for a gene g from spots (x_i, y_i) with similar relative depths $d_i = \widehat{\Phi}_\ell(x_i, y_i)$ into a single "binned" expression value $\widetilde{a}_{j,g}$ (Figure 2A). This binning procedure approximately preserves the slopes $\beta_{g,\ell}$ and y-intercepts $\alpha_{g,\ell}$ of the piecewise linear gene expression functions $f_g(x, y)$. See Methods for more details.

In general, for arbitrarily shaped layer boundaries Γ_ℓ, computing the maximum likelihood in (3) is challenging. We provide dynamic programming algorithms to solve two special cases of this problem which provide useful approximations on real data. First, given *approximate* layer boundaries $\widetilde{\Gamma}_\ell$ of the tissue slice T, such as from histological images or prior anatomical knowledge, we estimate the real part $\mathbf{Re}\,\Phi_\ell$ of the conformal maps Φ_ℓ by solving the heat equation (Figure 2B) with boundary conditions specified by the approximate layer boundaries $\widetilde{\Gamma}_\ell$. We then derive a dynamic programming algorithm for solving (3) given the estimated real parts $\mathbf{Re}\,\widetilde{\Phi}_\ell$ of the conformal maps Φ_ℓ. Second, when the layer boundaries Γ_ℓ are *lines*, we derive another dynamic programming algorithm to compute (3) in combination with solving multiple instances of the heat equation with different boundary conditions. Our algorithm is inspired by the classical dynamic programming algorithm for segmented regression [2, 14, 93] and Nussinov's algorithm for RNA

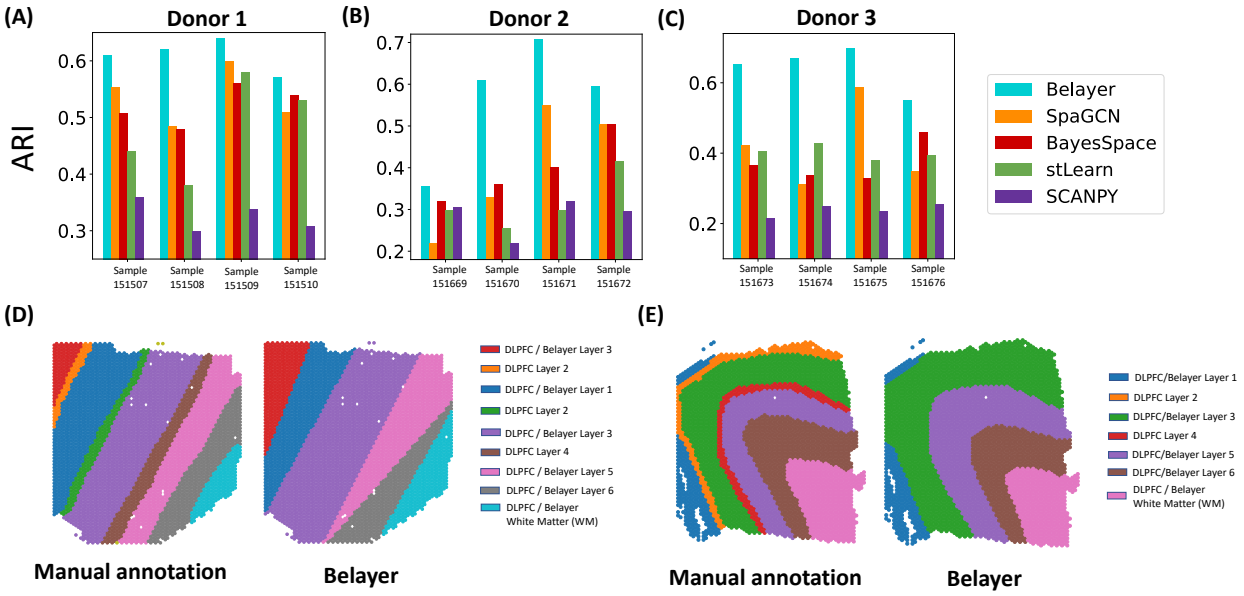

Figure 3: Comparison of Belayer, BayesSpace, stLearn, and SCANPY in identifying annotated layers in spatially resolved transcriptomics data from the dorsolateral prefrontal cortex (DLPFC) of three donors. **(A)** ARI for each method compared to manually annotated layers from each sample from Donor 1. **(B), (C)** Same as (A) for Donors 2 and 3. Since the tissue slices from these donors are not axis-aligned, Belayer finds conformal maps (Figure 2) to transform these tissue slices to axis-aligned tissue slices. **(D)** The layers identified by Belayer and the manually annotated layers for (axis-aligned) sample 151508 from Donor 1 and **(E)** sample 151673 from Donor 3. Layers identified by Belayer are labeled and colored according to the maximally overlapping layer from the manual annotation.

folding [69]. We also derive a more efficient algorithm for the special case when the layer boundaries Γ_ℓ are *parallel* lines, in which case it is not necessary to solve the heat equation. See Methods for more details on these algorithms as well as our model selection procedure for choosing the number L of layers.

2.2 Evaluation on simulated data

We evaluated Belayer on two sets of simulated SRT data. In the first simulation, we generated SRT data $(\mathbf{A}, \mathbf{S})$ from a layered tissue slice T according to our piecewise linear model (2). In the second simulation, we used the Splatter package [96] to generate realistic single-cell SRT data from a layered tissue slice. In both simulations, we found (Figure S5A and Figure S6) that Belayer had higher accuracy in identifying the tissue layers compared to two other existing methods for analysis of SRT data, BayesSpace [99] and SpaGCN [40], and compared to analysis with SCANPY [92] which does not use spatial information. In the first simulation, where we have knowledge of the true piecewise linear gene expression functions $f_g(x, y)$, we also observed (Figure S5B) that Belayer accurately estimated the parameters $\alpha_{g,\ell}, \beta_{g,\ell}$ of each piecewise linear gene expression function $f_g(x, y)$. The latter evaluation demonstrates the accuracy of using the parameters estimated by Belayer in downstream tasks such as identifying spatially varying genes. See Methods for details on both simulations.

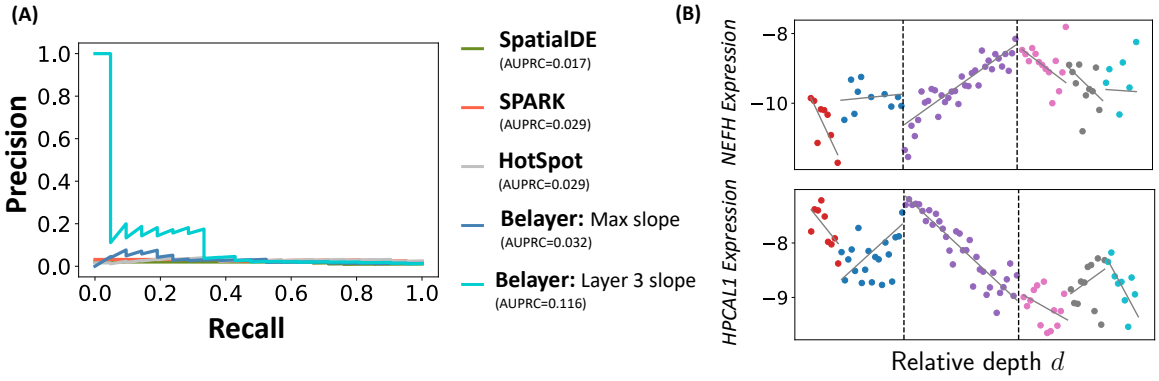

Figure 4: **(A)** Precision-recall curves for identifying marker genes in DLPFC sample 151508 using five different methods. "Belayer: Max slope" corresponds to ranking genes by $\max_{\ell=1,\ldots,L} |\widehat{\beta}_{g,l}|$ and "Belayer: Layer 3 slope" corresponds to ranking genes by $|\widehat{\beta}_{g,3}|$. **(B)** Expression functions $f_g(x)$ learned by Belayer for genes *NEFH* and *HPCAL1* that have large slopes $|\beta_{g,3}|$ in the third layer.

2.3 Dorsolateral Prefrontal Cortex

Next, we evaluated Belayer on spatially resolved transcriptomics data from the human dorsolateral prefrontal cortex (DLPFC) obtained using the 10X Visium technology [62]. This dataset consists of 12 DLPFC tissue slices from three donors. Each slice was manually annotated [62] into six layers and white matter (WM). A list of 128 marker genes for the DLPFC was obtained from [62] and previous analyses [63, 97].

2.3.1 Cortical layer identification

We first analyzed the tissue layers identified by running Belayer on the DLPFC tissue slices. For the four tissue slices from Donor 1, the manually annotated layer boundaries are approximately lines. Thus, we ran Belayer with linear layer boundaries. We compared Belayer against BayesSpace [99], stLearn [74], SCANPY [92], and SpaGCN [40] on these four tissue slices. We ran each method with the same number L of layers as Belayer (see Methods for details on our model selection procedure). We evaluated each method by computing the ARI between the manually annotated layer labels and the layers identified by the method. Note that in some samples, the manually annotated labels correspond to two layers separated in space. We counted these as separate layers and accordingly separated any clusters of BayesSpace, stLearn, or SpaGCN that are separated in space when computing ARIs. We found (Figure 3A) that Belayer has noticeably higher ARI compared to BayesSpace, stLearn, SCANPY, and SpaGCN. Our results demonstrate that Belayer learns more biologically relevant clusters by leveraging the layered structure of the tissue. In the Appendix, we also show that Belayer outperforms these methods using other evaluation metrics (Figure S9), and that Belayer outperforms BayesSpace when BayesSpace uses its own procedure for selecting the number L of layers (Table S1).

Next we compared the methods on the 8 DLPFC slices from Donors 2 and 3. The manually annotated layer boundaries of these slices are not linear. Thus, we ran Belayer with approximate layer boundaries, where we used the manually annotated layer boundaries to construct the conformal maps Φ_ℓ. We found that Belayer outperformed the other methods in the identification of layers (Figure 3B-C), suggesting that our piecewise linear assumption on the expression functions is appropriate. However, we emphasize that this comparison is overly generous to Belayer since Belayer uses information from the manually annotated layer boundaries to construct the conformal maps.

2.3.2 Identifying spatially varying genes

We also compared the genes with layer-specific expression patterns inferred by Belayer to the spatially varying genes identified by other methods. We derived a list of such layer-specific genes for DLPFC tissue slice sample "151508" (shown in Figure 3D) from the expression functions $\widehat{f}_g$ estimated by Belayer as follows. After excluding genes with low UMI counts (genes where more than 85% of spots had no UMIs), we ranked genes according to their largest layer-specific slope $\max_{\ell=1\ldots L} |\widehat{\beta}_{g,\ell}|$ across the $L = 6$ layers $\widehat{R}_1, \ldots, \widehat{R}_L$ identified by Belayer. In this way, we assign high ranks to genes with large expression gradients, as gene expression gradients are known to be associated with important biological functions in the brain [42, 51, 26]. We compared the overlap in rankings between these genes and known cortical marker genes from [62, 63, 97], and performed the same comparison for ranked lists of spatially varying genes from SpatialDE [83], SPARK [82], and HotSpot [27], three methods for identifying spatially varying genes in spatially resolved transcriptomics data. We found that Belayer achieved higher AUPRC (0.032) compared to ranking genes according to the p-values of spatial variation computed by SpatialDE (0.017), SPARK (0.029), and HotSpot (0.029) (Figure 4A). We emphasize that all methods have low AUPRC due to the many inherent challenges of marker gene identification in SRT data. These challenges include (1) the sparsity of SRT data, (2) that the list of known marker genes are curated from multiple samples and datasets while a specific SRT sample may have variation from the "consensus", and (3) that the list of marker genes is an incomplete representation of all genes that distinguish cortical layers.

We also observed that some layers are more predictive of marker genes than others. For example, ranking genes by the magnitude of their slope $|\widehat{\beta}_{g,3}|$ in the third layer identified by Belayer resulted in a much larger AUPRC (0.116) compared to scoring genes by their maximum slope $\max_{\ell=1,\ldots,L} |\widehat{\beta}_{g,\ell}|$ across all layers (Figure 4A). The genes g with large slope $|\widehat{\beta}_{g,3}|$ in the third layer are also biologically interesting, and we highlight two specific genes in Figure 4B. *NEFM* — the gene with the largest slope $|\widehat{\beta}_{g,3}|$ in the third layer — is a known cortical marker gene and is also a biomarker for neuronal damage [45]. On the other hand, *HPCAL1* — the gene with the fourth largest slope $|\widehat{\beta}_{g,3}|$ in the third layer — is not an annotated cortical marker gene but is reported to be involved in neuronal signalling [98, 91]. Our results demonstrate that incorporating layer-specific variation is important for identifying spatially varying genes, and suggest that the slopes $|\widehat{\beta}_{g,\ell}|$ identified by Belayer are a useful criteria for identifying cortical marker genes.

We also note that the third cortical layer identified by Belayer is a combination of two manually annotated layers. Ranking genes g by their estimated slope in either one of the two manually annotated layers has smaller AUPRC compared to ranking genes by their slope in the third layer identified by Belayer. This suggests that while the layers identified by Belayer do not exactly match the manually annotated layers, they potentially correspond to other biologically relevant partitions of the tissue.

2.4 Mouse skin dataset during wound healing

Next, we analyzed SRT data from a mouse skin wound obtained using the 10X Visium technology [34]. Foster et al. [34] manually annotated the spots in this dataset into one of the three layers of skin: epidermis, dermis, and hypodermis. We use the sample corresponding to postoperative day 14 which contains the largest number of spots.

We evaluated Belayer's ability to identify the 3 manually annotated layers of the skin. We used the left and right tissue boundaries as two approximate layer boundaries to estimate the conformal maps $\mathbf{\Phi}$ (Figure S13A) and ran Belayer with $L = 3$ layers as determined by the model selection precedure (Figure S7). We compared the $L = 3$ layers estimated by Belayer with the $L = 3$ clusters of BayesSpace, stLearn, SCANPY, and SpaGCN. We observe (Figure 5A) that stLearn and Belayer achieve a higher ARI compared to other methods. Moreover, while stLearn obtains a slightly higher ARI than Belayer ARI = 0.564 for stLearn, ARI = 0.520 for Belayer), stLearn predicts a discontinuous epidermal layer, which is inconsistent with

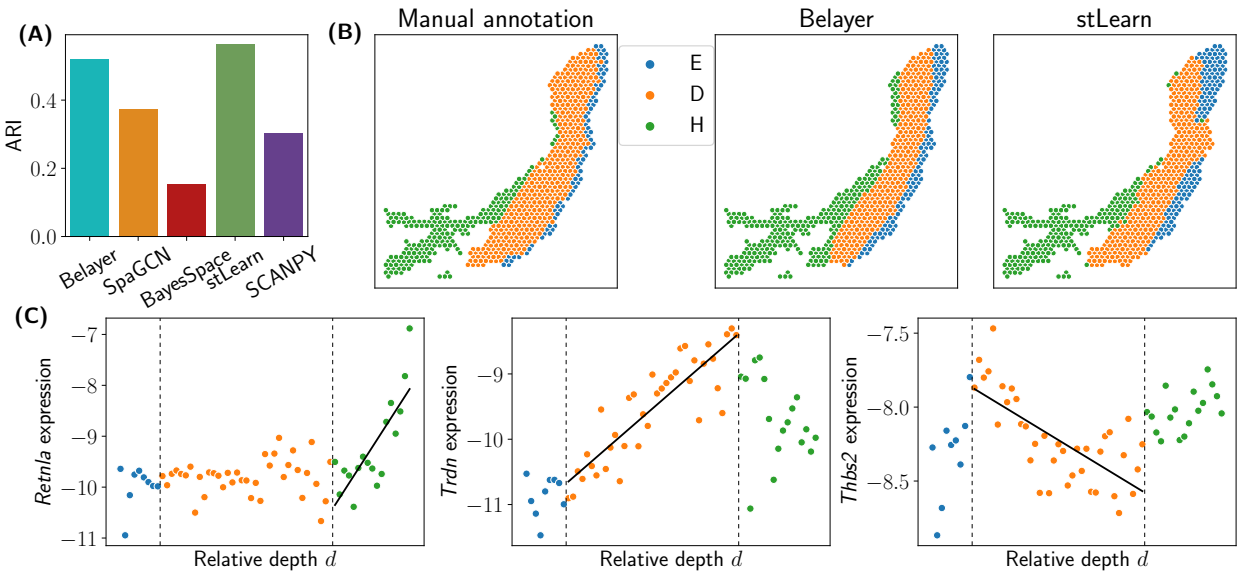

Figure 5: (A) Comparison of Belayer, BayesSpace, stLearn, SCANPY, and SpaGCN in identifying skin layers in a mouse skin Visium sample. (B) Manually annotated layers and layers identified by Belayer and stLearn. In the legend of manual annotation, "E" indicates epidermis, "D" indicates dermis, and "H" indicates hypodermis. (C) Expression functions learned by Belayer for three genes with large slopes ($|\widehat{\beta}_{g,l}|$) Dashed vertical lines indicate layer boundaries identified by Belayer. Solid lines show the expression function for the layer where the gene has a large slope.

the manual annotation (Figure 5B). By leveraging the layered geometry of the tissue, Belayer accurately identifies biologically meaningful skin layers.

We also analyze the genes g that Belayer infers to have large positive or negative slopes $|\beta_{g,\ell}|$ (Figure 5B). We find that Belayer recapitulates some skin wound marker genes reported in [34]. These include: *Thbs2*, a gene involved in mouse cutaneous wound healing [3] that has the 8th most negative slope $-\widehat{\beta}_{g,2}$ in the dermis layer (Figure 5C), and *Fn1*, a marker gene for a fibroblast cluster [34] that has the 11th largest slope $\widehat{\beta}_{g,3}$ in the hypodermis layer.

Belayer also identifies genes related to the wound healing process which were not *reported in* [34] (Figure 5C). For example, *Retnla* has the largest positive slope $\widehat{\beta}_{g,3}$ in the hypodermis layer, and it encodes a protein that is known to be an effector molecule in the skin wound healing process [46]. *Trdn* has the largest positive slope $\widehat{\beta}_{g,2}$ in the dermis layer, and it encodes an integral transmembrane protein involved in muscle contraction [70], a healing response for skin wounds.

2.5 Somatosensory Cortex

We evaluated Belayer on SRT data for a single slice of the somatosensory cortex [19] obtained using the Slide-SeqV2 technology [80]. The somatosensory cortex is a part of the neocortex and consists of six layers [44]. We obtained a list of 30 marker genes for the somatosensory cortex from [25, 64] that are also measured in this dataset.

We ran Belayer with two approximate layer boundaries, the top and bottom boundaries of the tissue slice which were chosen from visual inspection of the RCTD layer annotations [19] (Figure S12). We compared the $L = 5$ layers identified by Belayer to the clusters identified by SpaGCN [40], SCANPY, and stLearn. We evaluated the accuracy of each method according to the cell types annotated by RCTD [19], which integrated the Slide-SeqV2 data with a reference scRNA-seq dataset. We do not compare to BayesSpace [99], since the

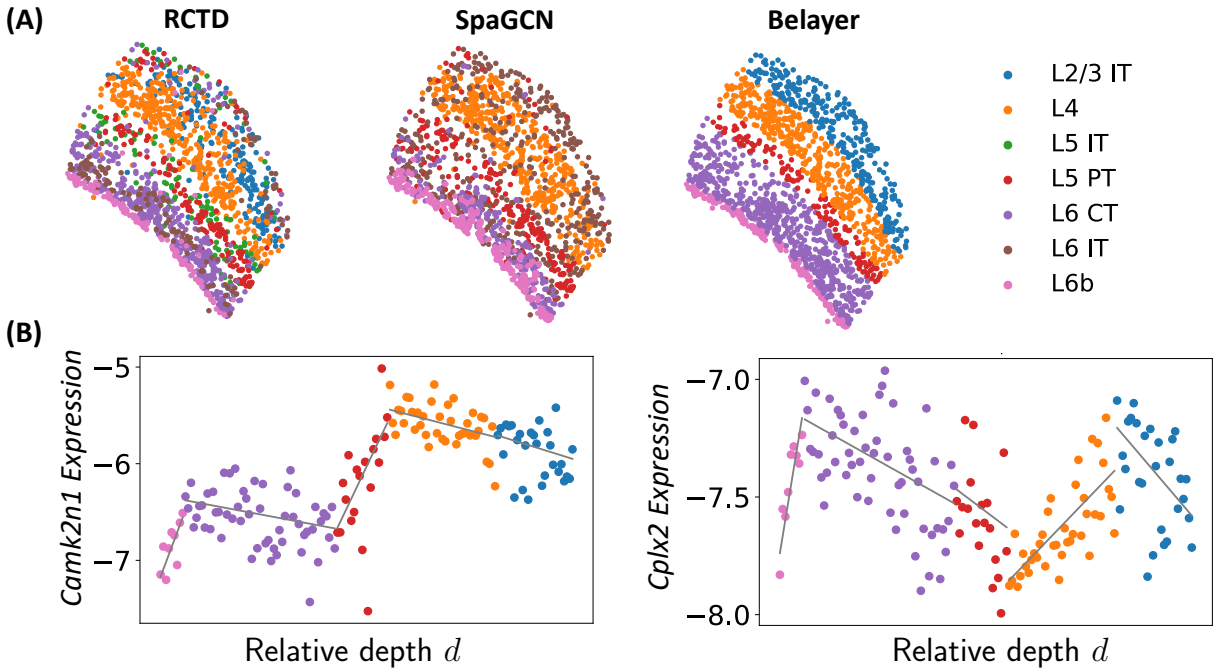

Figure 6: (A) Layers identified by RCTD, SpaGCN, and Belayer in a somatosensory cortex Slide-SeqV2 sample. The layers identified by SpaGCN and Belayer are labeled and colored according to the maximally overlapping layer from the RCTD cell type annotations. (B) Expression function learned by Belayer for genes g with large slopes $|\widehat{\beta}_{g,\ell}|$.

current implementation of this method requires SRT data obtained from the ST or 10X Visium platforms. We find that compared to SpaGCN, Belayer distinctly identifies the layers of the somatosensory cortex (Figure 6A). We also observe that the layers identified Belayer are more similar to the RCTD cell type annotations than the clusters identified by SpaGCN, stLearn, and SCANPY (Figure S11A). Moreover, compared to the RCTD cell type annotations — which are obtained through a reference scRNA-seq dataset with cell type annotations — we find that the layers identified by Belayer correspond more closely to the distinct layers of the somatosensory cortex. In particular, Belayer is able to clearly identify the L2/3 layer (Figure 6A) while RCTD models this layer as a mixture of different cell types from layers 2 through 6, which is biologically inconsistent with the layered geometry of the somatosensory cortex [44]. These results demonstrate that Belayer more accurately learns tissue layers by utilizing a global model of layered tissues.

We also compared the genes with layer-specific expression patterns inferred by Belayer to the spatially varying genes identified by SpatialDE [83], SPARK [82], and HotSpot [27] following the same procedure as in Section 2.3.2. We found (Figure S11B) that Belayer achieved higher AUPRC (0.06) compared to ranking genes by the p-values of spatial variation computed by SpatialDE (0.024) and SPARK (0.049), and had comparable AUPRC compared to HotSpot (0.061). Moreover, if we rank genes according to the magnitude of their slope $|\widehat{\beta}_{g,6}|$ in the layer 2/3 intratelencephalic region estimated by Belayer ("L2/3 IT" in Figure 6A), then Belayer has larger AUPRC (0.099) compared to SpatialDE, SPARK, and HotSpot. As in Section 2.3.2, we emphasize that all methods have low AUPRC because of the challenges in identifying marker genes in SRT data. In particular, the list of marker genes for the somatosensory cortex that we use for evaluation is most likely incomplete as it contains less than 30 genes, while more than 1,000 genes are estimated to be involved in neuronal functions in the cerebral cortex [97].

We highlight two genes with large layer-specific slope $|\widehat{\beta}_{g,\ell}|$ that are not in the list of marker genes but are biologically interesting (Figure 6B). *Camk2n1* has the fourth-largest layer-specific slope, and is reported

to have spatiotemporal patterns of regulation [56]. *Cplx2* has the tenth-largest layer-specific slope, and mutations in this gene are reported to contribute to cognitive dysfunction in schizophrenia [38]. Our results demonstrate that the spatial patterns identified by Belayer are important in identifying biologically relevant genes.

3 Discussion

We introduce a new method, Belayer, to analyze spatial variation in gene expression from spatially resolved transcriptomics (SRT) data from *layered tissues*. Belayer models the expression of each gene with a *piecewise linear function* of the relative depth of the tissue layers. This piecewise linear model allows for both discrete changes in expression between layers – e.g., due to changes in cell type compositions – as well as continuous variation in gene expression within layers – e.g., due to gradients of gene expression. In the simplest case of an axis-aligned tissue structure, we infer the maximum likelihood expression function using a dynamic programming algorithm that is related to the classical problems of changepoint detection [7] and segmented regression [2, 14, 93]. We extend our approach to arbitrary layered tissues using the theory of conformal maps [68] from complex analysis [4], which are related to harmonic functions – and more specifically the heat equation – and are often used to solve partial differential equations in engineering applications. We provide algorithms to solve two important special cases of the general problem for linear layer boundaries and pre-specified approximate layer boundaries.

We show that Belayer outperforms existing approaches in identifying the tissue layers and spatially varying genes in both simulated SRT data and three real SRT datasets: 10X Visium data from the human dorsolateral prefrontal cortex (DLPFC) [62], Slide-SeqV2 data from the mouse somatosensory cortex [80], and 10X Visium data from a mouse skin wound [34]. Additionally, Belayer discovers genes with layer-specific and continuous spatial variation in expression that correspond both to known tissue-specific marker genes and genes with potentially novel tissue-specific functions. These results demonstrate that our piecewise linear model is a reasonable approach for the identification of tissue layers and the discovery of layer-specific marker genes.

There are a number of directions for future investigation. The first direction is to further investigate the expression functions learned by Belayer. For example, the observation that the slopes of the expression functions learned by Belayer recovers known tissue-specific marker genes better than existing methods inspires further study of the novel genes that are identified by our model, including genes with discontinuities and changes in slope at layer boundaries. It would also be of interest to relate the slopes of the gene expression functions to biological quantities such as the cell type proportion at different spots, which could be estimated using SRT deconvolution methods [19, 78, 9, 47, 31]. Second, it would be desirable to provide an algorithm to solve (or approximately solve) the maximum likelihood problem in (3) for arbitrary layer boundaries. One possible approach is to extend the dynamic programming algorithm for lines to a larger class of layer boundaries. Third, it would helpful to further minimize overfitting by incorporating regularization and rigorous statistical testing into our algorithm; e.g., using the Chow test [24] for change-point detection. Such extensions might also allow for nonlinear piecewise continuous expression functions, assuming the data has sufficient spatial resolution. The fourth direction is to extend the definition of a layered tissue to account for more complex tissue geometries such as muscle tissues with concentric or striated layer structures or 3-D tissues [6]. While the heat equation can be solved over closed domains or in three dimensions, deriving an appropriate representation of piecewise constant/linear expression functions for more complicated geometries requires further work. In addition, the problem of inferring the layers and the conformal map from more complicated geometries may not be straightforward. We also note that while some tissues may not have a layered structure, they may be subdivided into layered regions. For example, the mouse neocortex contains a layered somatosensory cortex, which are segmented using prior biological knowledge [19]. It

would be useful to extend our definition of layered tissues towards these "piecewise" layered tissues as well as to systematically identify these layered tissue regions, perhaps by first coarsely subdividing a tissue using other SRT analysis methods. Finally, it would be interesting to extend our approach to model other spatially resolved data, including SRT data obtained from ISH technologies [57, 73] and spatial proteomics [58].

Methods

4 Layered tissues and expression functions

Modeling spatial patterns of gene expression is complicated by the large variability in the spatial structure of tissues — e.g. some parts of brain tissues have a layered structure while muscle tissues have a striated structure [6] — as well as the sparsity of the gene expression profiles produced by current spatially resolved transcriptomics (SRT) technologies. To avoid overfitting the data, it is helpful to make simplifying assumptions about the spatial structure of the tissue slice T and/or the spatial patterns of gene expression. Here, motivated by the layered structure of the skin, brain, eyes, and other organs [6], we focus on *layered tissue slices* (Figure 7A) which we define as follows.

Definition 1. *An* L-layered tissue slice *is a region* $T \subseteq \mathbb{R}^2$ *containing* $L-1$ *non-intersecting smooth curves* $\Gamma_1, \ldots, \Gamma_{L-1}$*, or* layer boundaries*, satisfying:*

1. *each curve* Γ_ℓ *has end points on the boundary* ∂T *of* T*;*
2. *every point* $p \in T$ *is contained in a region* R *bounded by* ∂T *and* at most two *curves* Γ_ℓ *and* $\Gamma_{\ell'}$*.*

The $L-1$ layer boundaries $\Gamma_1, \ldots, \Gamma_{L-1}$ partition the tissue slice T into L regions $R_1, \ldots, R_L$, or *layers*. The layers $R_1, \ldots, R_L$ of a L-layered tissue slice T represent biologically distinct regions in the tissue slice T. For example, in a tissue slice T from the skin, R_1, R_2, R_3 may represent the epidermis, dermis, and hypodermis layers, where each layer consists of unique cell types and has unique functions [49]. Without loss of generality we assume that the layers $R_1, \ldots, R_L$ and layer boundaries $\Gamma_1, \ldots, \Gamma_{L-1}$ are ordered so that layer R_1 is bounded by tissue boundary ∂T and layer boundary Γ_1, layer R_ℓ is bounded by layer boundaries $\Gamma_{\ell-1}$ and Γ_ℓ for $\ell = 2, \ldots, L-1$, and layer R_L is bounded by layer boundary Γ_{L-1} and tissue boundary ∂T.

We describe the spatial distribution of the expression of a gene g in a tissue T with an *expression function* $f_g : T \to \mathbb{R}$, where $f_g(x, y)$ measures the expression of gene g at spatial location $(x, y) \in T$. For example, a gene g whose expression is uniform across the tissue has a constant expression function $f_g(x, y) = c$, while a marker gene g for a specific region $R \subseteq T$ could have the expression function $f_g(x, y) = c \cdot 1_{\{(x,y)\in R\}} + c' \cdot 1_{\{(x,y)\notin R\}}$.

More generally, if T is a L-layered tissue slice with layers $R_1, \ldots, R_L$ and the expression of gene g in layer R_ℓ is given by layer-specific expression function $f_{g,\ell}(x, y)$, then gene g has the expression function $f_g(x, y) = \sum_{\ell=1}^{L} f_{g,\ell}(x, y) \cdot 1_{\{(x,y)\in R_\ell\}}$. We assume that the layer-specific expression functions $f_{g,\ell}(x, y)$ are continuous functions, and thus $f_g(x, y)$ is a piecewise-continuous function with discontinuities allowed at the layer boundaries Γ_ℓ. These discontinuities correspond to differences in expression due to changes in cell type composition between different layers R_ℓ of the tissue T. In the next section, we describe how one models SRT data using the expression functions $f_g(x, y)$.

5 Axis-aligned layered tissues

We begin by studying the simplest L-layered tissue slice: the *axis-aligned* L-layered tissue slice, where each layer boundary Γ_ℓ is a line $x = b_\ell$ parallel to the y-axis (Figure 7B). We assume that the expression of a gene g at position (x, y) depends only on the *layer depth*, or the distance from (x, y) to the nearest layer boundaries $x = b_\ell$. Under this assumption, the layer-specific expression functions $f_{g,\ell}(x, y)$ are functions of only the x-coordinate: $f_{g,\ell}(x, y) = f_{g,\ell}(x)$. Thus, the expression function $f_g(x, y)$ also only depends on the x-coordinate, i.e. $f_g(x) = \sum_{\ell=1}^{L} f_{g,\ell}(x) \cdot 1_{\{b_{\ell-1}<x\leq b_\ell\}}$ where for convenience we define $b_0 = -\infty$ and $b_L = \infty$. We call each b_ℓ a *breakpoint* of the expression function f_g.

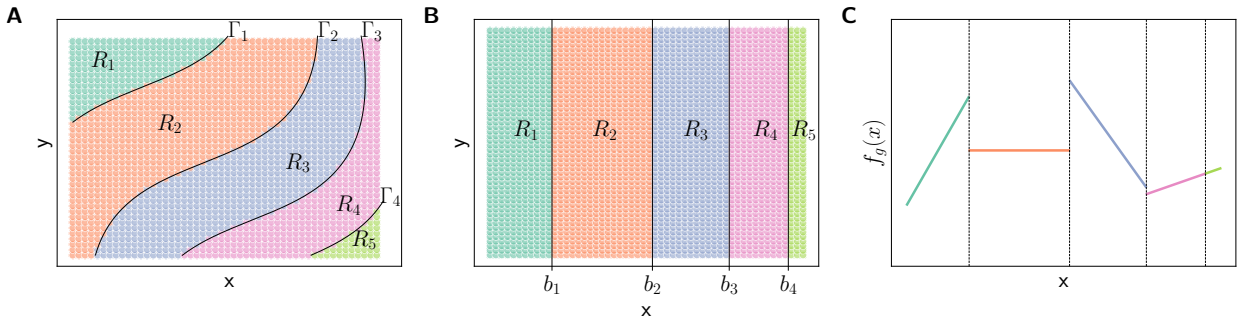

Figure 7: Layered tissue slices and piecewise linear expression functions. **(A)** A 5-layered tissue slice T with layers $R_1, \ldots, R_5$ and layer boundaries $\Gamma_1, \ldots, \Gamma_4$. **(B)** An axis-aligned 5-layered tissue slice with layers $R_1, \ldots, R_5$ and layer boundaries $x = b_1, \ldots, x = b_4$. **(C)** Expression function $f_g(x)$ for gene g along the x coordinate for the axis-aligned layered tissue slice in (B). For all panels, colors indicate layers and black lines indicate layer boundaries.

The simplest model for $f_g(x)$ is a piecewise constant function, which corresponds to each gene g having a constant expression in each layer R_ℓ. Such a piecewise constant model is implicit in methods that assume constant expression in contiguous regions of cell types; e.g. methods that use hidden Markov random field (HMRF) models [99, 94, 28, 85]. We generalize these approaches by modeling continuous variation in expression within a layer; e.g. due to gradients of gene expression [66, 39, 17, 21]. Since current SRT technologies have limited dynamic range and spatial resolution, inference of complicated expression functions may be prone to overfitting. Thus, we use the simplest form of continuous, non-constant spatial variation within each layer and model each layer-wise expression function $f_{g,\ell}(x)$ as a *linear function*. Then each expression function $f_g(x)$ is a piecewise linear function (Figure 7C) with shared breakpoints $b_1, \ldots, b_{L-1}$ across all genes corresponding to the axis-aligned layer boundaries $\Gamma_1, \ldots, \Gamma_{L-1}$, i.e.

$$f_g(x) = \sum_{\ell=1}^{L} (\beta_{g,\ell} x + \alpha_{g,\ell}) \cdot 1_{\{b_{\ell-1} < x \leq b_\ell\}}, \tag{4}$$

where $\beta_{g,\ell}$ and $\alpha_{g,\ell}$ are the slope and y-intercept, respectively, of gene g in layer R_ℓ. We define $\mathcal{L}(b_1, \ldots, b_{L-1})$ to be the set of piecewise linear functions $f(x)$ with breakpoints $b_1, \ldots, b_{L-1}$, and we define $\mathcal{L}$ to be the set of linear functions $f(x)$.

We contrast our model for the expression functions f_g with another commonly used model for expression functions, Gaussian Processes (GPs) [83, 82, 86, 55]. In this approach each expression function $f_g(x, y) \sim \mathcal{GP}\big(0, k(\mathbf{s}, \mathbf{s}')\big)$ is an independent sample from a GP with mean function 0 and covariance function $k(\mathbf{s}, \mathbf{s}')$ between spatial locations $\mathbf{s} = (x, y)$ and $\mathbf{s}' = (x', y')$. The covariance functions $k(\mathbf{s}, \mathbf{s}')$ used by existing methods, including SpatialDE [83] and SPARK [82], generate continuous functions and thus do not model piecewise continuous expression functions $f_g(x)$. We show in the Appendix that it is possible to model piecewise linear functions with GPs using a one-dimensional blockwise covariance function; however, to our knowledge such covariance functions have not been used to model SRT data.

5.1 Axis-Aligned L-Layered Problem

We aim to infer the layer boundaries $x = b_\ell$ and expression functions $f_g(x)$ that maximize the likelihood of the observed SRT data. We represent SRT data as an expression matrix $\mathbf{A} = [a_{i,g}] \in \mathbb{R}^{N \times G}$ and a spatial location matrix $\mathbf{S} \in \mathbb{R}^{2 \times N}$ with column $\mathbf{s}_i = (x_i, y_i)$ indicating the spatial location of spot i. We define the Axis-Aligned L-Layered Problem as the the following maximum likelihood estimation problem.

Axis-Aligned L-Layered Problem. *Given SRT data $(\mathbf{A}, \mathbf{S})$ and a number L of layers, find layer boundaries $x = b_1, \ldots, x = b_{L-1}$ and piecewise linear expression functions $f_1, \ldots, f_G \in \mathcal{L}(b_1, \ldots, b_{L-1})$ that maximize the log-likelihood of the data:*

$$\underset{\substack{b_1<b_2<\cdots<b_{L-1} \\ f_1,\ldots,f_G\in\mathcal{L}(b_1,\ldots,b_{L-1})}}{\arg\max} \sum_{g=1}^{G}\left(\sum_{i=1}^{N}\log P\big(a_{i,g} \mid f_g(x_i)\big)\right). \tag{5}$$

When there is $L = 1$ layer, each expression function $f_g(x)$ is a linear function, and thus the maximum log-likelihood (5) in the Axis-Aligned L-Layered Problem reduces to

$$\max_{f_1,\ldots,f_G\in\mathcal{L}} \sum_{g=1}^{G}\left(\sum_{i=1}^{N}\log P\big(a_{i,g} \mid f_g(x_i)\big)\right) = \sum_{g=1}^{G}\max_{f_g\in\mathcal{L}}\left(\sum_{i=1}^{N}\log P\big(a_{i,g} \mid f_g(x_i)\big)\right). \tag{6}$$

The maximization on the right-hand side of (6) is a regression problem of finding the linear function f_g that best fits the observed data. Thus, (6) is solved by computing a separate regression for each gene $g = 1, \ldots, G$.

More generally, when each expression function $f_g(x) = \sum_{\ell=1}^{L} f_{g,\ell}(x) \cdot 1_{\{b_{\ell-1}<x\le b_\ell\}}$ is an L-piecewise linear function with *known* breakpoints $b_1, \ldots, b_{L-1}$, then the maximum log-likelihood (5) in the Axis-Aligned L-Layered Problem reduces to

$$\max_{f_1,\ldots,f_G\in\mathcal{L}(b_1,\ldots,b_{L-1})} \sum_{g=1}^{G}\left(\sum_{i=1}^{N}\log P\big(a_{i,g} \mid f_g(x_i)\big)\right) = \sum_{g=1}^{G}\sum_{\ell=1}^{L}\left(\max_{f_{g,\ell}\in\mathcal{L}}\sum_{i:b_{\ell-1}<x_i\le b_\ell}\log P\big(a_{i,g} \mid f_{g,\ell}(x_i)\big)\right), \tag{7}$$

Then, each expression function f_g is computed by solving L separate regression problems, one regression problem for each pair $x = b_{\ell-1}, x = b_\ell$ of consecutive layer boundaries. On the other hand, when the breakpoints $b_1, \ldots, b_{L-1}$ of the piecewise linear expression functions f_g are *unknown*, one will have to compute a regression over *all* pairs of possible layer boundaries for each gene g, as we describe in the next section.

We emphasize that the regression approach described above can be used with different probability distributions $P\big(a_{i,g} \mid f_g(x_i)\big)$ for the observed counts $a_{i,g}$. In particular, following [87, 77, 71] we model UMI counts $a_{i,g}$ with a Poisson distribution $a_{i,g} \overset{\text{i.i.d.}}{\sim} \text{Poisson}\Big(C_i \cdot \exp\big(f_g(x_i)\big)\Big)$ where C_i is the total UMI count at spot i. Using the Poisson model we solve the regression problems in (7) with Poisson regression. Another alternative is to model normalized expression values $a_{i,g}$, with a Gaussian distribution $a_{i,g} \overset{\text{i.i.d.}}{\sim} N(f_g(x_i), \sigma^2)$ where σ^2 is a shared variance parameter. Under the Gaussian model, the regression problems in (7) are linear regressions.

Visualization and binning of SRT data. To simplify the visualization of SRT data $(\mathbf{A}, \mathbf{S})$ from an axis-aligned tissue slice, we combine expression values for a gene from spots with the same x-coordinate into a single "binned" expression value (Figure 2A). Specifically, for each gene g we construct a binned expression value $\widetilde{a}_{j,g}$ that estimates the gene expression for all spots with the same x-coordinate $\widetilde{x}_j$. For the Poisson model the binned expression $\widetilde{a}_{j,g} = \sum_{i:x_i=\widetilde{x}_j} a_{i,g}$ is the sum of the UMI counts $a_{i,g}$ at spots $\mathbf{s}_i = (x_i, y_i)$ with x-coordinate $x_i = \widetilde{x}_j$, while for the Gaussian model the binned expression $\widetilde{a}_{j,g} = \frac{1}{|\{i:x_i=\widetilde{x}_j\}|}\sum_{i:x_i=\widetilde{x}_j} a_{i,g}$ is the average expression. We show in the Appendix that binning the data does not affect inference of the expression functions $f_1, \ldots, f_G$ in the Axis-Aligned L-Layered Problem, as the expression functions obtained by maximizing the log-likelihood (5) with the binned expression $[\widetilde{a}_{j,g}]$ are equal to the expression functions obtained by maximizing (5) with the original expression $[a_{i,g}]$.

In the applications presented below, it is often the case that there are spots $\mathbf{s}_i = (x_i, y_i)$ with *approximately* equal x-coordinates x_i. Thus, we extend the binning approach to construct binned expression $\widetilde{a}_{j,g}$ from spots $\mathbf{s}_i = (x_i, y_i)$ with x-coordinates $x_i \approx \widetilde{x}_j$. Further, for UMI count data we plot the *normalized* binned expression $\log(\widetilde{a}_{i,g}/\widetilde{C}_i)$, where $\widetilde{C}_i$ is the sum of the total UMI counts C_i for all spots i in bin j (Figure 2A). These normalized counts have the same scale as the expression functions f_g. See the Appendix for more details.

5.2 A dynamic programming algorithm for the Axis-Aligned L-Layered Problem

When there is only one gene, i.e. $G = 1$, then the Axis-Aligned L-Layered Problem is also known as *segmented regression*. Segmented regression is a classical problem in statistics and time-series analysis and has a well-known dynamic programming (DP) solution [13, 2, 14, 93]. Here, we present a DP algorithm for the Axis-Aligned L-Layered Problem that extends the classical DP algorithm for segmented regression [2, 14, 93] to any number G of genes.

Briefly, the DP algorithm is as follows. Let $M_{n,\ell}$ be the maximum log-likelihood of the first n datapoints fit to ℓ-piecewise linear expression functions $f_1, \ldots, f_G$, i.e.

$$M_{n,\ell} = \max_{\substack{b_1 < b_2 < \cdots < b_{\ell-1} \\ f_1, \ldots, f_G \in \mathcal{L}(b_1, \ldots, b_{\ell-1})}} \sum_{g=1}^{G} \left(\sum_{i=1}^{n} \log P\big(a_{i,g} \mid f_g(x_i)\big) \right), \tag{8}$$

where we assume without loss of generality that the spots are ordered by x-coordinate so that $x_1 \leq x_2 \leq \cdots \leq x_N$.

The best piecewise linear fit for the first n datapoints with ℓ pieces corresponds to: (1) the best piecewise linear fit for the first n' datapoints with $\ell - 1$ pieces and (2) the best linear function fit for the remaining $n - n'$ datapoints, for some $n' < n$. This yields the following recurrence:

$$M_{n,\ell} = \max_{n' < n} \left[M_{n',\ell-1} + \sum_{g=1}^{G} \left(\max_{f_g \in \mathcal{L}} \sum_{i=n'}^{n} \log P\big(a_{i,g} \mid f_g(x_i)\big) \right) \right], \tag{9}$$

where the inner maximization is solved using regression.

The DP algorithm consists of using the recursion (9) to fill in a DP table column-by-column followed by a pass backwards through the table to identify the L-piecewise linear expression functions $f_1, \ldots, f_G$ and breakpoints $b_1, \ldots, b_{L-1}$. The run-time of the DP algorithm is upper-bounded as $O(LN^2G \cdot P_0)$, where P_0 is the runtime to solve the inner maximization in (9) for a single gene g. For the case where the gene expression values $a_{i,g}$ follow the Gaussian model, the run-time can be shortened to $O(LN^2G)$ by using linear algebra techniques from [2]. When N is large, it is also possible to reduce the run-time of the dynamic programming algorithm – at the expense of spatial resolution – by restricting the coordinates n', n in the recurrence (9) to a subsequence of the N data points. In practice, in our analyses, we observe marginal loss in spatial resolution when restricting to subsequences of size 150 or larger.

6 Modeling L-layered tissues with conformal maps

For an arbitrary L-layered tissue slice with layer boundaries $\Gamma_1, \ldots \Gamma_{L-1}$, we similarly assume that the expression of a gene g depends only on the layer depth, i.e., the distance to the layer boundaries Γ_ℓ. However, because the layer boundaries Γ_ℓ are not necessarily axis-aligned, the depth cannot be immediately computed. An elegant solution is obtained by using a conformal map, a tool from complex analysis [4]. A conformal map $\Phi : \mathbb{C} \to \mathbb{C}$ is a complex function that locally preserves angles; equivalently, Φ is conformal if it

is analytic and has non-zero derivative everywhere. Note that the inverse Φ^{-1} of a conformal map Φ is also conformal. Conformal maps are used to solve differential equations in the plane $\mathbb{R}^2$ with complicated boundary conditions – e.g. heat flow in a plate or airflow around a wing – by identifying $\mathbb{R}^2$ with the complex plane $\mathbb{C}$ [68].

Here, we use conformal maps to transform a layered tissue slice T into an axis-aligned layered tissue slice T'. Ideally, we would derive a conformal map $\Phi : T \to T'$ that transforms tissue slice T to axis-aligned tissue slice T', with the constraints that Φ maps each layer boundary Γ_ℓ in T to the corresponding layer boundary $x = b_\ell$ in axis-aligned tissue slice T'. Unfortunately, constructing a conformal map satisfying these additional contraints is challenging, as the standard constructions of conformal maps only allow constraints on the boundary of T [68]. In general, it is unclear whether a conformal map satisfying additional constraints on non-boundary curves always exists.

As an alternative, we derive L conformal maps $\Phi_\ell : R_\ell \to R'_\ell$ for $\ell = 1, \ldots, L$, where Φ_ℓ transforms layer R_ℓ of tissue slice T to the corresponding layer R'_ℓ of axis-aligned tissue slice T'. We require that each conformal map Φ_ℓ maps the layer boundaries $\Gamma_{\ell-1}, \Gamma_\ell$ of layer R_ℓ to the corresponding layer boundaries $x = b_{\ell-1}, x = b_\ell$ of layer R'_ℓ, respectively, which is equivalent to the following constraints on the conformal map Φ_ℓ:

$$\mathbf{Re}\,\Phi_\ell(\Gamma_{\ell-1}) = b_{\ell-1} \text{ and } \mathbf{Re}\,\Phi_\ell(\Gamma_\ell) = b_\ell, \tag{10}$$

where $\mathbf{Re}$ denotes the real part of a complex number. We define $\mathcal{C}_T(b_{\ell-1}, b_\ell)$ as the set of conformal maps Φ_ℓ that satisfy (10). Notice that (10) are constraints on the boundary of each layer R_ℓ, and thus such maps are guaranteed to exist. Then, the map $\Phi : T \to T'$ from tissue slice T to axis-aligned tissue slice T' is given by the piecewise sum of the L conformal maps $\Phi_1, \ldots, \Phi_L$, that is

$$\Phi(x, y) = \sum_{\ell=1}^{L} \Phi_\ell(x, y) \cdot \mathbf{1}_{(x,y)\in R_\ell}. \tag{11}$$

By the symmetry principle [68], if the limits of Φ_ℓ and $\Phi_{\ell+1}$ at Γ_ℓ are the same and are continuous for all ℓ, then Φ is a conformal map. However, it is unknown whether these equality and continuity conditions are guaranteed to hold for arbitrary layer boundaries $\Gamma_1, \ldots, \Gamma_L$.

Analogous to the axis-aligned setting, we call the real part $\mathbf{Re}\,\Phi_\ell(x, y)$ of the conformal map $\Phi_\ell(x, y)$ the *relative* depth of position (x, y) in layer R_ℓ. We model the expression of gene g in axis-aligned tissue slice T' using a piecewise linear function $f_g(x)$ as in Section 5. Then, our goal is to find conformal maps $\Phi_1, \ldots, \Phi_L$ and piecewise linear functions $f_g(x)$ such that the measured gene expression $\mathbf{A}$ at spots $\mathbf{S}$ in the tissue slice T are best fit by piecewise linear functions $f_g(x)$ in the corresponding axis-aligned tissue T'. We define the L-Layered Problem as the following maximum likelihood estimation problem.

L-Layered Problem. *Given SRT data $(\mathbf{A}, \mathbf{S})$, layered tissue slice $T \subseteq \mathbb{C}$, and a number L of pieces, find an axis-aligned tissue slice T' with layers $x = b_1, \ldots, x = b_{L-1}$, conformal maps $\Phi_\ell \in \mathcal{C}_T(b_{\ell-1}, b_\ell)$, and L-piecewise linear expression functions $f_1, \ldots, f_G \in \mathcal{L}(b_1, \ldots, b_{L-1})$ that maximize the log-likelihood of the data:*

$$\underset{\substack{b_1<b_2<\cdots<b_{L-1} \\ \Phi_\ell\in\mathcal{C}_T(b_{\ell-1},b_\ell) \text{ for } \ell=1,\ldots,L \\ f_1,\ldots,f_G\in\mathcal{L}(b_1,\ldots,b_{L-1})}}{\arg\max} \sum_{g=1}^{G} \left(\sum_{i=1}^{N} \log P\Big(a_{i,g} \mid f_g\big(\Phi(x_i, y_i)\big)\Big) \right). \tag{12}$$

The L-Layered Problem generalizes the assumption in the Axis-Aligned L-Layered Problem that the expression function $f_g(x, y)$ of a gene g at position (x, y) depends only on the distance from the layer boundaries $x = b_{\ell-1}$ and $x = b_\ell$; in this more general formulation, the expression function $f_g(x, y)$ of gene g is constant along contours that "interpolate" between adjacent layer boundaries $\Gamma_{\ell-1}$ and Γ_ℓ (Figure 2B).

In general, solving the L-Layered Problem without additional constraints on the conformal maps $\Phi_1, \dots, \Phi_L$ is challenging. Below, we give algorithms for solving two special cases of L-Layered Problem which provide useful approximations on real data: (1) when the layer boundaries $\Gamma_1, \dots, \Gamma_{L-1}$ are known and (2) when the layer boundaries $\Gamma_1, \dots, \Gamma_{L-1}$ are lines.

6.1 Approximate layer boundaries Γ_ℓ are given

Suppose we are given prior information about the spatial organization of a tissue slice T in the form of approximate layer boundaries $\widetilde{\Gamma}_\ell$. While these approximate layer boundaries may not directly correspond to the true layer boundaries Γ_ℓ, they can nevertheless be used to compute conformal maps Φ_ℓ. From these conformal maps, we can then estimate piecewise linear expression functions $f_1, \dots, f_G$, but without requiring that the breakpoints b_ℓ of the piecewise linear functions f_g occur at the mapped approximate layer boundaries $\Phi_\ell(\Gamma_\ell)$. In this way, we can incorporate prior information about tissue geometry (e.g. from prior knowledge of the tissue structure or from H&E images) without requiring that the expression functions conform exactly to this prior information.

Specifically, suppose we are given approximate layer boundaries $\widetilde{\Gamma}_1, \dots, \widetilde{\Gamma}_{L'-1}$, where L' is not necessarily equal to the desired number L of layer boundaries. We constrain each conformal map Φ_ℓ so that the layer boundaries $\widetilde{\Gamma}_{\ell-1}$ and $\widetilde{\Gamma}_\ell$ are mapped to lines $x = \widetilde{b}_{\ell-1}$ and $x = \widetilde{b}_\ell$, respectively, for some choice of real numbers $\widetilde{b}_1, \dots, \widetilde{b}_{L'-1}$. These constraints on the conformal maps Φ_ℓ are equivalent to the constraints $\mathbf{Re}\,\Phi_\ell(\widetilde{\Gamma}_{\ell-1}) = \widetilde{b}_{\ell-1}$ and $\mathbf{Re}\,\Phi_\ell(\widetilde{\Gamma}_\ell) = \widetilde{b}_\ell$ for all $\ell = 1, \dots, L'$, which is in turn equivalent to the constraint $\Phi_\ell \in \mathcal{C}_{\widetilde{T}}(\widetilde{b}_{\ell-1}, \widetilde{b}_\ell)$, where $\widetilde{T}$ is tissue slice T equipped with the approximate layer boundaries $\widetilde{\Gamma}_1, \dots, \widetilde{\Gamma}_{L'-1}$ and approximate layers $\widetilde{R}_1, \dots, \widetilde{R}_{L'-1}$. We define the conformal map $\Phi = \sum_{\ell=1}^{L'} \Phi_\ell(x,y) \cdot \mathbf{1}_{(x,y)\in\widetilde{R}_\ell}$ as the piecewise sum of the L' conformal maps Φ_ℓ.

Thus, given approximate layer boundaries $\widetilde{\Gamma}_1, \dots, \widetilde{\Gamma}_{L'-1}$ and lines $x = \widetilde{b}_1, \dots, x = \widetilde{b}_{L'-1}$, we solve the following optimization problem which we call the *L-Layered Problem with Approximate Layer Boundaries*:

$$\underset{\substack{b_1<b_2<\cdots<b_{L-1}\\ \Phi_\ell\in\mathcal{C}_{\widetilde{T}}(\widetilde{b}_{\ell-1},\widetilde{b}_\ell)\text{ for }\ell=1,\dots,L'\\ f_1,\dots,f_G\in\mathcal{L}(b_1,\dots,b_{L-1})}}{\arg\max} \sum_{g=1}^{G}\left(\sum_{i=1}^{N}\log P\Big(a_{i,g} \mid f_g\big(\Phi(x_i,y_i)\big)\Big)\right). \tag{13}$$

The conformal maps Φ_ℓ in (13) can be solved for separately from the breakpoints b_ℓ and expression functions f_g by using the observation that the constraint $\Phi_\ell \in \mathcal{C}_{\widetilde{T}}(\widetilde{b}_{\ell-1}, \widetilde{b}_\ell)$ *uniquely* defines the real part of the conformal map Φ_ℓ. This is because, in the constraints $\mathbf{Re}\,\Phi_\ell(\widetilde{\Gamma}_{\ell-1}) = \widetilde{b}_{\ell-1}$ and $\mathbf{Re}\,\Phi_\ell(\widetilde{\Gamma}_\ell) = \widetilde{b}_\ell$, the real part $\mathbf{Re}\,\Phi_\ell$ of each conformal map Φ_ℓ is a harmonic function, and thus can be written as the solution to the heat equation with specified boundary conditions. That is, if one fixes curve $\widetilde{\Gamma}_{\ell-1}$ to have constant temperature $\widetilde{b}_{\ell-1}$ and $\widetilde{\Gamma}_\ell$ to have constant temperature $\widetilde{b}_\ell$, then $\mathbf{Re}\,\Phi_\ell(x,y)$ is the temperature at point (x,y) and has a unique solution (Figure 2B). In practice, because there is no closed-form solution to the continuous heat equation for arbitrarily shaped boundaries, we solve the heat equation on a discretized tissue slice using a random walk-based approach [50, 36]. See the Appendix for more details.

After computing the conformal maps $\Phi_1, \dots, \Phi_{L'}$, the optimization problem in (13) reduces to

$$\underset{\substack{b_1<b_2<\cdots<b_{L-1}\\ f_1,\dots,f_G\in\mathcal{L}(b_1,\dots,b_{L-1})}}{\arg\max} \sum_{g=1}^{G}\left(\sum_{i=1}^{N}\log P\Big(a_{i,g} \mid f_g\big(\Phi(x_i,y_i)\big)\Big)\right). \tag{14}$$

This is an instance of the Axis-Aligned L-Layered Problem with transcript counts $a_{i,g}$ and transformed spots $\Phi(\mathbf{s}_i)$. Thus, we solve (14) by using the dynamic programming algorithm for the Axis-Aligned L-Layered Problem (Section 5.2).

We note that in general, there are multiple ways to specify lines $x = \widetilde{b}_1, \ldots, x = \widetilde{b}_{L'-1}$ in the L-Layered Problem with Approximate Layer Boundaries. Two reasonable choices include: (1) defining $b_1, \ldots, b_{L'-1}$ such that the difference $\widetilde{b}_\ell - \widetilde{b}_{\ell-1}$ between consecutive lines is proportional to the physical distance between layer boundaries Γ_ℓ and $\Gamma_{\ell-1}$, or (2) setting $b_\ell = \ell$ for $\ell = 1, \ldots, L'$. We note that while the optimization objective (13) is invariant under multiplicative scaling of the lines $\widetilde{b}_\ell$, the estimated slopes $\widehat{\beta}_{g,\ell}$ will differ. In the first choice, the estimated layer-specific slopes $\beta_{g,\ell}$ correspond to physical gradients of expression and are comparable both across genes g within each layer and across layers, while in the second choice the slopes are only comparable across genes g in the same layer R_ℓ. In this work, we follow (1) and set $\widetilde{b}_\ell - \widetilde{b}_{\ell-1} = d(\widetilde{\Gamma}_\ell, \widetilde{\Gamma}_{\ell-1})$ where $d(\widetilde{\Gamma}_\ell, \widetilde{\Gamma}_{\ell-1})$ is the partial Hausdorff distance, a standard distance measure in computer vision [41].

6.2 Layer boundaries Γ_ℓ are lines

The second special case of the L-Layered Problem is when the layer boundaries $\Gamma_1, \ldots, \Gamma_{L-1}$ are lines. Let $\mathcal{Q}(T)$ be the set of lines Γ with endpoints on the boundary ∂T of the tissue T. In this case, given lines $x = b_1, \ldots, x = b_{L-1}$, the L-Layered Problem reduces to solving the following optimization problem which we call the *Linear L-Layered Problem*:

$$\underset{\Gamma_1,\ldots,\Gamma_{L-1}\in\mathcal{Q}(T)}{\arg\max} \sum_{\ell=1}^{L} c(R_\ell), \tag{15}$$

$$\text{where } c(R_\ell) = \max_{\substack{f_1,\ldots,f_G\in\mathcal{L}(b_1,\ldots,b_{L-1}) \\ \Phi_\ell\in\mathcal{C}_T(b_{\ell-1},b_\ell)}} \sum_{i:(x_i,y_i)\in R_\ell} \sum_{g=1}^{G} \log P\Big(a_{i,g} \mid f_g\big(\Phi_\ell(x_i, y_i)\big)\Big). \tag{16}$$

We derive a dynamic programming algorithm to solve (15) for any convex tissue slice T and any function $c(R)$ that maps subsets $R \subseteq T$ of the tissue to $\mathbb{R}$. We note that the specific function $c(R)$ in (16) can be computed by solving the heat equation as described in Section 6.1. Our dynamic programming algorithm generalizes the dynamic programming algorithm for the Axis-Aligned L-Layered Problem from Section 5.2.

Briefly, the dynamic programming algorithm for solving (15) is as follows. Without loss of generality, let $\mathbf{s}_1, \ldots, \mathbf{s}_{N_{\text{boundary}}}$ be the spots on the boundary ∂T of the tissue T in clockwise order. For convenience, we define $[n, m]$ to be the sequence $(n+1, \ldots, m-1)$ of indices when $m > n$, and the sequence $(n+1, \ldots, N_{\text{boundary}}, 1, \ldots, m-1)$ when $n > m$. We define $T_{n,m} \subseteq T$ to be the region of the tissue T that is formed by drawing a line Γ between spots $\mathbf{s}_n$ and $\mathbf{s}_m$ and has boundary spots $\{\mathbf{s}_i\}_{i\in[n,m]}$.

Let $M_{n,m,\ell}$ be the best fit with ℓ nested layers (i.e. with $\ell-1$ linear layer boundaries) in the region $T_{n,m}$, that is

$$M_{n,m,\ell} = \max_{\text{nested } \Gamma_1,\ldots,\Gamma_{\ell-1}\in\mathcal{Q}(\mathrm{T_{n,m}})} \sum_{\ell'=1}^{\ell} c(R_{\ell'}). \tag{17}$$

Then, the best fit with ℓ layers in the region $T_{n,m}$ can be decomposed to the sum of: (1) the best fit with $\ell-1$ layers in the region $T_{n',m'}$ and (2) the fit $c(T_{n,m} \setminus T_{n',m'})$ for the region $T_{n,m} \setminus T_{n',m'}$, for some $n', m' \in [n, m[$. Thus, we have the following recurrence relationship:

$$M_{n,m,\ell} = \max_{\substack{n',m' \\ \text{s.t. } [n',m']\subseteq[n,m]}} \left(M_{n',m',\ell-1} + c(T_{n,m} \setminus T_{n',m'})\right). \tag{18}$$

After computing the values $M_{n,m,\ell}$ for all $1 \le n, m \le N_{\text{boundary}}$ and $\ell = 1, \ldots, L-1$, the optimal value of (15) is then derived as

$$\max_{\Gamma_1,\ldots,\Gamma_{L-1} \in \mathcal{Q}(T)} \sum_{\ell=1}^{L} c(R_\ell) = \max_{n \neq m} (M_{n,m,L-1} + M_{m,n,1}). \tag{19}$$

The dynamic programming algorithm consists of using the recursion (18) to fill in a table, followed by computing (19) to identify the best fit with $L-1$ linear layer boundaries for the entire tissue T. See Figure S4 and the Appendix for more details.

6.2.1 Layer boundaries Γ_ℓ are parallel lines

We also derive a more computationally efficient algorithm for the case when the layer boundaries $\Gamma_1, \ldots, \Gamma_{L-1}$ are *parallel* lines. In this case, each conformal map Φ_ℓ is a rotation map $P_\theta : \mathbb{R}^2 \to \mathbb{R}^2$ with the same angle θ, where $P_\theta(x, y)$ rotates points (x, y) about the origin by angle $\theta \in [0, \pi)$. Then, the L-Layered Problem reduces to the following optimization problem which we call the *θ-Rotated L-Layered Problem*:

$$\underset{\substack{\theta \in [0,\pi) \\ b_1 < b_2 < \cdots < b_{L-1} \\ (f_1,\ldots,f_G) \in \mathcal{L}(b_1,\ldots,b_{L-1})}}{\arg\max} \sum_{g=1}^{G} \sum_{i=1}^{N} \log P\Big(a_{i,g} \mid f_g\big(P_\theta(x_i, y_i)\big)\Big). \tag{20}$$

For a fixed angle θ, the optimization problem in (20) reduces to the Axis-Aligned L-Layered Problem with expression matrix A and rotated spots $P_\theta(x_i, y_i)$. Thus, we solve the θ-Rotated L-Layered Problem by performing a parameter sweep over angles $\theta \in [0, \pi)$ and finding the value of θ that maximizes the objective of (20).

7 Improving run-time with dimensionality reduction

The algorithms described above for solving different versions of the L-Layered Problem solve a separate regression problem for each gene. This can result in large run-times when the number of genes G is large, particularly when using a Poisson model for the UMI counts $a_{i,g}$. This is because Poisson regression does not have a closed-form solution and requires solving a convex optimization problem, which is generally slower to solve than linear regression.

To reduce the run-time for SRT data $(\mathbf{A}, \mathbf{S})$, we use generalized principal components analysis (GLM-PCA) [87] to produce a low-dimensional representation $\mathbf{U}$ of the expression matrix $\mathbf{A}$, and then run the algorithms presented above on the top-$2L$ generalized principal components $u_1, \ldots, u_{2L} \in \mathbb{R}^N$ using the Gaussian model. This results in large reductions in run-time: for example, for the Axis-Aligned L-Layered Problem, we reduce the run-time of our DP algorithm from $O(LN^2G \cdot P_0)$ under the Poisson model to $O(LN^2 \cdot 2L) = O(N^2L^2)$, corresponding to the run-time for DP with the Gaussian model and $G = 2L$ genes. After computing the breakpoints $b_1, \ldots, b_L$, we then estimate the expression functions f_g for each gene g by computing L separate Poisson regressions as described in Section 5.1, for which the run-time is only $O(LGP_0)$. In practice, on the DLPFC Donor I data we observe a $100\times$ reduction in run-time when solving the Axis-Aligned L-Layered Problem with GLM-PCA-reduced data matrix $\mathbf{U}$, as opposed to the original data matrix $\mathbf{A}$, with nearly identical performance in layer identification.

Our use of GLM-PCA is motivated by the observation that for SRT data $(\mathbf{A}, \mathbf{S})$ following the Poisson model with piecewise linear expression functions f_g and breakpoints $b_1, \ldots, b_{L-1}$, the top-$2L$ generalized principal components $u_1, \ldots, u_{2L}$ of the expression matrix $\mathbf{A}$ are approximately piecewise linear with breakpoints $b_1, \ldots, b_{L-1}$. See the Appendix for more details.

8 Simulation details

For the first simulation, we simulated SRT data $(\mathbf{A}, \mathbf{S})$ from a rectangular tissue slice T with parallel layer boundaries lines $\Gamma_1, \ldots, \Gamma_{L-1}$ that are rotated by an angle $\theta = 40^\circ$ from the x-axis. The observed spots $\mathbf{S}$ are equally spaced in the tissue slice T and form a 50×40 grid. We generated piecewise linear expression functions $f_g(x, y)$ for $G = 1000$ genes with slopes $\beta_{g,\ell}$ and y-intercepts $\alpha_{g,\ell}$ chosen uniformly at random from $[-0.1, 0.1]$. We generated transcript counts $a_{i,g}$ using the Poisson model with added Gaussian noise $\epsilon \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$:

$$a_{i,g} \sim \text{Poisson}(C_i \cdot \exp(f_g(P_\theta(x_i, y_i)) + \epsilon_i) \tag{21}$$

where C_i was chosen so that each spot had total UMI count of approximately 2000. We ran Belayer to solve the θ-Rotated L-Layered Problem (Section 6.2.1), and compared Belayer to three other approaches: BayesSpace [99], SpaGCN [40], and the SCANPY [92] implementation of leiden clustering algorithm [88]. (We do not compare against stLearn [74] as an H&E image provides crucial information in its spatial cluster step, which smooths gene expression across space by weighted averaging where the weights are derived from H&E image.) We assume all methods know the true number L of layers. We assessed the performance of each method by computing the Adjusted Rand Index (ARI) between the L layers/clusters estimated by each method and the true layers.

For the second simulation, we evaluated Belayer on simulated SRT data $(\mathbf{A}, \mathbf{S})$ from a rectangular tissue slice T similar to the tissue slice described above, except θ is uniformly randomly chosen from $[-\frac{\pi}{4}, \frac{\pi}{4}]$, and the spots form a 70×50 grid. We generated transcript counts $a_{i,g}$ using Splatter [96] with $G = 1000$ genes such that: the expression function $f_g(x, y)$ is approximately piecewise linear in each layer for each gene g (using the trajectory simulation feature of Splatter and placing cells in each spot by the pseudotime of each trajectory), the median UMI count per spot is around 1100, and the probability that a gene has a different linear expression function across layers, which we call DE probability, is set to a fixed value using the differential expression probability parameter in Splatter. We generated 5 simulated SRT datasets $(\mathbf{A}, \mathbf{S})$ for each value of $L = 4 \ldots, 8$ and each value of DE probability in $\{0.1, 0, 2, 0.3, 0.4, 0.5\}$. The details for running Belayer and comparing to other methods are the same as the first simulation.

9 Data processing and parameter selection

For Belayer, we choose the number L of layers by identifying the elbow in the consecutive differences of the negative log-likelihood. See Figure S7 for an example for DLPFC sample 151508 and the skin wound dataset.

We use the following data processing and parameter settings for running other SRT methods.

BayesSpace: We use function `spatialPreprocess` implemented in BayesSpace for data pre-processing. This function normalizes the count matrix, selects a given number of highly variable genes, and computes a given number of principle components using PCA. We compute BayesSpace clustering results with both 2000 highly variable genes for real SRT datasets, and use all genes for simulated data. We follow the instructions in BayesSpace paper and set the number of PCs to be 15 for DLPFC dataset, and we use $2L$ PCs for simulated data and other SRT datasets. The number of clusters is either set to be the same as L chosen by Belayer or selected by its model selection function `qTune` in real SRT datasets, and set to be the true number of clusters in simulated data.

SCANPY: We follow the SCANPY tutorial to pre-process DLPFC and simulated data. The UMI count matrix is normalized to a target sum of 10^6 per spot for the DLPFC data and 10^3 per spot for the simulated data, and then transformed by $\log 1\text{p}$. We use all genes to compute PCs using SCANPY with the default number of PCs. We then use SCANPY to construct a neighborhood graph with various number of neighbors, and apply leiden algorithm in SCANPY to the neighborhood graph for clustering with various resolution

parameter. Since SCANPY does not have its own model selection function, we vary the number of neighbors parameter within the range of 10 and 50 and vary the resolution parameter within the range of 0.1 and 1.5 until the number of clusters matches the number L of layers chosen by Belayer.

stLearn: We follow the stSME clustering tutorial of stLearn to cluster DLPFC, mouse somatosensory, and mouse skin data. Briefly, stSME clustering of stLearn first combines spatial location and morphology from image to normalize gene expression, and then applies PCA and KMeans clustering to the normalized gene expression. Slide-SeqV2 does not generate H&E images, and stLearn uses only spatial location for gene expression normalization and further clustering.

SpaGCN: We follow the SpaGCN tutorial for applying SpaGCN to 10X Visium and Slide-Seq data. As suggested by the SpaGCN tutorial and manuscript, we use parameter $p = 0.5$ for 10X Visium data and $p = 1$ for Slide-SeqV2 data.

HotSpot: We run HotSpot with the default settings (as suggested by the Github repository): setting the null model to be the depth-adjusted negative binomial model and setting the number of neighbors in the k-NN graph to be 30. We rank genes by their estimated FDRs.

SpatialDE: We follow the instructions of SpatialDE github for normalizing SRT data and running SpatialDE. We use the variance stabilizing normalization implemented in `NaiveDE.stabilize` for normalizing UMI count matrix, and then regress out the log of total UMI count per spot by `NaiveDE.regress_out`. SpatialDE is then applied to compute the p-value and q-value for each gene being a spatially varying gene.

SPARK: SPARK uses a Poisson distribution to model the count data, and thus normalization is not needed. We apply SPARK to the UMI count data directly, and set the gene-filtering parameter in SPARK to filter out genes that are expressed in less than 10% of spots or have less than 10 UMI counts in total.

10 Code and Data Availability

Belayer is available at github.com/raphael-group/belayer. The DLPFC dataset is available at http://spatial.libd.org/spatialLIBD/. The mouse skin wound dataset can be accessed from GEO with accession GSE178758. The full size Slide-SeqV2 mouse cortex dataset is available at https://singlecell.broadinstitute.org/single_cell/study/SCP815/highly-sensitive-spatial-transcriptomics-at-near-cellular-resolution-with-slide-seqv2, and the code to extract the mouse somatosensory cortex is located ast https://github.com/dmcable/spacexr/blob/master/AnalysisPaper/SuppFigures/supp_part2.Rmd in the section "Somatosensory Cortex Cortical Cell Types".

11 Acknowledgements

U.C. is supported by NSF GRFP DGE 2039656. This research is supported by National Cancer Institute (NCI) grant U24CA264027 to B.J.R.

References

[1] 10x Genomics. Spatial transcriptomics, 2021.

[2] J. Acharya, I. Diakonikolas, J. Li, and L. Schmidt. Fast algorithms for segmented regression. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2878–2886, New York, New York, USA, 20–22 Jun 2016. PMLR.

[3] A. Agah, T. R. Kyriakides, N. Letrondo, B. Björkblom, and P. Bornstein. Thrombospondin 2 levels are increased in aged mice: consequences for cutaneous wound healing and angiogenesis. *Matrix Biology*, 22(7):539–547, 2004.

[4] L. V. Ahlfors. *Complex Analysis*. McGraw-Hill Book Company, 2 edition.

[5] L. V. Ahlfors. *Conformal invariants: topics in geometric function theory*, volume 371. American Mathematical Soc., 2010.

[6] B. Alberts, D. Bray, K. Hopkin, A. D. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Essential cell biology*. Garland Science, 2015.

[7] S. Aminikhanghahi and D. J. Cook. A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51(2):339–367, 2017.

[8] A. Andersson, J. Bergenstråhle, M. Asp, L. Bergenstråhle, A. Jurek, J. Fernández Navarro, and J. Lundeberg. Single-cell and spatial transcriptomics enables probabilistic inference of cell type topography. *Communications Biology*, 3(1):565, 2020.

[9] A. Andersson, J. Bergenstråhle, M. Asp, L. Bergenstråhle, A. Jurek, J. Fernández Navarro, and J. Lundeberg. Single-cell and spatial transcriptomics enables probabilistic inference of cell type topography. *Communications Biology*, 3(1):1–8, 2020.

[10] E. Armingol, A. Officer, O. Harismendy, and N. E. Lewis. Deciphering cell–cell interactions and communication from gene expression. *Nature Reviews Genetics*, 22(2):71–88, 2021.

[11] M. Asp, S. Giacomello, L. Larsson, C. Wu, D. Fürth, X. Qian, E. Wärdell, J. Custodio, J. Reimegård, F. Salmén, et al. A spatiotemporal organ-wide gene expression and cell atlas of the developing human heart. *Cell*, 179(7):1647–1660, 2019.

[12] L. Atta and J. Fan. Computational challenges and opportunities in spatially resolved transcriptomic data analysis. *Nature Communications*, 12(1):5283, 2021.

[13] R. Bacher, N. Leng, L.-F. Chu, Z. Ni, J. A. Thomson, C. Kendziorski, and R. Stewart. Trendy: segmented regression analysis of expression dynamics in high-throughput ordered profiling experiments. *BMC bioinformatics*, 19(1):380–380, 10 2018.

[14] J. Bai and P. Perron. Estimating and testing linear models with multiple structural changes. *Econometrica*, 66(1):47–78, 1998.

[15] E. Berglund, J. Maaskola, N. Schultz, S. Friedrich, M. Marklund, J. Bergenstråhle, F. Tarish, A. Tanoglidi, S. Vickovic, L. Larsson, et al. Spatial maps of prostate cancer transcriptomes reveal an unexplored landscape of heterogeneity. *Nature Communications*, 9(1):1–13, 2018.

[16] J. Beringer, J. F. Arguin, et al. Review of particle physics. *Phys. Rev. D*, 86:010001, Jul 2012.

[17] J. Briscoe and S. Small. Morphogen rules: design principles of gradient-mediated embryo patterning. *Development*, 142(23):3996–4009, 12 2015.

[18] D. M. Cable, E. Murray, V. Shanmugam, S. Zhang, M. Diao, H. Chen, E. Z. Macosko, R. A. Irizarry, and F. Chen. Cell type-specific differential expression in spatial transcriptomics. *bioRxiv*, 2021.

[19] D. M. Cable, E. Murray, L. S. Zou, A. Goeva, E. Z. Macosko, F. Chen, and R. A. Irizarry. Robust decomposition of cell type mixtures in spatial transcriptomics. *Nature Biotechnology*, 2021.

[20] Z. Cang and Q. Nie. Inferring spatial and signaling relationships between cells from single cell transcriptomic data. *Nature Communications*, 11(1):1–13, 2020.

[21] M. S. Cembrowski, J. L. Bachman, L. Wang, K. Sugino, B. C. Shields, and N. Spruston. Spatial gene-expression gradients underlie prominent heterogeneity of ca1 pyramidal neurons. *Neuron*, 89(2):351–368, 2016.

[22] K. H. Chen, A. N. Boettiger, J. R. Moffitt, S. Wang, and X. Zhuang. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science*, 348(6233), 2015.

[23] B. Chidester, T. Zhou, and J. Ma. Spicemix: Integrative single-cell spatial modeling for inferring cell identity. *bioRxiv*, 2021.

[24] G. C. Chow. Tests of equality between sets of coefficients in two linear regressions. *Econometrica*, 28(3):591–605, 1960.

[25] S. Codeluppi, L. E. Borm, A. Zeisel, G. La Manno, J. A. van Lunteren, C. I. Svensson, and S. Linnarsson. Spatial organization of the somatosensory cortex revealed by osmfish. *Nature Methods*, 15(11):932–935, 2018.

[26] F. B. D., M. J. D., Z. Valerio, and W. Xiao-Jing. Multimodal gradients across mouse cortex. *Proceedings of the National Academy of Sciences*, 116(10):4689–4695, 2022/04/10 2019.

[27] D. DeTomaso and N. Yosef. Hotspot identifies informative gene modules across modalities of single-cell genomics. *Cell Systems*, 12(5):446–456.e9, 2021.

[28] R. Dries, Q. Zhu, R. Dong, C.-H. L. Eng, H. Li, K. Liu, Y. Fu, T. Zhao, A. Sarkar, F. Bao, R. E. George, N. Pierson, L. Cai, and G.-C. Yuan. Giotto: a toolbox for integrative analysis and visualization of spatial expression data. *Genome Biology*, 22(1):78, 2021.

[29] D. Edsgärd, P. Johnsson, and R. Sandberg. Identification of spatial expression trends in single-cell gene expression data. *Nature Methods*, 15(5):339–342, 2018.

[30] M. Elosua-Bayes, P. Nieto, E. Mereu, I. Gut, and H. Heyn. SPOTlight: seeded NMF regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic Acids Research*, 49(9):e50–e50, 02 2021.

[31] M. Elosua-Bayes, P. Nieto, E. Mereu, I. Gut, and H. Heyn. SPOTlight: seeded NMF regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic Acids Research*, 49(9):e50–e50, 2021.

[32] R. Elyanow, R. Zeira, M. Land, and B. J. Raphael. STARCH: copy number and clone inference from spatial transcriptomics data. *Physical Biology*, 18(3):035001, 03 2021.

[33] C.-H. L. Eng, M. Lawson, Q. Zhu, R. Dries, N. Koulena, Y. Takei, J. Yun, C. Cronin, C. Karp, G.-C. Yuan, et al. Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH+. *Nature*, 568(7751):235–239, 2019.

[34] D. S. Foster, M. Januszyk, K. E. Yost, M. S. Chinta, G. S. Gulati, A. T. Nguyen, A. R. Burcham, A. Salhotra, R. C. Ransom, D. Henn, et al. Integrated spatial multiomics reveals fibroblast fate during tissue repair. *Proceedings of the National Academy of Sciences*, 118(41), 2021.

[35] R. C. Geary. The contiguity ratio and statistical mapping. *The incorporated statistician*, 5(3):115–146, 1954.

[36] L. J. Grady and E. L. Schwartz. *Anisotropic interpolation on graphs: The combinatorial Dirichlet problem*. Citeseer, 2003.

[37] F. J. Grisanti Canozo, Z. Zuo, J. F. Martin, and M. A. H. Samee. Cell-type modeling in spatial transcriptomics data elucidates spatially variable colocalization and communication between cell-types in mouse brain. *Cell Systems*, 13(1):58–70.e5, 2022/01/30 2022.

[38] J. Hass, E. Walton, H. Kirsten, J. Turner, R. Wolthusen, V. Roessner, S. R. Sponheim, D. Holt, R. Gollub, V. D. Calhoun, and S. Ehrlich. Complexin2 modulates working memory-related neural activity in patients with schizophrenia. *European archives of psychiatry and clinical neuroscience*, 265(2):137–145, 03 2015.

[39] F. Hildebrandt, A. Andersson, S. Saarenpää, L. Larsson, N. Van Hul, S. Kanatani, J. Masek, E. Ellis, A. Barragan, A. Mollbrink, E. R. Andersson, J. Lundeberg, and J. Ankarklev. Spatial transcriptomics to define transcriptional patterns of zonation and structural components in the mouse liver. *Nature Communications*, 12(1):7046, 2021.

[40] J. Hu, X. Li, K. Coleman, A. Schroeder, N. Ma, D. J. Irwin, E. B. Lee, R. T. Shinohara, and M. Li. Spagcn: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nature Methods*, 18(11):1342–1351, 2021.

[41] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 1993.

[42] B. K. Ip, I. Wappler, H. Peters, S. Lindsay, G. J. Clowry, and N. Bayatti. Investigating gradients of gene expression involved in early human cortical development. *Journal of anatomy*, 217(4):300–311, 10 2010.

[43] A. L. Ji, A. J. Rubin, K. Thrane, S. Jiang, D. L. Reynolds, R. M. Meyers, M. G. Guo, B. M. George, A. Mollbrink, J. Bergenstråhle, et al. Multimodal analysis of composition and spatial architecture in human squamous cell carcinoma. *Cell*, 182(2):497–514, 2020.

[44] E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. Siegelbaum, A. J. Hudspeth, S. Mack, et al. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.

[45] M. Khalil, C. E. Teunissen, M. Otto, F. Piehl, M. P. Sormani, T. Gattringer, C. Barro, L. Kappos, M. Comabella, F. Fazekas, A. Petzold, K. Blennow, H. Zetterberg, and J. Kuhle. Neurofilaments as biomarkers in neurological disorders. *Nature Reviews Neurology*, 14(10):577–589, 2018.

[46] S. Y. Kim and M. G. Nair. Macrophages in wound healing: activation and plasticity. *Immunology and Cell Biology*, 97(3):258–267, 2019.

[47] V. Kleshchevnikov, A. Shmatko, E. Dann, A. Aivazidis, H. W. King, T. Li, R. Elmentaite, A. Lomakin, V. Kedlian, A. Gayoso, et al. Cell2location maps fine-grained cell types in spatial transcriptomics. *Nature Biotechnology*, pages 1–11, 2022.

[48] V. Kleshchevnikov, A. Shmatko, E. Dann, A. Aivazidis, H. W. King, T. Li, R. Elmentaite, A. Lomakin, V. Kedlian, A. Gayoso, M. S. Jain, J. S. Park, L. Ramona, E. Tuck, A. Arutyunyan, R. Vento-Tormo, M. Gerstung, L. James, O. Stegle, and O. A. Bayraktar. Cell2location maps fine-grained cell types in spatial transcriptomics. *Nature Biotechnology*, 2022.

[49] P. A. Kolarsick, M. A. Kolarsick, and C. Goodwin. Anatomy and physiology of the skin. *Journal of the Dermatology Nurses' Association*, 3(4):203–213, 2011.

[50] G. F. Lawler. *Random walk and the heat equation*, volume 55. American Mathematical Soc., 2010.

[51] A. Le Bras. Mapping gradients in the mouse cortex. *Lab Animal*, 48(5):140–140, 2019.

[52] D. Lee, H. C. Tien, C. P. Luo, and H. N. Luk. Hexagonal grid methods with applications to partial differential equations. *International Journal of Computer Mathematics*, 91(9):1986–2009, 09 2014.

[53] E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, A. Bensinger, A. Bernard, A. F. Boe, M. S. Boguski, K. S. Brockway, E. J. Byrnes, et al. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445(7124):168–176, 2007.

[54] D. Li, J. Ding, and Z. Bar-Joseph. Identifying signaling genes in spatial single-cell expression data. *Bioinformatics*, 37(7):968–975, 2021.

[55] Q. Li, M. Zhang, Y. Xie, and G. Xiao. Bayesian modeling of spatial molecular profiling data via Gaussian process. *Bioinformatics*, 37(22):4129–4136, 06 2021.

[56] K.-H. Ling, C. A. Hewitt, T. Beissbarth, L. Hyde, P.-S. Cheah, G. K. Smyth, S.-S. Tan, C. N. Hahn, T. Thomas, P. Q. Thomas, and H. S. Scott. Spatiotemporal Regulation of Multiple Overlapping Sense and Novel Natural Antisense Transcripts at the Nrgn and Camk2n1 Gene Loci during Mouse Cerebral Corticogenesis. *Cerebral Cortex*, 21(3):683–697, 08 2010.

[57] R. Littman, Z. Hemminger, R. Foreman, D. Arneson, G. Zhang, F. Gómez-Pinilla, X. Yang, and R. Wollman. Joint cell segmentation and cell type annotation for spatial transcriptomics. *Molecular Systems Biology*, 17(6):e10108, 2021.

[58] E. Lundberg and G. H. H. Borner. Spatial proteomics: a powerful discovery tool for cell biology. *Nature Reviews Molecular Cell Biology*, 20(5):285–302, 2019.

[59] M. Mantri, G. J. Scuderi, R. Abedini-Nassab, M. F. Wang, D. McKellar, H. Shi, B. Grodner, J. T. Butcher, and I. De Vlaminck. Spatiotemporal single-cell RNA sequencing of developing chicken hearts identifies interplay between cellular differentiation and morphogenesis. *Nature Communications*, 12(1):1–13, 2021.

[60] R. E. Marc. Functional neuroanatomy of the retina. *Albert and Jakobiec's Principles and Practice of Ophthalmology. 3rd ed: Elsevier*, pages 1565–1592, 2008.

[61] F. Martini, M. J. Timmons, R. B. Tallitsch, W. C. Ober, C. W. Garrison, K. B. Welch, and R. T. Hutchings. *Human anatomy*. Pearson/Benjamin Cummings San Francisco, CA, 2006.

[62] K. R. Maynard, L. Collado-Torres, L. M. Weber, C. Uytingco, B. K. Barry, S. R. Williams, J. L. Catallini, M. N. Tran, Z. Besich, M. Tippani, et al. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nature Neuroscience*, 24(3):425–436, 2021.

[63] B. J. Molyneaux, P. Arlotta, J. R. Menezes, and J. D. Macklis. Neuronal subtype specification in the cerebral cortex. *Nature Reviews Neuroscience*, 8(6):427–437, 2007.

[64] B. J. Molyneaux, P. Arlotta, J. R. L. Menezes, and J. D. Macklis. Neuronal subtype specification in the cerebral cortex. *Nature Reviews Neuroscience*, 8(6):427–437, 2007.

[65] R. Moncada, D. Barkley, F. Wagner, M. Chiodin, J. C. Devlin, M. Baron, C. H. Hajdu, D. M. Simeone, and I. Yanai. Integrating microarray-based spatial transcriptomics and single-cell rna-seq reveals tissue architecture in pancreatic ductal adenocarcinomas. *Nature Biotechnology*, 38(3):333–342, 2020.

[66] A. E. Moor, Y. Harnik, S. Ben-Moshe, E. E. Massasa, M. Rozenberg, R. Eilam, K. B. Halpern, and S. Itzkovitz. Spatial reconstruction of single enterocytes uncovers broad zonation along the intestinal villus axis. *Cell*, 175(4):1156–1167, 2018.

[67] P. A. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37(1/2):17–23, 1950.

[68] Z. Nehari. *Conformal mapping*. Courier Corporation, 2012.

[69] R. Nussinov and A. B. Jacobson. Fast algorithm for predicting the secondary structure of single-stranded rna. *Proceedings of the National Academy of Sciences of the United States of America*, 77(11):6309–6313, 11 1980.

[70] S. Oddoux, J. Brocard, A. Schweitzer, P. Szentesi, B. Giannesini, J. Brocard, J. Fauré, K. Pernet-Gallay, D. Bendahan, J. Lunardi, et al. Triadin deletion induces impaired skeletal muscle function. *Journal of Biological Chemistry*, 284(50):34918–34929, 2009.

[71] L. Pachter. Models for transcript quantification from rna-seq, 2011.

[72] G. Palla, D. S. Fischer, A. Regev, and F. J. Theis. Spatial components of molecular tissue biology. *Nature Biotechnology*, 40(3):308–318, 2022.

[73] V. Petukhov, R. J. Xu, R. A. Soldatov, P. Cadinu, K. Khodosevich, J. R. Moffitt, and P. V. Kharchenko. Cell segmentation in imaging-based spatial transcriptomics. *Nature Biotechnology*, 40(3):345–354, 2022.

[74] D. Pham, X. Tan, J. Xu, L. F. Grice, P. Y. Lam, A. Raghubar, J. Vukovic, M. J. Ruitenberg, and Q. Nguyen. stlearn: integrating spatial location, tissue morphology and gene expression to find cell types, cell-cell interactions and spatial trajectories within undissociated tissues. *bioRxiv*, 2020.

[75] A. Rao, D. Barkley, G. S. França, and I. Yanai. Exploring tissue architecture using spatial transcriptomics. *Nature*, 596(7871):211–220, 2021.

[76] S. G. Rodriques, R. R. Stickels, A. Goeva, C. A. Martin, E. Murray, C. R. Vanderburg, J. Welch, L. M. Chen, F. Chen, and E. Z. Macosko. Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science*, 363(6434):1463–1467, 2019.

[77] A. Sarkar and M. Stephens. Separating measurement and expression models clarifies confusion in single-cell rna sequencing analysis. *Nature Genetics*, 53(6):770–777, 2021.

[78] Q. Song and J. Su. DSTG: deconvoluting spatial transcriptomics data through graph-based artificial intelligence. *Briefings in Bioinformatics*, 22(5):bbaa414, 2021.

[79] P. L. Ståhl, F. Salmén, S. Vickovic, A. Lundmark, J. F. Navarro, J. Magnusson, S. Giacomello, M. Asp, J. O. Westholm, M. Huss, et al. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science*, 353(6294):78–82, 2016.

[80] R. R. Stickels, E. Murray, P. Kumar, J. Li, J. L. Marshall, D. J. Di Bella, P. Arlotta, E. Z. Macosko, and F. Chen. Highly sensitive spatial transcriptomics at near-cellular resolution with Slide-seqV2. *Nature Biotechnology*, 39(3):313–319, 2021.

[81] T. Stuart, A. Butler, P. Hoffman, C. Hafemeister, E. Papalexi, r. Mauck, William M, Y. Hao, M. Stoeckius, P. Smibert, and R. Satija. Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902.e21, 06 2019.

[82] S. Sun, J. Zhu, and X. Zhou. Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies. *Nature Methods*, 17(2):193–200, 2020.

[83] V. Svensson, S. A. Teichmann, and O. Stegle. Spatialde: identification of spatially variable genes. *Nature Methods*, 15(5):343–346, 2018.

[84] M. Tenenbaum and H. Pollard. *Ordinary differential equations: an elementary textbook for students of mathematics, engineering, and the sciences*. Courier Corporation, 1985.

[85] H. Teng, Y. Yuan, and Z. Bar-Joseph. Clustering spatial transcriptomics data. *Bioinformatics*, 38(4):997–1004, 10 2021.

[86] F. W. Townes and B. E. Engelhardt. Nonnegative spatial factorization, 2021.

[87] F. W. Townes, S. C. Hicks, M. J. Aryee, and R. A. Irizarry. Feature selection and dimension reduction for single-cell rna-seq based on a multinomial model. *Genome Biology*, 20(1):295, 2019.

[88] V. A. Traag, L. Waltman, and N. J. Van Eck. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):1–12, 2019.

[89] B. Velten, J. M. Braunger, R. Argelaguet, D. Arnol, J. Wirbel, D. Bredikhin, G. Zeller, and O. Stegle. Identifying temporal and spatial patterns of variation from multimodal data using mefisto. *Nature Methods*, 2022.

[90] F. Walter, O. Stegle, and B. Velten. Fishfactor: A probabilistic factor model for spatial transcriptomics data with subcellular resolution. *bioRxiv*, page 2021.11.04.467354, 01 2021.

[91] W. Wang, Q. Zhong, L. Teng, N. Bhatnagar, B. Sharma, X. Zhang, W. Luther, L. P. Haynes, R. D. Burgoyne, M. Vidal, S. Volchenboum, D. E. Hill, and R. E. George. Mutations that disrupt phoxb interaction with the neuronal calcium sensor hpcal1 impede cellular differentiation in neuroblastoma. *Oncogene*, 33(25):3316–3324, 2014.

[92] F. A. Wolf, P. Angerer, and F. J. Theis. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19(1):1–5, 2018.

[93] Y. Yamamoto and P. Perron. Estimating and testing multiple structural changes in linear models using band spectral regressions. *The Econometrics Journal*, 16(3):400–429, 2013.

[94] Y. Yang, X. Shi, W. Liu, Q. Zhou, M. Chan Lau, J. Chun Tatt Lim, L. Sun, C. C. Y. Ng, J. Yeong, and J. Liu. SC-MEB: spatial clustering with hidden Markov random field using empirical Bayes. *Briefings in Bioinformatics*, 23(1), 11 2021. bbab466.

[95] Y. Yuan and Z. Bar-Joseph. GCNG: graph convolutional networks for inferring gene interaction from spatial transcriptomics data. *Genome Biology*, 21(1):1–16, 2020.

[96] L. Zappia, B. Phipson, and A. Oshlack. Splatter: simulation of single-cell RNA sequencing data. *Genome Biology*, 18(1):1–15, 2017.

[97] H. Zeng, E. H. Shen, J. G. Hohmann, S. W. Oh, A. Bernard, J. J. Royall, K. J. Glattfelder, S. M. Sunkin, J. A. Morris, A. L. Guillozet-Bongaarts, et al. Large-scale cellular-resolution gene profiling in human neocortex reveals species-specific molecular signatures. *Cell*, 149(2):483–496, 2012.

[98] D. Zhang, X. Liu, X. Xu, J. Xu, Z. Yi, B. Shan, and B. Liu. Hpcal1 promotes glioblastoma proliferation via activation of wnt/beta-catenin signalling pathway. *Journal of Cellular and Molecular Medicine*, 23(5):3108–3117, 2019.

[99] E. Zhao, M. R. Stone, X. Ren, J. Guenthoer, K. S. Smythe, T. Pulliam, S. R. Williams, C. R. Uytingco, S. E. B. Taylor, P. Nghiem, J. H. Bielas, and R. Gottardo. Spatial transcriptomics at subspot resolution with bayesspace. *Nature Biotechnology*, 2021.

[100] J. Zhu, S. Sun, and X. Zhou. Spark-x: non-parametric modeling enables scalable and robust detection of spatial expression patterns for large spatial transcriptomic studies. *Genome Biology*, 22(1):184, 2021.

Supplemental Information

A Relationship to Gaussian Process Models

Piecewise linear functions can be modeled using a Gaussian process (GP) having a one-dimensional linear covariance function $k(\mathbf{s}, \mathbf{s}')$ of the form

$$k(\mathbf{s}, \mathbf{s}') = \begin{cases} \sigma^2_{\beta,\ell} x x' + \sigma^2_{\alpha,\ell}, & \text{if } x, x' \in (b_\ell, b_{\ell+1}] \text{ for some } \ell \in \{0, \dots, L\} \\ 0 & \text{otherwise,} \end{cases} \tag{22}$$

for breakpoints $\infty = b_0 < b_1 < \cdots < b_{L-1} < b_L = \infty$, where x and x' are the x-coordinates of spots $\mathbf{s} = (x, y)$ and $\mathbf{s} = (x', y')$, respectively. Note that the linear covariance function is different from the widely used squared exponential covariance function, which is defined as $k(\mathbf{s}, \mathbf{s}') \propto \exp\{-\frac{\|\mathbf{s}-\mathbf{s}'\|_2^2}{l^2}\}$.

Sampling an expression function $f_g(x)$ from the distribution $\mathcal{GP}\big(0, k(\mathbf{s}, \mathbf{s}')\big)$ with covariance function (22) generates a piecewise linear model of the form (4) with Gaussian priors $\beta_{g,\ell} \sim N(0, \sigma^2_{\beta,\ell})$ and $\alpha_{g,\ell} \sim N(0, \sigma^2_{\alpha,\ell})$ on the slope $\beta_{g,\ell}$ and y-intercept $\alpha_{g,\ell}$, respectively, of the gene expression in layer ℓ.

Using Gaussian Process (GP) prior $f_g(x) \sim \mathcal{GP}(0, k(\mathbf{s}, \mathbf{s}'))$, we define the following maximum a posteriori (MAP) version of the Axis-Aligned L-Layered Problem.

Axis-Aligned L-Layered Problem (MAP Version). *Given SRT data $(\mathbf{A}, \mathbf{S})$ and a number L of layers, find layer boundaries $x = b_1, \dots, x = b_{L-1}$ and piecewise linear expression functions $f_1, \dots, f_G \in \mathcal{L}(b_1, \dots, b_{L-1})$ that maximize the posterior distribution of the data under the prior $f_g(x) \sim \mathcal{GP}\big(0, k(\mathbf{s}, \mathbf{s}')\big)$ for covariance function $k(\mathbf{s}, \mathbf{s}')$ in (22):*

$$\underset{\substack{b_1 < b_2 < \cdots < b_{L-1} \\ f_1, \dots, f_G \in \mathcal{L}(b_1, \dots, b_{L-1})}}{\arg\max} \sum_{g=1}^{G} \left(\sum_{i=1}^{N} \Big(\log P\big(a_{i,g} \mid f_g(x_i)\big) + \log P\big(f_g(x_i) \mid b_1, \dots, b_{L-1}\big) \Big) \right) \tag{23}$$

$$= \underset{\substack{b_1 < b_2 < \cdots < b_{L-1} \\ f_1, \dots, f_G \in \mathcal{L}(b_1, \dots, b_{P-1})}}{\arg\max} \sum_{g=1}^{G} \left(\sum_{i=1}^{N} \log P\big(a_{i,g} \mid f_g(x_i)\big) \right) - \sum_{g=1}^{G} \sum_{\ell=1}^{L} \left(\frac{\beta^2_{g,\ell}}{\sigma^2_{\beta,\ell}} + \frac{\alpha^2_{g,\ell}}{\sigma^2_{\alpha,\ell}} \right) \tag{24}$$

B Dimensionality reduction using GLM-PCA

Let $(\mathbf{A}, \mathbf{S})$ be SRT data generated from an axis-aligned tissue slice T with breakpoints $b_1, \dots, b_{L-1}$ and L-piecewise expression functions $f_g(x)$. Define the *mean expression matrix* $\mathbf{F} = [f_g(x_i)] \in \mathbb{R}^{N \times G}$ as the matrix of expression functions $f_g(x)$ evaluated at spatial locations $x = x_i$. Given integer $K > 0$, GLM-PCA aims to find matrices $\mathbf{U}, \mathbf{V}$ such that $\mathbf{F} \approx \mathbf{UV}$, i.e. $\mathbf{UV}$ is a rank-K approximation of the mean expression matrix $\mathbf{F}$. The columns u_i of $\mathbf{U}$ are the *generalized principal components* of $\mathbf{A}$.

In the Proposition below, we show that $\mathbf{F}$ has rank $2L$ by constructing an explicit rank decomposition $\mathbf{F} = \mathbf{UV}$ where the columns u_i of $\mathbf{U}$ are *piecewise linear*. We note that our rank decomposition in Proposition 2 does not correspond to the matrices $\mathbf{U}, \mathbf{V}$ found by GLM-PCA; nevertheless, we empirically observe that the top-$2L$ generalized principal components of $\mathbf{A}$ are approximately piecewise linear (Figure S1).

Proposition 2. *Let $(\mathbf{A}, \mathbf{S})$ be SRT data generated from an axis-aligned L-layered tissue slice T with layer boundaries $x = b_1, \dots, x = b_{L-1}$ and expression functions $f_g(x)$. Let $\mathbf{F} = [f_g(x_i)] \in \mathbb{R}^{N \times G}$ be the mean expression matrix. Then $\mathbf{F}$ has a decomposition of the form $\mathbf{F} = \mathbf{UV}$ for matrices $\mathbf{U} \in \mathbb{R}^{N \times 2L}$ and $\mathbf{V} \in \mathbb{R}^{2L \times G}$, where the columns $u_i \in \mathbb{R}^N$ of $\mathbf{U}$ are piecewise linear with breakpoints $b_1, \dots, b_{L-1}$.*

Proof. Without loss of generality assume that the spots are ordered by increasing x-coordinate, i.e. $x_1 \leq x_2 \leq \cdots \leq x_N$. We define vectors $u_1, u_2, \ldots, u_{2L} \in \mathbb{R}^N$ such that

$$u_{2\ell-1,i} = x_i \cdot 1_{\{b_{\ell-1}<x_j<b_\ell\}} \tag{25}$$

$$u_{2\ell,j} = 1_{\{b_{\ell-1}<x_i<b_\ell\}} \tag{26}$$

for all $\ell = 1, \ldots, L$ and $i = 1, \ldots, N$ We similarly define vectors $v_1, \ldots, v_{2L} \in \mathbb{R}^G$ as

$$v_{2\ell-1,g} = \beta_{g,\ell} \tag{27}$$

$$v_{2\ell,g} = \alpha_{g,\ell} \tag{28}$$

for all $\ell = 1, \ldots, L$ and $g = 1, \ldots, G$.

We define matrices $\mathbf{U} \in \mathbb{R}^{N\times 2L}$ and $\mathbf{V} \in \mathbb{R}^{2L\times G}$ with columns $u_1, \ldots, u_{2L} \in \mathbb{R}^N$ and rows $v_1, \ldots, v_{2L} \in \mathbb{R}^G$, respectively. Then

$$(\mathbf{UV})_{i,g} = \sum_{\ell=1}^{L} \mathbf{U}_{i,2\ell-1}\mathbf{V}_{2\ell-1,g} + \mathbf{U}_{i,2\ell}\mathbf{V}_{2\ell,g} = (\beta_{g,l}x_i + \alpha_{g,l}) \cdot 1_{\{b_{\ell-1}<x_i<b_\ell\}} = \mathbf{F}_{i,g}, \tag{29}$$

as desired. □

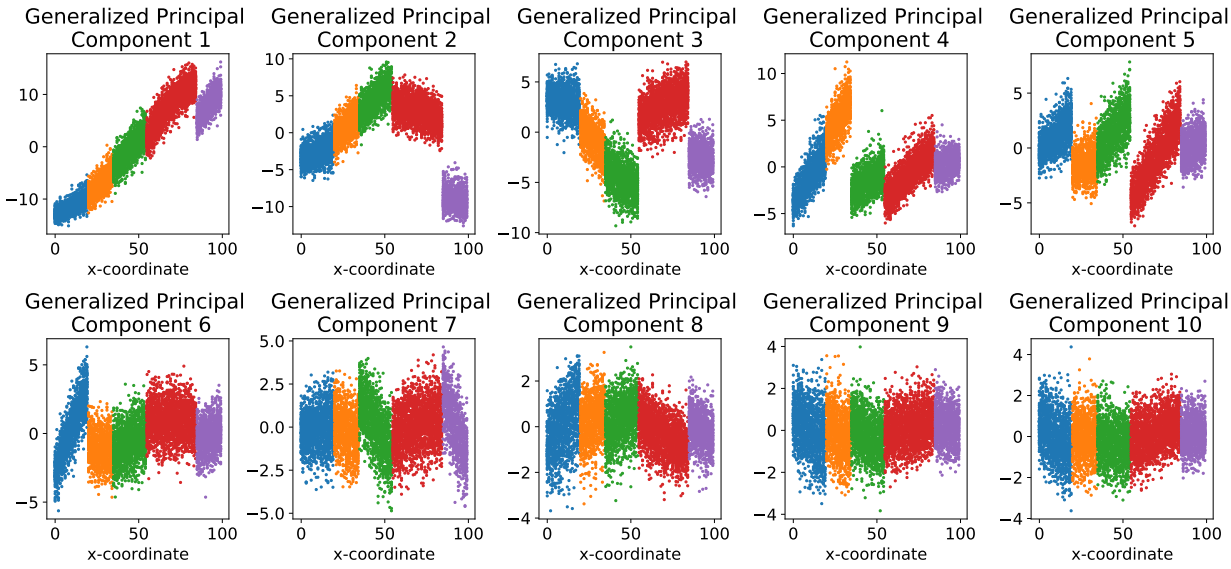

Figure S1: Top $2L$ generalized principal components for simulated SRT data $(\mathbf{A}, \mathbf{S})$ generated from an axis-aligned L-layered tissue slice T with $L = 5$ layers.

C Improving visualization by binning spots

In our visualizations, we bin the expression values $a_{i,g}$ of spots $\mathbf{s}_i = (x_i, y_i)$ with similar x-coordinates x_i. Specifically, we divide the range $[\min_i x_i, \max_i x_i]$ of x-coordinates into N_{binned} equal-width bins, where $B_j \subseteq [N]$ is the set of spots $\mathbf{s}_i = (x_i, y_i)$ in bin j and $\widetilde{x}_j$ is the center of bin j. We then construct "binned expressions" $\widetilde{a}_{j,g}$ for each gene g and bin j, where $\widetilde{a}_{j,g}$ is the maximum likelihood estimate of the expression

at x-coordinate $\widetilde{x}_j$, given the expressions $\{a_{i,g}\}_{i \in B_j}$ of spots in bin j and under the assumption these spots have x-coordinate equal to $\widetilde{x}_j$. If the expressions $a_{i,g}$ follow the Poisson expression model, then the binned expression $\widetilde{a}_{j,g} = \sum_{i:\mathbf{s}_i \in B_j} a_{i,g}$ is the total expression of all spots in bin j, while if the expressions follow the Gaussian expression model, then the binned expression $\widetilde{a}_{j,g} = \frac{1}{|B_j|} \sum_{i:\mathbf{s}_i \in B_j} a_{i,g}$ is the average expression of all spots in bin j (see e.g. Chapter 36 of [16]).

The binned expressions $[\widetilde{a}_{j,g}]$ follow the same expression model and have approximately the same expression functions as the original expressions $[a_{i,g}]$, with the expression functions being equal for a sufficiently large number N_{binned} of bins [16]. Moreover, it is easier to visualize the piecewise linear expression functions $f_g(x)$ in the binned expressions $[\widetilde{a}_{j,g}]$ versus the unbinned expressions $[a_{i,g}]$ (Figure S2).

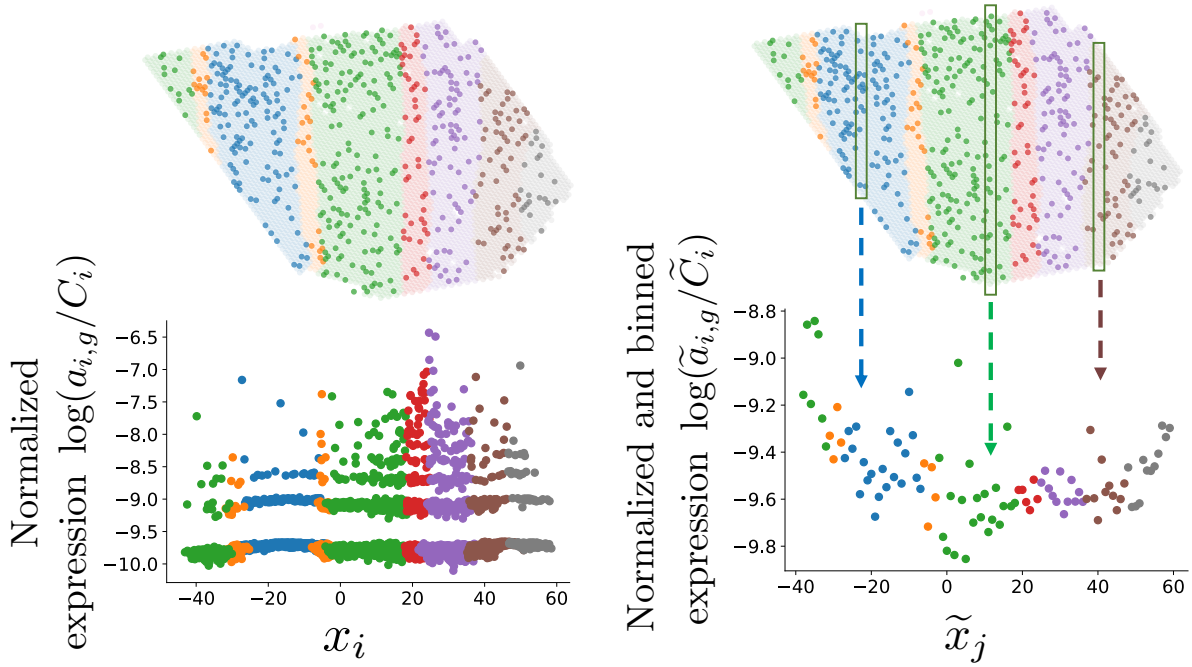

Figure S2: (Left) Visualization of x-coordinate versus normalized expression values from an axis-aligned layered tissue slice. (Right) Left plot with expressions binned across spots (x, y) with similar x-coordinates.

D Solving the L-Layered Problem with Approximate Layer Boundaries

D.1 Solving heat equation on a discretized tissue slices with boundary conditions

As described in Section 6.1, we estimate the real part $u = \mathbf{Re}\,\Phi_\ell$ of each conformal map $\Phi_\ell(x, y) = u + iv$ by solving the following heat equation:

$$\begin{aligned} \nabla^2 u &= 0 \\ u(x, y) &= \widetilde{b}_{\ell-1} \text{ for all } (x, y) \in \Gamma_{\ell-1}, \\ u(x, y) &= \widetilde{b}_\ell \text{ for all } (x, y) \in \Gamma_\ell. \end{aligned} \tag{30}$$

We solve (30) using the finite difference method [84], which estimates the values $u(x, y)$ at grid points on either a regular square grid [84] or a regular hexagonal grid [52]. For our analyses, the two datasets

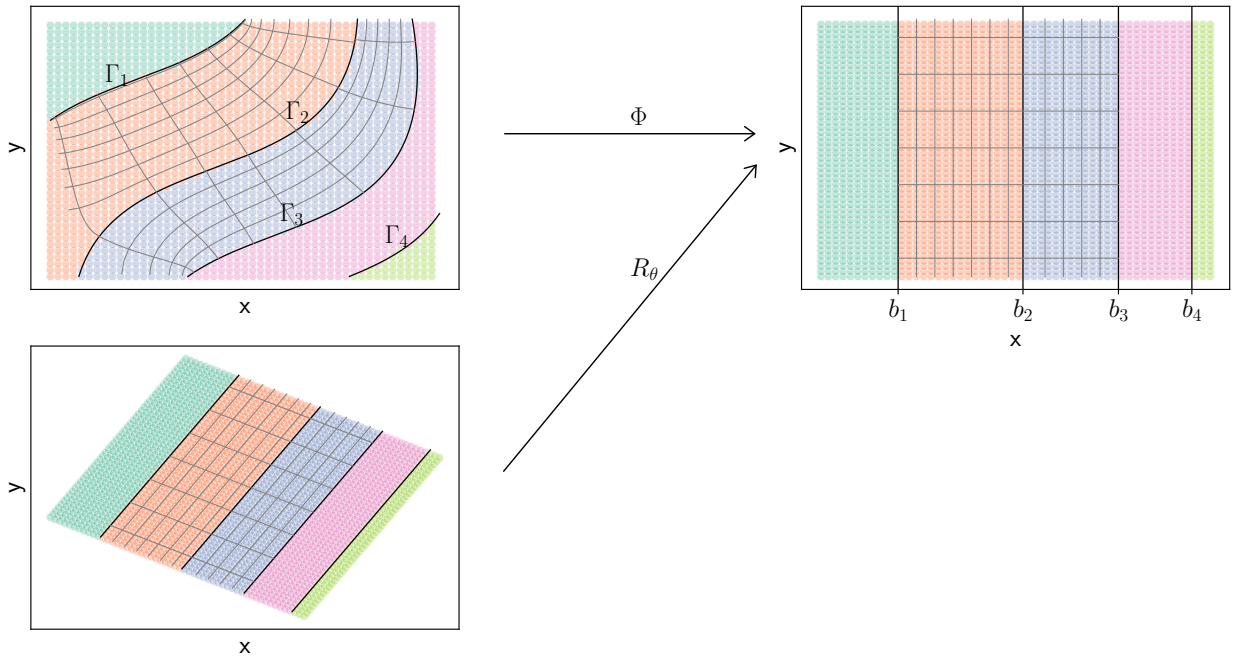

Figure S3: A conformal map Φ maps a 5-layered tissue slice to an axis-aligned tissue slice. Color indicates layers. The gray lines of the right panel in the second and third layer are grid lines parallel to x and y coordinate, their corresponding lines before mapping Φ are also known in gray lines. A special case when Φ is a rotation is shown in the lower panel.

obtained using the 10X Visium platform have spots that form a regular hexagonal grid. For the Slide-SeqV2 dataset, we overlay a regular square grid on the tissue and approximate the harmonic function value of each spot by that of its nearest grid point. We address the special cases of $\ell = 1, L$ at the end of this section.

We now briefly describe the finite difference method we implement to solve (30). Abusing notation, let $s_1, s_2, \ldots, s_N$ be the grid points (from either a square grid or a hexagonal grid). Let $B_{\ell-1}$ be the grid points closest to/lying on the layer boundary $\Gamma_{\ell-1}$. We define B_ℓ similarly. Let $\mathbf{u} = (u(s_1), u(s_2), \ldots, u(s_N))$ be the vector of the function $u(x, y)$ evaluated at grid points s_i, let $\mathbf{u}_{B_{\ell-1}}, \mathbf{u}_{B_\ell}$ be the subset of $\mathbf{u}$ corresponding to $u(x, y)$ evaluated at points in $B_{\ell-1}, B_\ell$, respectively. Let $\mathbf{u}_I$ be the function $u(x, y)$ evaluated at points not in $B_{\ell-1}, B_\ell$, i.e. the interior points. Let E_i be the set of grid points adjacent to s_i. In a regular hexagonal grid case, $|E_i| = 6$, while in a regular square grid case, $|E_i| = 4$. We partition the vector $\mathbf{u}$ into two subsets: the spots $\mathbf{u}_B$

We discretize (30) into a system of linear equations of the form

$$u_i = \frac{1}{|E_i|} \sum_{j \in E_i} u_j \tag{31}$$

for all $i = 1, \ldots, N$, where we constrain $\mathbf{u}_{B_{\ell-1}}, \mathbf{u}_{B_\ell}$ as

$$\mathbf{u}_{B_{\ell-1}} = \widetilde{b}_{\ell-1} \tag{32}$$

$$\mathbf{u}_{B_\ell} = \widetilde{b}_\ell. \tag{33}$$

This linear system can be rewritten in matrix form as $L\mathbf{u} = 0$, where L is the Laplacian matrix of the

grid, i.e.

$$L_{i,j} = \begin{cases} |E_i| & \text{if } j = i \\ -1 & \text{if } j \in E_i \\ 0 & \text{otherwise.} \end{cases}$$

We also partition the Laplacian matrix L as $L = \begin{pmatrix} L_B & D \\ D^T & L_I \end{pmatrix}$, where $B = B_{\ell-1} \cup B_\ell$. Then, the solution to the discretized heat equation (31) is given by

$$\mathbf{u}_I = -L_I^{-1} D^T \mathbf{u}_B,$$

where $\mathbf{u}_B$ is constrained by (32), (33).

For the first layer R_1 and the last layer R_L, these regions are bounded by a layer boundary Γ_ℓ on one side and a tissue boundary ∂T on all other sides. Moreover, the tissue boundaries are generated by tissue cutting and preparation in experiments and they may not be parallel to layer boundaries. Therefore, tissue boundaries are not informative of relative layer depth, and we do not solve the heat equation (30) with two boundaries condition We instead define the layer depth by heat diffusion from the one layer boundary, for which the temperature is fixed. As before, we denote the spots on the layer boundary by B and interior spots or the ones on tissue boundaries as I, and we partition the temperature $\mathbf{u}$ of spots by $\mathbf{u}_B$ and $\mathbf{u}_I$. Let W be the weighted adjacency matrix defined by

$$W_{i,j} = \begin{cases} \dfrac{1}{|N_i|} & (j \in N_i) \\ 0 & (\text{otherwise}) \end{cases}$$

for $i \in I$ and

$$W_{i,j} = \begin{cases} 1 & (j = i) \\ 0 & (\text{otherwise}) \end{cases}$$

for $i \in B$. Without loss of generality, we assume that spots in B are placed on the top left of W. The temperature of spots in I after t step diffusion is $W^t \begin{pmatrix} \mathbf{u}_B \\ \mathbf{0} \end{pmatrix}$. We use the diffusion after a fixed number of diffusion steps to define relative layer depth. The number of diffusion steps t is chosen as the smallest number of steps such that $\min \mathbf{u}_I = \delta \min \mathbf{u}_B$. That is, all spots have a nonzero temperature (δ is chosen to be 0.01) but far from reaching steady state. Note that the following regression is invariant by any affine transformation of $\mathbf{u}$. We choose an affine transformation such that the largest difference of temperature $\max \mathbf{u} - \min \mathbf{u}$ is the partial Hausdorff distance.

E Dynamic programming algorithm for Linear L-Layered Problem

We derive a DP algorithm for solving an optimization of the form (15) for any function $\mathrm{c}(R)$ that maps subsets $R \subseteq T$ of the tissue to $\mathbb{R}$. We present the algorithm for the case where the tissue slice T is a circle, but we note that the DP algorithm is directly applicable to any convex shape T.

Circle Problem Statement

We are given a circle T with $b = N_{\text{boundary}}$ fixed points around its circumference ∂T and an objective function $\mathrm{c}(R)$. We define $\mathcal{Q}(T)$ to be the set of lines Γ with endpoints on the boundary ∂T of the tissue T. Without loss of generality, let the points on ∂T be labeled clockwise $\mathbf{s}_1, \ldots, \mathbf{s}_b$ in clockwise order. We

expect $b = O(\sqrt{N})$, where N is the total number of points in the interior of T. Note that all arithmetic regarding points in ∂T will be done mod b, unless otherwise stated. Let M_L be the best fit with L layers (i.e., with $L-1$ linear layer boundaries) in T. Then our goal is to find:

$$M_L = \underset{\Gamma_1,\ldots,\Gamma_{L-1} \in \mathcal{Q}(T)}{\arg\max} \sum_{\ell=1}^{L} c(R), \tag{15}$$

Circular Segment Dynamic Program

For convenience, we define $[n, m]$ to be the sequence $(n+1, \ldots, m-1)$ of indices when $m > n$, and the sequence $(n+1, \ldots, N_{\text{boundary}}, 1, \ldots, m-1)$ when $n > m$. We define the circular segment $T_{n,m} \subseteq T$ to be the region of the tissue T that is formed by drawing a line Γ between spots $\mathbf{s}_n$ and $\mathbf{s}_m$ and has boundary spots $\{\mathbf{s}_i\}_{i \in [n,m]}$. Call layer boundaries $\mathbf{\Gamma} = \Gamma_1, \ldots, \Gamma_{\ell-1}$ *nested* if for any two lines $\overline{\text{uv}}$ and $\overline{\text{xy}}$ in $\mathbf{\Gamma}$ defined such that if x is the closest clockwise point to u in $\{x, y, v\}$, it is the case that y is the closest counterclockwise point to v in $\{x, y, u\}$. Note that nestedness is a stronger condition than the layer boundaries $\mathbf{\Gamma}$ not intersecting; see Figure S4A for an example of layer boundaries that are non-intersecting but are also not nested.

First, we consider the computation of the following related quantity (Figure S4B). Let $M_{n,m,\ell}$ be the optimal nested layer boundaries with ℓ layers in the region $T_{n,m}$, i.e.

$$M_{n,m,\ell} = \max_{\text{nested } \Gamma_1,\ldots,\Gamma_{\ell-1} \in \mathcal{Q}(\mathrm{T_{n,m}})} \sum_{\ell'=1}^{\ell} c(R). \tag{17}$$

We first give a dynamic programming algorithm for computing $M_{n,m,\ell}$ in $O(b^4 \cdot \ell)$ time. We will later leverage this algorithm in order to compute M_L.

The best fit with ℓ layers in the region $T_{n,m}$ can be decomposed as the sum of: (1) the best fit with $\ell - 1$ layers in the region $T_{n',m'}$ and (2) the fit $c(T_{n,m} \setminus T_{n',m'})$ for the region $T_{n,m} \setminus T_{n',m'}$, for some $n', m' \in [n, m]$. Thus, we have the following recurrence relationship:

$$M_{n,m,\ell} = \max_{\substack{n',m' \\ \text{s.t. } [n',m'] \subseteq [n,m]}} \left(M_{n',m',\ell-1} + c(T_{n,m} \setminus T_{n',m'}) \right) \tag{18}$$

Our base cases are:

$$M_{n,m,1} = c(T_{n,m}),$$
$$M_{n,m,\ell} = -\infty \; \forall n, m \text{ s.t. } m - n < 2\ell + 1$$

The first base case says if we have a circular segment $T_{n,m}$ with 1 layer (i.e. 0 layer boundaries), then that layer must include the entire circular segment. The second base case says that if there are not enough possible endpoints between n and m to place ℓ lines, then there is no possible solution. To see why, note that the endpoints of any layer boundary in $T_{n,m}$ must be in $V = [n+1, m-1]$ and thus $|V| = m - n - 1$. For $M_{n,m,\ell}$, we must add ℓ lines using the endpoints in V, each with 2 endpoints. $m - n < 2\ell + 1 \Rightarrow \frac{|V|}{2} < \ell$, which implies there are not enough possible endpoints to add ℓ lines. By the pigeonhole principle, there is no possible solution for $M_{n,m,\ell}$.

Our recurrence (18) finds the ℓ best nested layer boundaries contained within $T_{n,m}$. If the outermost layer boundary (i.e. the layer boundary closest to n, m) has endpoints n', m', then we must place an additional $\ell - 1$ layer boundaries within $T_{n',m'}$, otherwise the layer boundaries would not be nested. The optimal value of the objective function in (18) is then the optimal value of placing $\ell - 1$ lines within $T_{n',m'}$ added to the

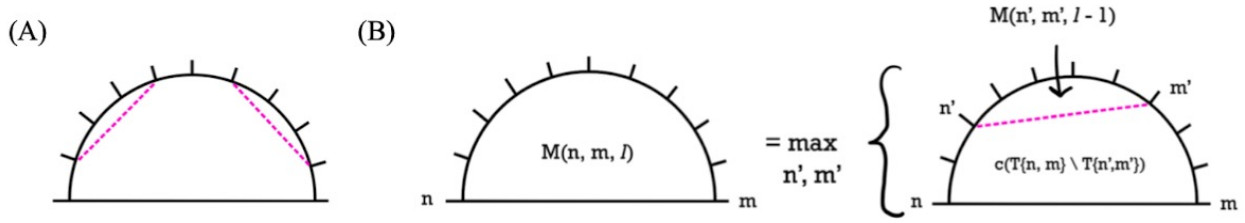

Figure S4: (**A**) Example of non-nested layer boundaries. (**B**) Visualization of recurrence for dynamic programming algorithm for linear layer boundaries.

value of the objective function for the region $T_{n,m} \setminus T_{n',m'}$. We check all possible values of n', m' and save the maximal value. See Figure S4B for a visualization of the recurrence (18).

Finding the First Layer Boundary

We find the optimal layer boundaries for the circle T by adding the value of the optimal nested layers for circular segment $T_{n,m}$ with the value of its complementary region $T_{m,n}$ for all n, m and taking the maximal result:

$$M_L = \max_{\Gamma_1,\ldots,\Gamma_{L-1} \in \mathcal{Q}(T)} \sum_{\ell=1}^{L} c(R_\ell) = \max_{n,m} \left(M_{n,m,L-1} + M_{m,n,1}\right). \tag{19}$$

Note that the optimal set of layer boundaries has some layer boundary $\overline{xy}$ such that all other layer boundaries have endpoints in $[x, y]$. Because we are maximizing over all starting values of n, m, we are guaranteed to use x, y as the starting value at some point during the maximization of (19).

The above maximization (19) computes M_L in $O(b^2)$ time, assuming a constant-time lookup for $M_{n,m,\ell}$. Furthermore, we construct a lookup table for all values of $M_{n,m,\ell}$ for $n, m = 1, \ldots, b$ and $\ell = 1, \ldots, L$ in time $O(b^4 \cdot L)$ using the following algorithm:

Algorithm 1: Creating lookup table for all $M_{n,m,\ell}$

```
Input: {s_1, ..., s_b}, L, c(R)
Output: M_{n,m,ℓ} ∀ n, m and ℓ ≤ L
for ℓ ← 0 to L do
    for n ← 0 to b − 1 do
        for m ← 0 to b − 1 do
            if n ≠ m then
                Compute M_{n,m,ℓ} using recurrence in (18)
            end
        end
    end
end
```

Note that for any $\ell \in \{1, \ldots, L\}$, the values of $M_{n,m,\ell'}$ for all n, m, and $\ell' < \ell$ will be populated in the lookup table before they are referenced by Algorithm 1 to compute $M_{n,m,\ell}$. In Algorithm 1, n and m each contribute a factor of b to the run-time. The inner maximization in the recurrence contributes a factor of b^2, and iterating over $\ell = 1, \ldots, L$ contributes a factor of L, so that Algorithm 1 has a run-time of $O(b^4 \cdot L)$. Therefore, the overall run-time to find $M_L = O(b^4 \cdot L) + O(b^2) = O(b^4 \cdot L)$. We note that this approach

is easily adapted to any convex shape T by adding the constraint that no layer boundary can have endpoints on the same edge of the shape.

In practice, for the DLPFC analysis (Figure 3), computing $c(T_{n,m} \setminus T_{n',m'})$ takes approximately 20 hours on a 100-node cluster while computing $M_{n,m,\ell}$ and M_L takes approximately $5L$ minutes in total. The bottleneck in computing $c(T_{n,m} \setminus T_{n',m'})$ is solving large linear systems in the heat equation. Improving the computation time is one future direction for improving Belayer.

F Additional approaches for marker gene identification

We additionally evaluated (Figure S10) five other approaches for ranking genes in marker gene identification. We describe these ranking approaches below.

- Ranking genes by their Moran's I score, a standard measure of spatial autocorrelation [67]
- Ranking genes by their Geary's C score, another standard measure of spatial autocorrelation [35]
- Ranking genes by their likelihood ratio test (LRT) p-value, where we compare the maximum log-likelihood (12) with a piecewise constant expression function f_g to the maximum log-likelihood (12) with a piecewise linear expression function f_g.
- Ranking genes by their LRT p-value as above but restricted to Belayer Layer 3 (Figure 3D)
- Ranking genes by the sum $\sum_{\ell=1}^{L} |\widehat{\beta}_{g,\ell}|$ of their (absolute) layer-specific slopes across the L layers identified by Belayer.
- Ranking genes by the maximum difference $\max_{\ell=1,\dots,L-1} |\widehat{f}_{g,\ell+1}(\Phi_{\ell+1}(\Gamma_\ell)) - \widehat{f}_{g,\ell+1}(\Phi_\ell(\Gamma_\ell))|$ between layer-specific expression functions $\widehat{f}_{g,\ell})$ at layer boundaries Γ_ℓ across the L layers identified by Belayer.

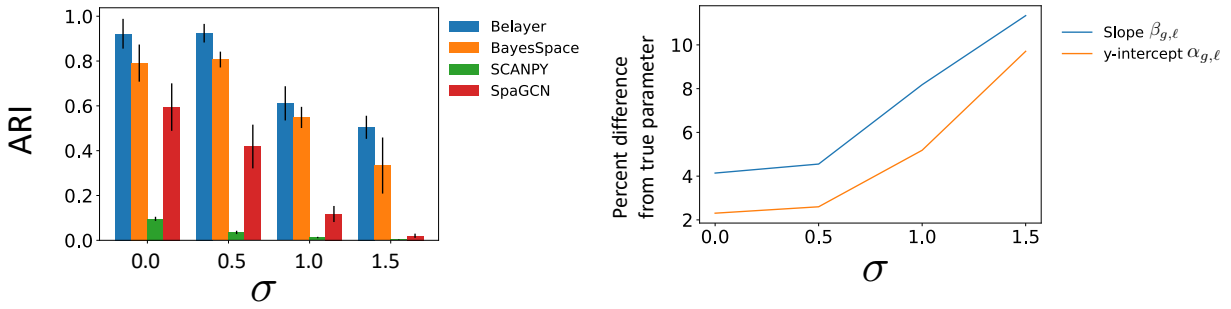

Figure S5: Comparison of Belayer, BayesSpace, SCANPY, and SpaGCN in identifying spatially distinct cell clusters in the first simulation. Performance of each method is evaluated according to the Adjusted Rand Index (ARI) and shown for different values of the number L of layers and standard deviation σ of the added Gaussian noise. Error bars indicate variation from 5 randomly simulated datasets for each parameter setting.

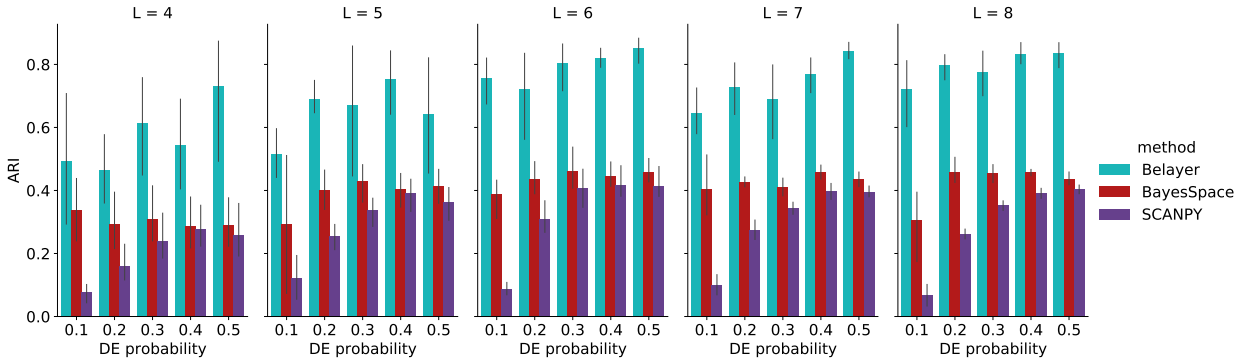

Figure S6: Comparison of Belayer, BayesSpace, and SCANPY in identifying spatially distinct cell clusters in the second simulation. Performance of each method is evaluated according to the Adjusted Rand Index (ARI) and shown for different values of the number L of layers and differential expression (DE) probability. Error bars indicate variation from 5 randomly simulated datasets for each parameter setting.

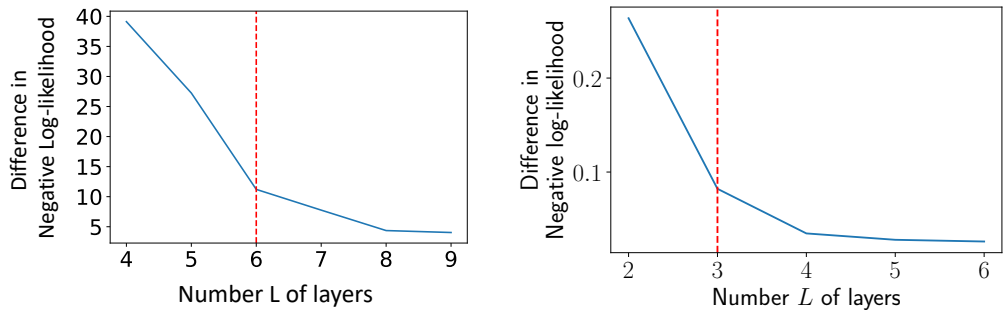

Figure S7: Number L of layers vs. difference in negative log-likelihood at L layers and $L-1$ layers for DLPFC sample 151508 (left) and the mouse skin wound dataset (right). Vertical dashed line indicates manually annotated elbow.

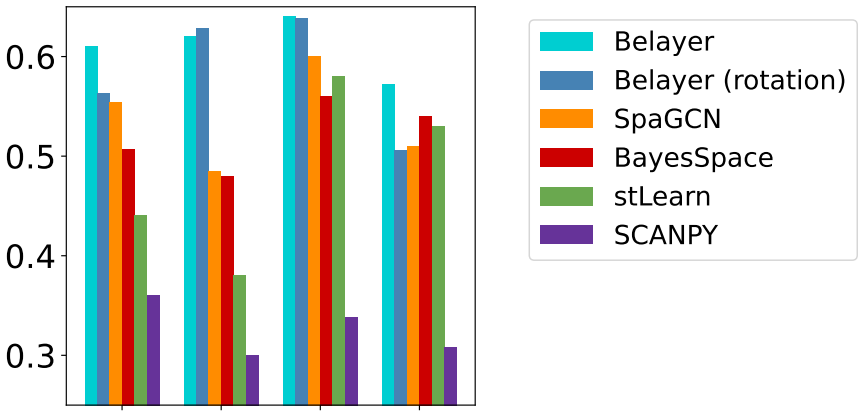

Figure S8: Comparison of Belayer (solving the Linear L-Layered Problem), Belayer (solving the θ-Rotated L-Layered Problem) and other methods in identifying annotated layers in SRT data from the DLPFC of Donor 1 as in Figure 3.

	Donor 1		Donor 2				Donor 3			
	151507	151508	151669	151670	151671	151672	151673	151674	151675	151676
Belayer ARI (Belayer model selection)	0.61	0.621	0.356	0.61	0.707	0.596	0.653	0.669	0.697	0.551
BayesSpace ARI (Belayer model selection)	0.507	0.48	0.322	0.363	0.403	0.505	0.364	0.338	0.329	0.459
BayesSpace ARI (BayesSpace model selection)	0.482	0.472	0.257	0.230	0.398	0.484	0.528	0.322	0.425	0.406

Table S1: Comparisons of Belayer, BayesSpace with Belayer's model selection, and BayesSpace with its own model selection in identifying cortical layers of DLPFC data. Two samples in Donor 1 are excluded from this table because both model selections choose the same number of layers. The first two rows match the ARIs in Figure 3A.

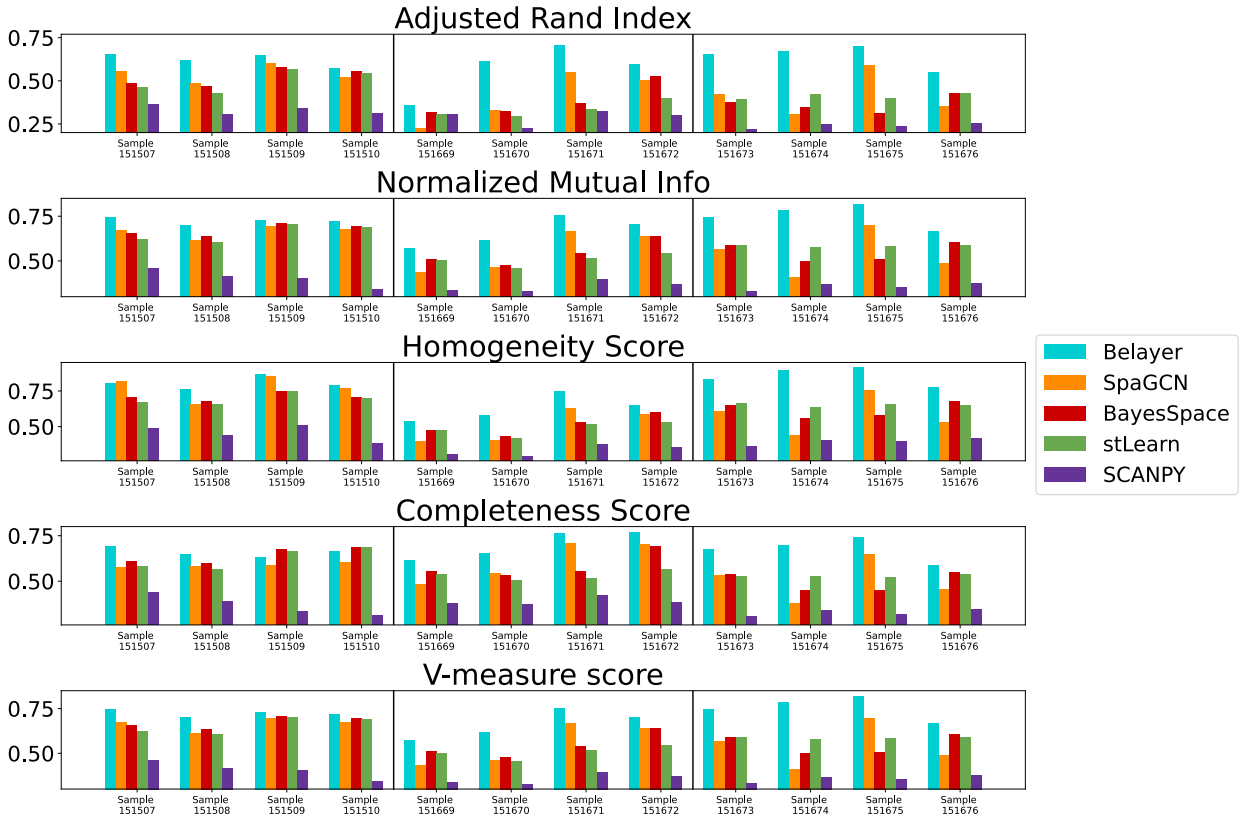

Figure S9: Comparison of Belayer and other methods from Figure 3 with different metrics.

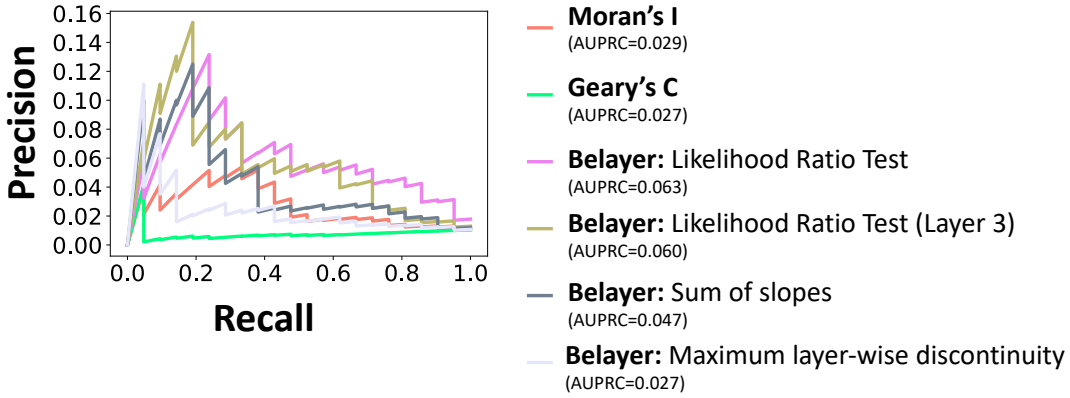

Figure S10: Precision-recall curves for identifying marker genes in DLPFC sample 151508 as in Figure 4A but with different rankings; see Section F for further explanation.

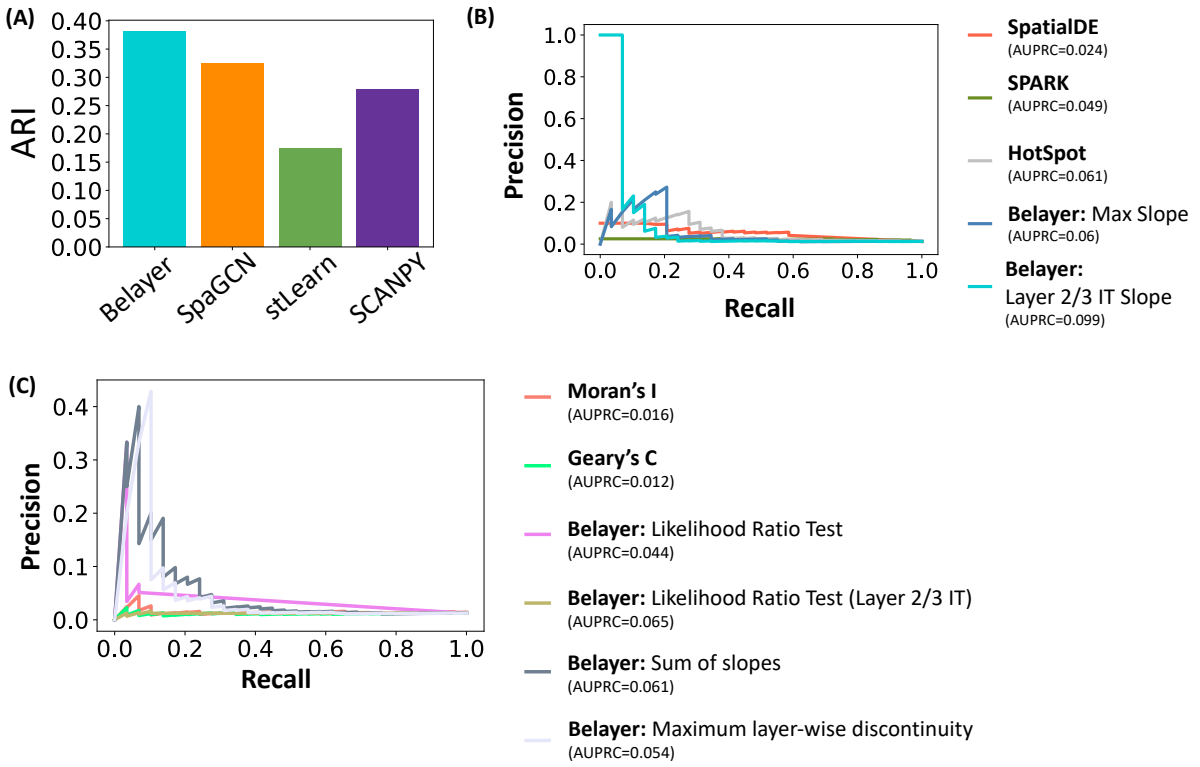

Figure S11: (A) Comparison of Belayer and other methods in identifying cortical layers in Slide-SeqV2 mouse somatosensory dataset. (B)/(C) Precision-recall curves for identifying marker genes in the same dataset using ten different approaches for ranking genes.

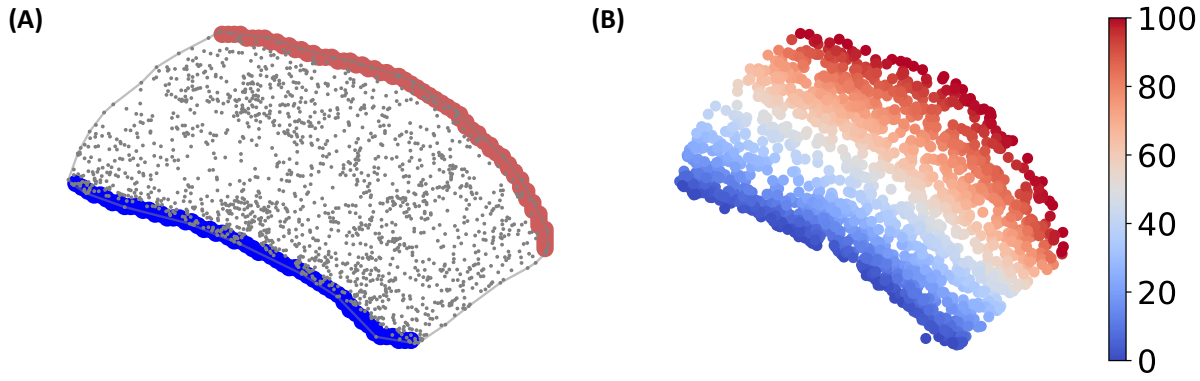

Figure S12: (A) Two parts of the tissue boundary are used to construct a conformal map over the entire tissue in the mouse somatosensory cortex dataset. Each spot is shown in spatial coordinate. Tissue boundary is outlined by a black segmented line; the blue and red points indicate the two involved parts respectively. (B) The real part of the conformal map is equivalent to the solution of heat equation whose values are shown by the colormap.

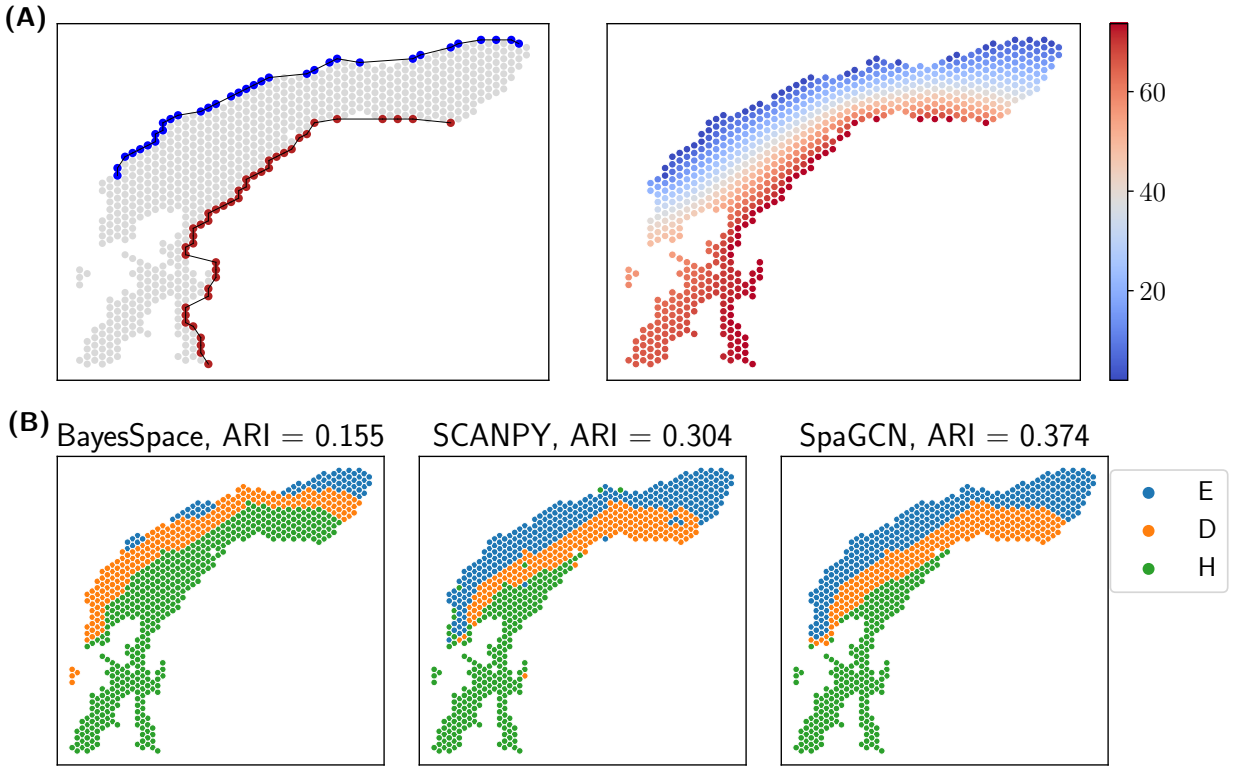

Figure S13: (A) Two parts of the tissue boundary are used to construct a conformal map over the entire tissue (left) in the skin wound dataset. Each spot is shown in spatial coordinate. Tissue boundary is outlined by a black segmented line; the blue and red points indicate the two involved parts respectively. The real part of the conformal map is equivalent to the solution of heat equation whose values are shown by the colormap (right). (B) Clusters identified by three other methods, BayesSpace, SCANPY, and SpaGCN. Clusters are labeled and colored by the maximal overlap with annotated skin layers.

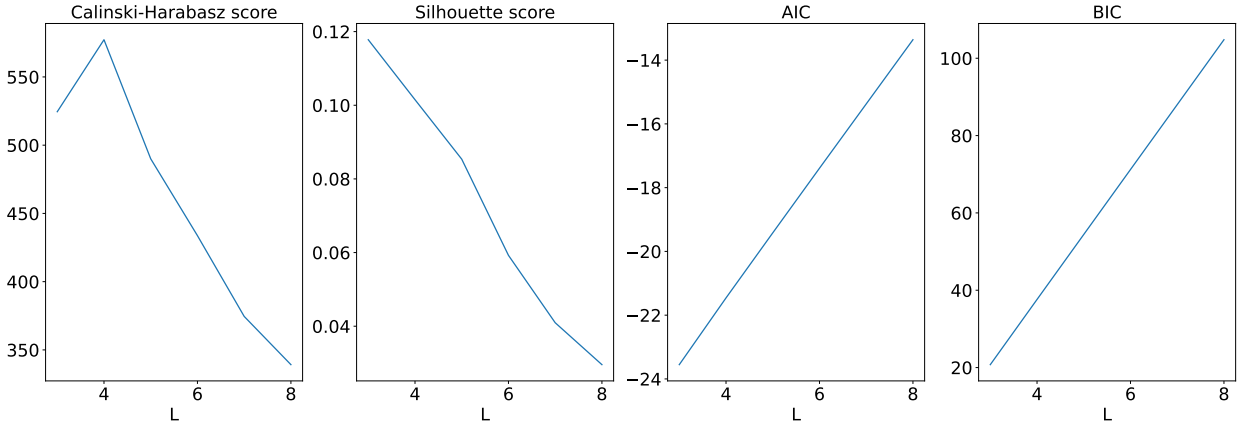

Figure S14: Alternative model selection procedures for choosing the number L of layers.