AMSI SUMMER RESEARCH SCHOLARSHIPS 2025–26

Get a taste for Research this Summer

Cell type deconvolution for spatial transcriptomics data at isoform resolution

Ashton Lu

Supervised by Dr Heejung Shim

The University of Melbourne

Jane Street®

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Abstract

Sequencing based Spatial Transcriptomics (ST) technology enables the measurement of gene and isoform expression directly within tissue samples to capture molecular information at pixel resolution while retaining the sample's original spatial arrangement. However since each pixel encompasses multiple cell types, this introduces a critical challenge in cell type deconvolution - how to accurately estimate cell type proportions for each spatial location. There are key biological implications in resolving this challenge as precise spatial mapping of cell types can advance biomedical research including predicting tumour spread. Current methodologies of cell type deconvolution either rely entirely on a single cell RNA sequencing (scRNA-seq) reference gene-expression profile or operate reference-free, using only the ST data to infer distinct cell types. Each approach carries inherent limitations such as the inability to infer new cell types absent in the reference for reference-based approaches or hindered detection of minor cell types due to low ST counts for reference-free methods. As such we introduce FlexiDeconv, a semi-supervised deconvolution framework that incorporates the scRNA-seq reference as priors within a Latent Dirichlet Allocation (LDA) model to guide inference while preserving the model's capacity to adapt to spatial data. This research project specifically focuses on extending FlexiDeconv to operate at the isoform-level, motivated by the hypothesis that isoform-level expression captures additional information beyond gene level counts to achieve finer granularity in cell type proportion estimates. We first applied FlexiDeconv under simulation settings where it was able to fully recover the ground truth cell type proportion before applying to real mouse brain ST data where gene-level and isoform-level deconvolution results were quite similar.

1 Introduction

Cell types form the functional base unit of biological tissue with each cell type possessing distinct specialised roles. Thus tissue samples often consist of a mixture of cell types to reflect the tissue's physiological needs. The spatial structure of these cell types underpins tissue function, raising the need for technology capable of measuring cell characteristics whilst preserving spatial arrangement. Consequently Spatial Transcriptomics (ST) short-read sequencing technology was developed to measure gene expression at pixel resolution for a given tissue sample (Figure 1).

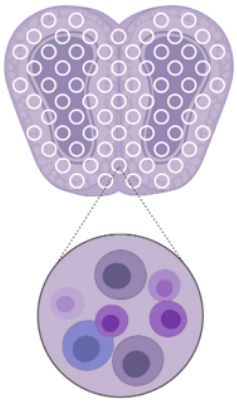

Figure 1: Spatial data visualisation from Miller et al. (2022)

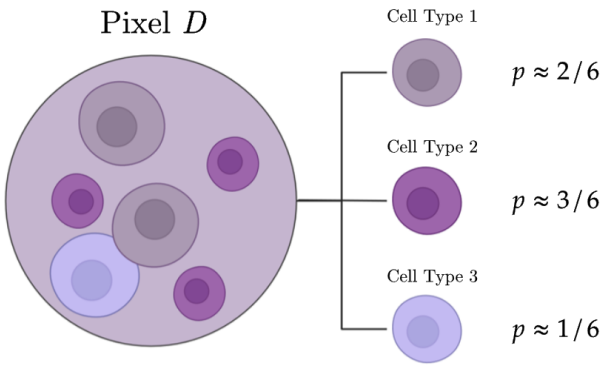

Figure 2: Cell type deconvolution diagram

However, quantifying RNA expression at the gene level fails to quantify the relative proportion of the underlying mRNA isoforms which may differ in function. This in turn drove the development of long-read Nanopore technology to mRNA expression at isoform-level granularity. Yet despite the granularity in mRNA isoform structure achieved through ST technology, the problem remains that isoform expression in each pixel originates from a mixture of multiple cell types (Figure 2). This introduces a key problem behind cell type deconvolution in appropriately classifying cell types for these mRNAs, which in turn affects the estimation of cell type proportions for each pixel.

Indeed, existing methodologies in this area can be broadly categorised as reference-based or reference-free approaches. In our case references are known cell type specific gene expression profiles obtained from single-cell RNA sequencing (scRNA-seq) of another tissue sample. As such, reference-based methods like Robust Cell Type Decomposition (RCTD) [1] fully rely on these reference gene signatures to infer cell type proportions. Conversely, reference-free approaches omit such reference datasets to infer cell types directly from the spatial data. One such reference-free approach is STdeconvolve [2] which treats the gene expression profile as a parameter updated through Latent Dirichlet Allocation (LDA) [3].

Yet the binary use of the reference dataset introduces inherent limitations to both methodologies. With reference-based approaches, the availability of appropriate, same-condition reference tissue is often limited and such approaches are by design unable to recover novel cell types in the ST data that are absent in the reference dataset. Likewise, reference-free approaches have lower sensitivity in detecting minor cell types with low spatial counts due to their entire reliance on the ST dataset.

To tackle these limitations, FlexiDeconv was proposed as a modification of the LDA model, where the reference acts as a prior in informing the ST dataset's cell type specific gene expression profile (Figure 3). In particular, this research project focused on extending FlexiDeconv's generative process to operate on isoform-level ST data with the intention of leveraging the extra layer of granularity in mRNA structure through long-read nanopore sequencing to provide more accurate cell type deconvolution results.

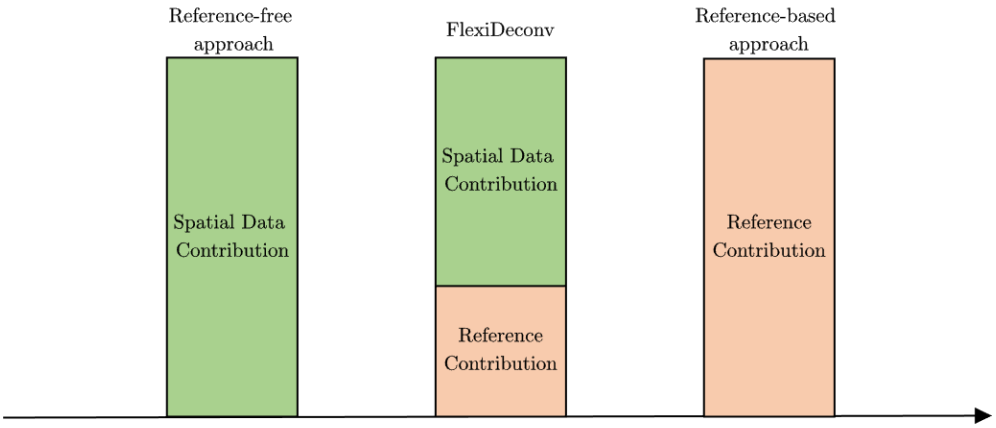

Figure 3: Visualisation of FlexiDeconv compared to reference-free and reference-based methods

1.1 Statement of Authorship

This research project was completed under the supervision of Dr Heejung Shim and assistance of Master of Data Science graduate Yichen Jiang. All gene-level mathematical derivations, program implementation and analysis were conducted by Yichen Jiang. All isoform-level program implementation and analysis were done with consistent guidance from my supervisor and Yichen Jiang throughout the duration of the project.

2 Input Datasets and Output

Two input datasets are required for FlexiDeconv - the ST data providing pixel-level counts data for an test tissue sample, and the scRNA-seq reference providing cell-type specific gene expression profile for a separate known tissue sample. The goal output for FlexiDeconv is to produce a pixel-specific cell type deconvolution profile for the ST dataset.

2.1 Spatial Transcriptomics

Short-read sequencing technology provides RNA expression at gene-level resolution. This dataset can be represented as a count matrix C of dimension $D \times G$, where D denotes the number of pixels and G the number of genes. Each entry C_{dg} records the RNA count of gene g detected in pixel d.

	Gene 1	Gene 2	…	Gene G	**Sum**
Pixel 1	5	0	…	6	**305**
Pixel 2	13	14	…	9	**274**
Pixel 3	8	2	…	13	**139**
⋮	⋮	⋮	⋱	⋮	⋮
Pixel D	11	5	…	10	**772**

Figure 4: Short-read gene expression profile for ST data

Long-read sequencing technology provides mRNA expression at isoform-level resolution. This dataset can be represented as count matrix C of dimension $D \times I$ where I denotes the total number of distinct isoforms across all genes. Each entry C_{di} similarly records the mRNA count for isoform i detected in pixel d.

	Isoform 1	Isoform 2	…	Isoform I	**Sum**
Pixel 1	2	3	…	0	**382**
Pixel 2	17	0	…	13	**342**
Pixel 3	8	4	…	4	**187**
⋮	⋮	⋮	⋱	⋮	⋮
Pixel D	7	5	…	9	**890**

Figure 5: Long-read isoform expression profile for ST data

2.2 Single Cell Reference

scRNA-seq provides the cell-type specific gene expression profile for a particular reference tissue sample. This dataset can be represented as a proportion matrix η of dimension $K \times G$ where K is the number of cell types and G is the number of genes. Hence each entry η_{kg} denotes the proportion of gene g occurring in cell type k. As such, each row of the matrix follows the row-wise constraint $\sum_{g=1}^{G} \eta_{kg} = 1$.

	Gene 1	Gene 2	…	Gene G
Cell type 1	0%	2.8%	…	6%
Cell type 2	11%	1%	…	0.5%
Cell type 3	2%	0%	…	2%
⋮	⋮	⋮	⋱	⋮
Cell type K	1%	1%	…	17%

Figure 6: Cell type specific gene expression profile for reference scRNA-seq data

Due to the potential gene expression profile inaccuracy for the reference, within FlexiDeconv η is instead incorporated as a prior in the inference step, allowing the spatial data to compensate for potential reference inaccuracies.

It is important to note that current scRNA-seq are limited to quantifying expression at the gene level and that isoform-resolution references are not yet obtainable. This presents a practical challenge when extending cell type deconvolution to isoform-level spatial data while the reference η remains at gene-level resolution. To address this resolution mismatch, our goal is to instead have our LDA model iteratively update the gene-specific isoform distribution using the ST dataset.

2.3 Output Cell Type Deconvolution

The output of cell type deconvolution can be represented as a proportion matrix θ of dimension $D \times K$ where θ_d represents the probability vector of the estimated cell type composition for pixel d. Hence each entry θ_{dk} denotes the estimated proportion of cell type k in pixel d. As such, each row of the matrix follows the row-wise constraint $\sum_{k=1}^{K} \theta_{dk} = 1$.

	Cell Type 1	Cell Type 2	…	Cell Type K
Pixel 1	p_{11}	p_{12}	$\cdots$	p_{1K}
Pixel 2	p_{21}	p_{22}	$\cdots$	p_{2K}
Pixel 3	p_{31}	p_{32}	$\cdots$	p_{3K}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Pixel D	p_{D1}	p_{D2}	$\cdots$	p_{DK}

Figure 7: Goal output pixel-specific cell type proportion profile

3 Methodology

The generative process behind FlexiDeconv is based upon the Latent Dirichlet Allocation (LDA) model [3] first proposed in the context of Natural Language Processing where the goal is to discover latent topics underlying a collection of documents.

In the original LDA paper [3], a word w represents the atomic unit of data, drawn from a vocabulary with index $\{1, 2, \ldots, V\}$. Each word w_n in a document is assumed to be generated from a single unobserved latent topic z_n. Furthermore, a document $\mathbf{w} = (w_1, \ldots, w_N)$ represents a collection of words with a corpus $\mathbf{M} = (\mathbf{w}_1, \ldots, \mathbf{w}_M)$ consisting of a collection of documents. Crucially, the model is described to be exchangeable - that is the order of words and documents within the corpus are irrelevant to the inference result, a property formalised by de Finetti's theorem.

When applied to ST data, we have that each pixel d in the tissue sample is analogous to a document, each mRNA is analogous to a word and each cell type k represents a latent topic variable. As such, two key inferential challenges arise from this correspondence: what is the distribution of cell types k within each pixel d, and can we accurately estimate the distribution of mRNA features for a given cell type k? These questions are the genetic parallel of recovering topic proportions per document and word-emission probabilities per topic - both unobserved latent quantities that LDA aims to recover.

3.1 Latent Dirichlet Allocation in Cell Type Deconvolution

For the sake of brevity, the following notation will be used throughout the report:

- D represents the number of pixels in the spatial dataset.
- K represents the number of cell types in the reference dataset.
- G represents the number of genes contained in both datasets.
- I represents the total number of isoforms present in the spatial dataset.
- T_g represents the number of isoforms in a given gene g in the spatial dataset.

As a probabilistic model the adapted LDA model can be described by the generative process below (Figure 8). For each pixel $d \in \{1, 2, \ldots, D\}$:

1. Generate $\theta_d \sim Dir(\alpha)$, where θ_d denotes the cell type proportion of pixel d and $Dir(\alpha)$ denotes the Dirichlet distribution with parameter α.
2. Supposed that there are M_d mRNA counts in pixel d, then for each molecule $m \in \{1, 2, \ldots, M_d\}$:
   (a) Given θ_d, assign cell type $z_{dm} | \, \theta_d \sim Categorical(\theta_d)$.
   (b) Given cell type z_{dm}, assign gene $w_{dm} \sim Categorical(\beta_{z_{dm}})$ where $\beta_{z_{dm}} = (\beta_{z_{dm}1}, \beta_{z_{dm}2}, \ldots, \beta_{z_{dm}G})$ represents the probability gene expression vector for cell type z_{dm}.
   (c) Given gene w_{dm} for cell type z_{dm}, assign isoform $i_{dm} \sim Categorical(f_{zw})$ where $f_{z_{dm}w_{dm}}$ represents probability isoform expression vector of gene w_{dm} for cell type z_{dm}.

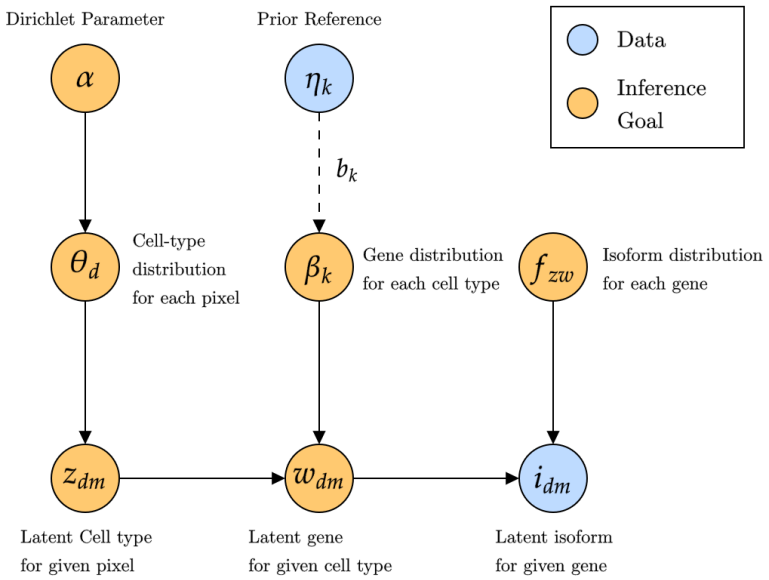

Figure 8: Generative process of LDA model applied in cell type deconvolution at isoform-level resolution with prior for β incorporated

The parameter assignment i_{dm} is given by the ST dataset mRNA count. Pixel specific cell type proportion parameter $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$, cell type specific gene expression parameter $\beta = (\beta_1, \beta_2, \ldots, \beta_K)$ and cell type assignment z_{dm}

were all estimated using Variational Inference in section 3.2. While parameters $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ and f_{zw} were iteratively updated as constants.

3.1.1 FlexiDeconv

Rather than treating β as a free parameter to be estimated from spatial data alone - FlexiDeconv introduces a Dirichlet prior η where η represents the cell type reference matrix in section 2.2 and η_{kg} represents the proportion of gene n in cell type k. Thus the prior on β_k for $k \in \{1, 2 \ldots, K\}$ is given by:

$$\beta_k \sim Dir(b_k \eta_k), \quad \text{where } b_k \in \mathbb{R}^+$$

Where b_k is a scalar hyperparameter controlling the influence of the prior. This choice is motivated by the role of b in scaling the Dirichlet concentration parameter to preserve the prior mean $\mathbb{E}[\beta_{kg}] = \eta_{kg}$ while scaling the variance and concentrating the density around the reference profile as b increases. Intuitively, this provides the representation that a large b anchors inference tightly around η cell-type specific gene expression, while a near-zero b value recovers approximately reference-free behaviour. As such β_k can thus be interpreted as the underlying cell type specific gene proportions.

3.2 Variational Inference

With the generative model specified in Section 3.1, the inference goal is to compute the posterior distribution over the latent variables θ, z and β given the observed count matrix. By Bayes' theorem, we define the target posterior probability as:

$$p(\theta, z, \beta \mid i, \alpha) = \frac{p(\theta, z, \beta, i \mid \alpha)}{p(i \mid \alpha)} = \frac{p(\theta, z, \beta, i \mid \alpha)}{\int_\beta \int_z \int_\theta p(\theta, z, \beta, i \mid \alpha)\, d\theta\, dz\, d\beta}$$

However due to the multi-dimensionality of (θ, z, β), the integral $\int \int \int_{\theta,z,\beta} p(\theta, z, \beta, i \mid \alpha)\, d\theta\, dz\, d\beta$ remains intractable - that is impossible to evaluate exactly. Hence Variational Inference (VI) seeks to approximate the target posterior using a tractible family of distribution Q. As such this transforms the inference problem into an optimisation problem of the Evidence Lower Bound (ELBO) as described in Appendix A which can be broadly interpreted as a loss function of the fitted target posterior from the counts data i.

3.3 Mean-Field Variational Family

In the FlexiDeconv model, we chose to use the Mean-Field Variational Family for the tractable family of distributions Q which is defined by the assumption of mutual independence between latent variables:

$$q(\theta, z, \beta) = \prod_{d=1}^{D} q(\theta_d) \times \prod_{d=1}^{D} \prod_{m=1}^{M_d} q(z_{dm}) \times \prod_{k=1}^{K} q(\beta_k)$$

The following variational distributions were chosen for each parameter:

- $q(\theta_d) = Dir(\gamma_d)$ where $d \in \{1, 2, \ldots, D\}$

- $q(z_{dm}) = Categorical(\phi_{dm})$ where $d \in \{1, 2, \ldots, D\}$ and $m = \{1, 2, \ldots, M_d\}$
- $q(\beta_k) = Dir(\tau_k)$ where $k \in \{1, 2, \ldots, K\}$

where γ, ϕ, τ are variational parameters updated at each iteration in addition to parameters α, f_{zw}. For the LDA model considered here the conjugate Dirichlet-Categorical structure ensures each factor's optimal update has a closed-form solution. Furthermore with the variational distributions specified above we can describe the ELBO function as follows:

$$\begin{aligned}
\text{ELBO} &= \mathbb{E}_q\left[\frac{\log p(\theta, z, \beta, w, i \mid \alpha, f)}{\log q(\theta, z, \beta)}\right] \\
&= \mathbb{E}_q[\log p(\theta, z, \beta, w, i \mid \alpha, f)] - \mathbb{E}_q[\log q(\theta, z, \beta)] \\
&= \mathbb{E}_q[\log p(\beta)] + \mathbb{E}_q[\log p(\theta \mid \alpha)] + \mathbb{E}_q[\log p(z \mid \theta)] + \mathbb{E}_q[\log p(w \mid z, \beta)] \\
&+ \mathbb{E}_q[\log p(i \mid z, w, f)] - \mathbb{E}_q[\log q(\theta)] - \mathbb{E}_q[\log q(z)] - \mathbb{E}_q[\log q(\beta)]
\end{aligned}$$

The detailed derivations can be found in Appendix B. We seek to give a brief description of the parameter update process for each parameter within the ELBO. Each expectation is differentiated with respect to the variational parameters involved before then solving for 0 to derive the update formula for each parameter. Meanwhile, α is updated via the Newton Raphson methods (Appendix B.6) due to the difficulty of a closed form analytical solution. Since isoform-level references do not exist, we cannot obtain individual isoform expression likelihoods $p(i)$ and so rather than imposing a distribution on f, we update it based upon its closed-form solution from the ST data to maximise the ELBO value.

3.4 Algorithm for parameter updates

As such, we present the pseudocode for the Variational Inference update procedure. Note that the introduction of parameter $\epsilon = 0.0001$ as a convergence criterion threshold and since Variational Inference optimises for the local maximum, multiple initialisations of the algorithm are necessary to obtain the highest ELBO solution. ϕ is defined within our model with subscript d, m, k which means that ϕ_{dmk} represents the likelihood that molecule m from pixel d belongs to cell type k. In practice however we can simplify this to ϕ_{dg_ik} which represents the likelihood that aa molecule in pixel d expressed as isoform g_i belongs to cell type k. Here g_i can be interpreted as an index for the i-th isoform in the g-th gene within our ST data.

To prevent overflow or underflow for likelihood parameters f and ϕ_{dg_ik}, we decided to update these parameters in the log-space to resolve numerical instability issues where these parameter updates are prematurely rounded to 0.

Algorithm 1 Variational Inference Parameter Update for FlexiDeconv

Input: R, C, ϵ

Initialize $\alpha \sim 50 \times Dir(v_1)$, v_1 denoted a uniform vector that gives sufficient variability in α

Initialize $\gamma_d \sim K \times Dir(v_2)$ for all d, v_2 denoted a uniform vector that gives sufficient variability in γ_d

Initialize $\phi_{dg_i} \sim Dir(v_3)$ for all d, g_i, v_3 denoted a uniform vector that gives sufficient variability in ϕ_{dg_i}

Initialize $\tau_k \sim K \times Dir(v_4)$ for all k, v_4 denoted a uniform vector that gives sufficient variability in τ_k

Initialize $f_{kg} \sim Dir(v_4)$ for all g, k, v_5 denoted a uniform vector that gives sufficient variability in f_{kg}

Convert f_{kg} and ϕ_{dg_i} into log-space parameters for numerical stability

while ΔELBO $> \epsilon$ **do**

  Update $\log f_{kg}(i) \propto \log \left[\sum_{d=1}^{D} \phi_{dg_ik} C_{dg_i}\right]$

  Update $\log \phi_{dg_ik} \propto \log\left[f_{kg}(i)\right] \cdot \left[\psi(\tau_{kg}) - \psi\left(\sum_{g=1}^{G} \tau_{kg}\right) + \psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right]$

  Update $\gamma_{dk} = \alpha_k + \sum_{g=1}^{G}\sum_{i=1}^{T_g} \phi_{dg_ik} \cdot C_{dg_i}$

  Update $\tau_{kg} = b_k R_{kg} + \sum_{d=1}^{D}\sum_{i=1}^{T_g} \phi_{dg_ik} \cdot C_{dg_i}$

  Update α using Newton-Raphson Method.

  Compute and store current ELBO.

end while

return α, normalized γ, ϕ, normalized τ and f

3.5 Interpretation for parameter updates

To align with the generative process, we provide interpretations for the updates of the parameters in our modified LDA model.

3.5.1 Interpretation for γ, τ

As γ_{dk} and τ_{kg} are both Dirichlet distribution parameters, their posterior update formulae can be thought of as a weighted sum of the prior and data counts. For γ_{dk}, the term $\sum_{g=1}^{G}\sum_{i=1}^{T_g} \phi_{dg_ik} \cdot C_{dg_i}$ represents the effective count of molecules assigned to cell type k in pixel d which can be interpreted as the information gained from the ST data. The prior component α_k represents the overall unscaled likelihood of cell type k appearing across a pixel. Likewise, the update logic for τ_{kg} has term $\sum_{d=1}^{D}\sum_{i=1}^{T_g} \phi_{dg_ik} \cdot C_{dg_i}$ which is the effective count of molecules assigned to gene g across all D pixels. The component $b_k R_{kg}$ can be interpreted as a scaled prior where the expression profile in the reference scRNA-seq dataset R_{kg} is scaled by the degree of trust in the reference denoted by hyperparameter b_k.

3.5.2 Interpretation for ϕ, f

Whilst calculated in the log-space, the interpretation for both parameters ϕ, f are most evident in the original probability-space. The update formula for $f_{kg}(i)$ is proportional to $\sum_{d=1}^{D} \phi_{dg_ik} C_{dg_i}$ which represents the effective total counts of isoform g_i assigned to cell type k across all D pixels. So the likelihood parameter $f_{kg}(i)$ can be interprated as the likelihood of isoform g_i in gene n for cell type k.

However the update formula for likelihood parameter ϕ remains more complicated. The update formula can be re-expressed as:

$$\phi_{dg_ik} \propto \exp\{\mathbb{E}_q[\log \theta_{dk}]\} \times \exp\{\mathbb{E}_q[\log \beta_{kg}]\} \times f_{kg}(i)$$

This new expression can be interpreted as the product of three probabilities: 1) molecule at pixel d belonging to cell type k; 2) that molecule in cell type k belonging to gene g; 3) that molecule from gene g belonging to isoform g_i which results in the likelihood that molecules expressed as isoform g_i in pixel d belongs to cell type k.

4 Results

To assess the performance of the proposed isoform-level model, we first test it against simulated ST datasets to see if FlexiDeconv can successfully recover the ground-truth cell type. When testing against real ST datasets, we benchmarked FlexiDeconv's performance using both short-read gene-level ST data and long-read isoform-level ST data to analyse whether added granularity in isoform structure would alter deconvolution outputs.

4.1 Simulation Studies

4.1.1 Data Simulation

The simulation uses $D = 256$ pixels arranged on a 16×16 grid, with a total of $K = 6$ cell types and $G = 120$ genes. Each gene has a maximum of 5 unique isoforms, where the number of isoforms for gene g is given by T_n.

Ground truth cell type composition. θ_d for each pixel d is drawn from a symmetric uniform Dirichlet distribution:

$$\theta_d \sim Dir(\mathbf{1}_K)$$

Ground truth gene-level expression profile. Each cell type k is characterised by a distinct block of 20 genes with elevated expression such that:

$$\beta_k \sim Dir(\alpha_k) \quad , \quad \alpha_{kg} = \begin{cases} 5 & \text{if } g \in [(20k-19), 20k] \\ 1 & \text{otherwise} \end{cases}$$

This ensures each cell type has a well-separated gene expression signature to provide unambiguous ground truth for validation.

Ground truth isoform-level expression profile. For each gene g in cell type k, isoform proportions are drawn independently from symmetric Dirichlet distribution with uniform concentration parameter $\frac{1}{T_g}$:

$$f_{kg} \sim Dir(\frac{1}{T_g}, \ldots, \frac{1}{T_g})$$

The choice of concentration parameter $\frac{1}{T_g}$ is reflective of genes with higher number of isoforms tend to sparser isoform distributions with lower concentration parameters.

Spatial count data generation. For each pixel d, the total molecule count is given by $M_d \sim \text{Poisson}(4000)$. Afterwards, each molecule is then generated through a three-stage samplig process consistent with our LDA generative model:

1. Given pixel d, cell type is sampled: $z \sim \text{Categorical}(\theta_d)$
2. Given that cell type z, a gene is sampled from the ground-truth cell type expression profile: $w \sim \text{Categorical}(\beta_z)$
3. Given that gene w, an isoform is sampled from the ground-truth isoform expression profile: $i \sim \text{Categorical}(f_{zw})$

The resulting isoform count C_{dg_i} is then incremented accordingly, yielding a $D \times I$ spatial count matrix at isoform reoslution

4.1.2 Result of Simulation Study

As we make no assumption on the spatial location of pixels nor do we incorporate any spatial priors in the generative process of our proposed model, deconvolution results would be identical even if the pixel positions were randomly reassigned. To begin analysis, we make a visual comparison of the deconvolved cell type proportion for FlexiDeconv against the ground truth cell type proportion using $vizAllTopics()$ function from STDeconvolve [2].

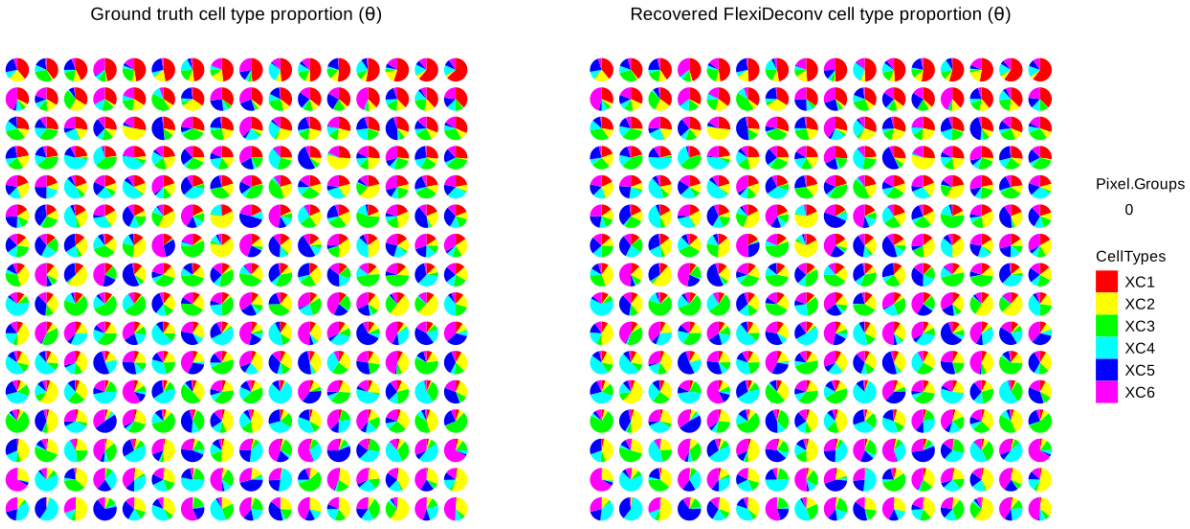

Figure 9: Ground truth compared to deconvolved cell type proportions for simulated ST dataset of $D = 256$ pixels arranged on a 16×16 grid for $K = 6$ cell types and $G = 120$ genes

From Figure 9, the estimated cell-type proportion θ_d is represented as an individual pie chart across all $D = 256$ pixels. The estimated proportion show strong visual agreement with the ground truth across all pixels, including those with mixed compositions involving all 6 cell types. FlexiDeconv managed to recover both the spatial arrangement and relative dominance of each cell type, suggesting its capability to recover pixel-resolution cell-type composition from isoform-level count data with a perfect gene-level reference in the simulation.

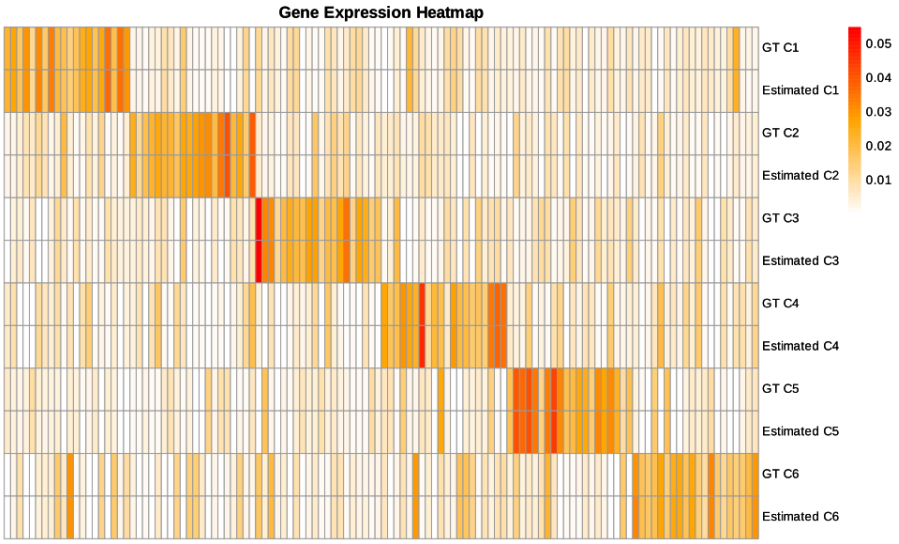

Figure 10: Ground truth compared to deconvolved gene expression heatmap for simulated ST dataset

Figure 10 similarly shows the gene expression heatmap comparing the ground truth and estimated column-wise gene expression for each row-wise cell type. The block structure of the underlying gene expression profile is clearly recovered, identifying the corresponding group of 20 dominant genes with elevated expression concentrated in the expected gene block and low expression elsewhere. Indeed, the close correspondence of cell-type specific gene expression profiles support that under noiseless conditions, FlexiDeconv can sufficiently estimate β_k from isoform-level spatial data.

4.2 Real Data Analysis

The isoform-level extension of FlexiDeconv is further evaluated on a real biological dataset from Lebrigand et al. (2023) [4] which provides long-read ST data from mouse brain tissue. In particular, mouse-brain tissue is known to contain closely related neuronal cell types whos transcriptional differences may be more apparent at isoform resolution than gene level. It is also worth noting the lack of availability in suitable long-read ST datasets in existing literature, making the dataset from [4] one of few applicable datasets. For this dataset there are $D = 918$ pixels with short-read ST data capturing gene-level expression across 31,053 genes while long-read ST data capturing isoform-level expression across 23,560 isoforms. The scRNA-seq reference is sourced from a separated matched dataset [4] with $N = 18,560$ genes and $K = 38$ cell types labelled in the metadata.

To test whether isoform-level spatial resolution yields more detailed cell type deconvolution, FlexiDeconv was applied in parallel to short-read gene-level (Gene SR) and long-read isoform-level (Isoform LR) ST data. For both gene-level and isoform-level implementations, a prior weight of $b_k = 2.0$ was used to reflect relatively strong trust in the scRNA-seq reference so that any differences in deconvolved results can be attributed to the differing resolution of the spatial count data rather than noise-driven inference.

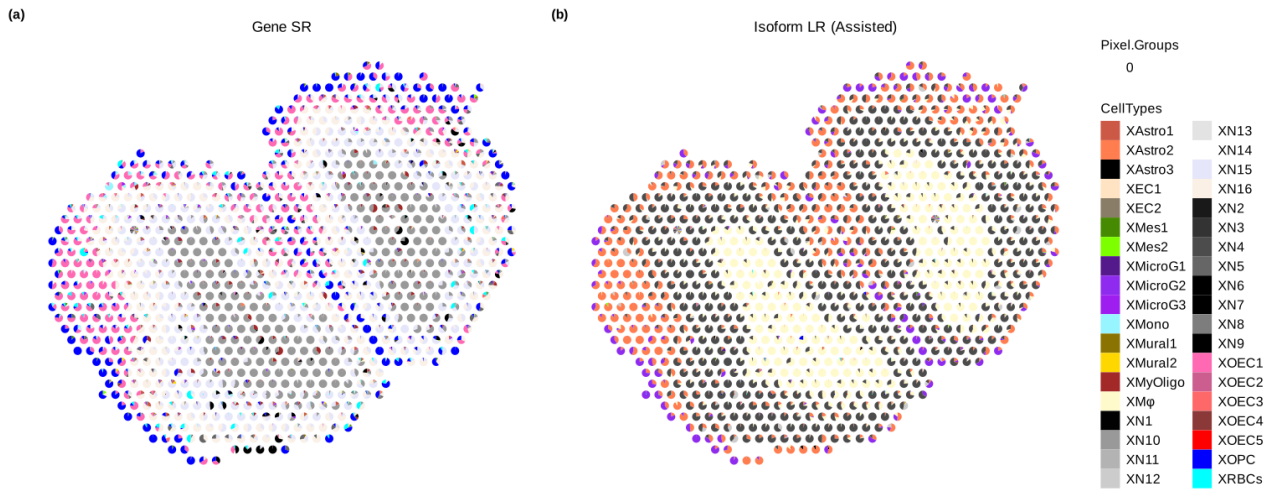

Figure 11: (a) Gene SR compared to (b) Isoform LR (highest ELBO) cell type deconvolution results

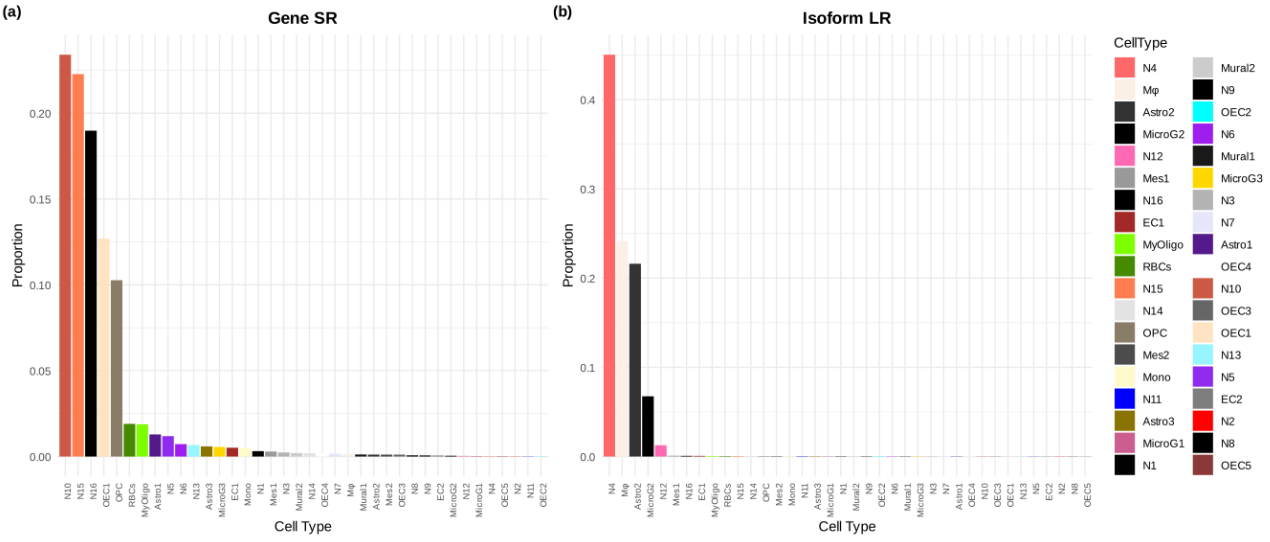

Figure 12: (a) Gene SR compared to (b) Isoform LR (highest ELBO) cell type proportion abundance results

By nature of VI algorithm (Section 3.2), the algorithm is run over five random initialisations to ensure the optimal ELBO is selected for comparison. Examining the deconvolved outputs in Figure 11 we see that both implementations broadly identified similar underlying cluster groupings across both initialisations. However the cell type labels clearly differ between the two implementations which is further supported when examining the proportion distriubution of estimated cell type abundances across all pixels in Figure 12. Both implementations exhibit similar proportion distributions dominated by selected major cell types. In particular, gene-level implementation recovers approximately 7 significantly expressed cell types, whereas the isoform-level implementation recovered only 5 major cell type proportions. This discrepancy may be attributable to differences in random initialisations combined with the enlarged

parameter space of the isofor-level implementation - thereby increasing the likelihood of ELBO convergence at a local maximum rather than global optimum for the isoform-level implementation.

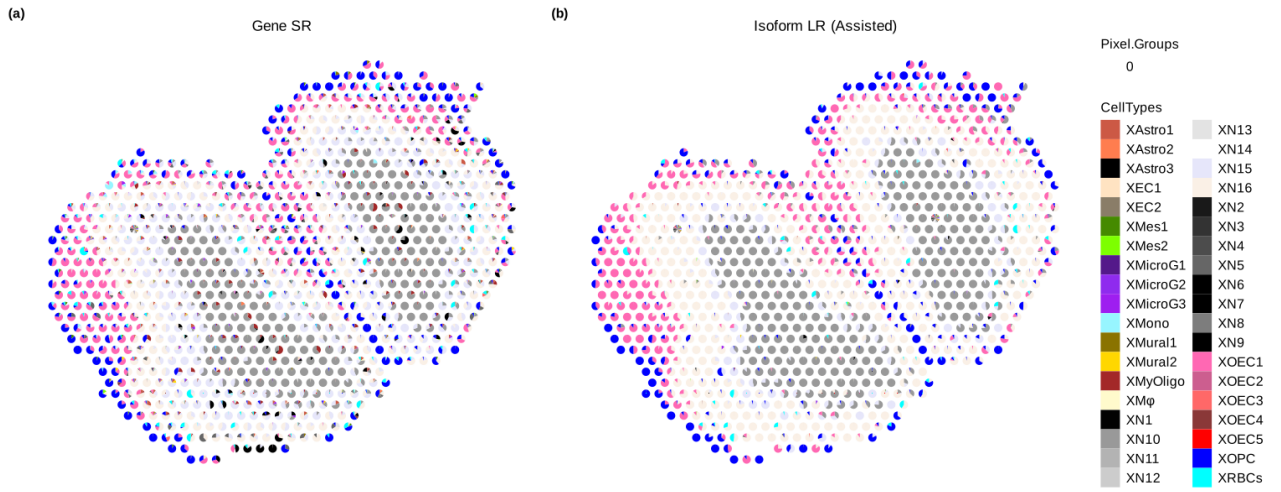

Figure 13: (a) Gene SR compared to (b) Isoform LR (assisted) cell type proportion abundance results

To address differences in cell type deconvolution being potentially caused by different parameter initialisations, we performed another run of the isoform-level implementation initialising the cell-type proportion variational parameter γ_d as the final converged estimate obtained from the gene-level run. Here we are providing the isoform-level algorithm with a more informed starting point, aiming to nudge inference towards a region of the parameter space already deemed meaningful by the gene-level implementation in Figure 13. Under this guided initialisation we see that the isoform-level run produces cell type classifications consistent with the gene-level estimate, thereby suggesting that the initialisation successfully guided the inferences towards a comparable solution while also highlighting the sensitivity of the final cell type label based upon the parameter initialisation values. However, the isoform-level implementation continues to recover only 5 major cell type clusters compared to the 7 cell types recovered at the gene level, raising two potential explanations: convergence instability arising from the large isoform-level parameter space, or that isoform-level count data fails to carry additional discriminative information beyond what is captured at the gene level in this particular dataset.

Thus to fully disentangle between these two explanations, we propose a targeted simulation study as the immediate next step. By constructing simulated datasets where the degree of isoform-level difference between cell types is varied, we can assess whether isoform-level implementation of FlexiDeconv is capable of capturing this signal across a spectrum of noise and identify the threshold of signal strength required for the model to yield meaningfully more detailed deconvolution than its gene-level counterpart. Indeed, this would provide a basis for interpreting subsequent real data analysis results and determining whether limitations observed are a consequence of the model methodology or data signalling.

5 Discussion

In this section we address the potential future directions implementable to the FlexiDeconv model.

Label instability and prior weight gene-specific extension. A practical challenge observed across repeated runs of FlexiDeconv on real biological data is the instability of cell type label assignments - the same underlying group of genes are frequently assigned different cell type labels across iterations. The persistence of label instability at low b values suggest that the prior influence $b\eta_k$ is too diffuse to meaningfully distinguish between cell types, resulting in the spatial data to dominate inference and result in cell type labels functioning as placeholder identifiers rather than biologically meaningful assignments. This observation motivates the potential extension towards developing gene-specific prior weights $\mathbf{b}_k \in \mathbb{R}^G$ to selectively strengthen the prior for known genes whose presence or absence within cell types are well established and shared across both datasets. Such modularity would be particularly valuable in cases where the reference and spatial data are derived from different tissues or experimental conditions and prior knowledge about gene-level reliability is available. As such, this extension would potentially stabilise label instability issues identified above without globally our prior weight b.

Isoform-level reference implementation. A key challenge in methodology addressed in this work is the resolution mismatch between isoform-level ST data and gene-level scRNA-seq reference data. Due to the absence of isoform-level single-cell sequencing technology, the current FlexiDeconv approach models the isoform distribution through the LDA topic modelling framework directly from the spatial data. However, as long-read single cell sequencing technologies mature and isoform-level scRNA-seq references become accessible, the incorporation of this more granular scRNA-seq reference becomes a natural extension to the FlexiDeconv methodology. Indeed, the gene specific isoform expression profile could be combined with or even replace current gene-level reference η to provide a more biologically informative prior on β, leading to more accurate cell type deconvolution results.

6 Conclusion

This research project presents an extension of FlexiDeconv to isoform-level spatial transcriptomics data, adapting the original LDA-based generative model [3] to incorporate long-read isoform-level count data while retaining gene-level scRNA-seq reference as a Dirichlet prior. Results suggest that under perfect simulation conditions FlexiDeconv was able to accurately recover cell type proportions. However for the mouse brain dataset [4], preliminary deconvolution results are unclear regarding if accuracy benefits exist in using isoform-level compared to gene-level ST data, and further simulation studies are required to investigate the recovery capabilities of isoform-level FlexiDeconv under varying signal strengths.

A Appendix: Evidence Lower Bound

To address the intractability of target posteriors $p(\theta|x)$, VI instead aims to approximate this distribution via a member $q(\theta)$ of a tractable family of distributions Q. The best member $q^*(\theta)$ of this family is found by minimising the Kullback-Leiber (KL) Divergence between the variational distribution and the target posterior:

$$q^*(\theta) = \arg\min_{q(\theta)\in Q} \mathsf{KL}(q(\theta)\,||\,p(\theta|x)) = \arg\min_{q(\theta)\in Q} \int_\Theta q(\theta)\log\frac{q(\theta)}{p(\theta|x)}\,d\theta$$

However due to intractible $\log p(\theta|x)$, we equivalent express this optimisation goal in terms of the Evidence Lower-Bound (ELBO) function:

$$\mathsf{ELBO}(q(\theta)) = \int_\Theta q(\theta)\log\frac{p(\theta,x)}{q(\theta)}\,d\theta$$

We can rewrite the KL Divergence as:

$$\mathsf{KL}(q(\theta)\,||\,p(\theta|x)) = -\mathsf{ELBO}(q(\theta)) + \log p(x)$$

Since $\log p(x)$ is treated as a constant and does not affect the optimisation result we have:

$$q^*(\theta) = \arg\min_{q(\theta)\in Q} \mathsf{KL}(q(\theta)\,||\,p(\theta|x)) = \arg\max_{q(\theta)\in Q} \mathsf{ELBO}(q(\theta))$$

B Appendix: Derivations

B.1 Exponential Family

Probability density of a distribution belonging to the exponential family can be expressed as:

$$p(x|\theta) = h(x)\exp\{\eta^T T(x) - A(\eta)\}$$

Where $\eta = f(\theta)$ represents the natural parameter (as a vector) of the distribution, $T(x)$ is a vector called sufficient statistics. Now we aim to compute :

$$\mathbb{E}[T(x)_i] = \int T(x)_i h(x)\exp\{\eta^T T(x) - A(\eta)\}dx$$

We can rearrange:

$$A(\eta) = \log\int h(x)\exp\{\eta^T T(x)\}dx$$

Hence:

$$\begin{aligned}\frac{\partial}{\partial\eta_i}A(\eta) &= \frac{\int h(x)T(x)_i\exp\{\eta^T T(x)\}dx}{\int h(x)\exp\{\eta^T T(x)\}dx}\\ &= \frac{\int h(x)T(x)_i\exp\{\eta^T T(x)\}dx}{\exp\{A(\eta)\}}\\ &= \mathbb{E}[T(x)_i]\end{aligned}$$

We can write:

$$\frac{\partial}{\partial\eta}A(\eta) = \mathbb{E}[T(x)]$$

B.2 Dirichlet Log Expectation

Based on B.1 we can now compute the special case for Dirichlet distribution, first note that:

$$\psi(z) = \frac{d}{dz}\log\Gamma(z)$$

Where $\psi(\cdot)$ is the digamma function, and $\Gamma(\cdot)$ is the gamma function.

Let $X \sim Dir(\alpha)$ with dimension k, hence:

$$\begin{aligned} p(x|\alpha) &= \frac{\Gamma(\sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \prod_{i=1}^k x_i^{\alpha_i - 1} \\ &= \exp\left\{ \sum_{i=1}^k (\alpha_i - 1)\log x_i + \log\Gamma(\sum_{i=1}^k \alpha_i) - \sum_{i-1}^k \log\Gamma(\alpha_i) \right\} \end{aligned}$$

- The sufficient statistic is $T(x) = [\log x_1, \log x_2, ..., \log x_k]^T$
- The natural parameter is $\eta = [\eta_1, \eta_2, ..., \eta_k] = [\alpha_1 - 1, \alpha_2 - 1, ..., \alpha_k - 1]^T$
- $h(x) = 1$
- $A(\eta) = -\log\Gamma(\sum_{i=1}^k \alpha_i) + \sum_{i-1}^k \log\Gamma(\alpha_i) = -\log\Gamma(\sum_{i=1}^k (\eta_i + 1)) + \sum_{i=1}^k \log\Gamma(\eta_i + 1)$

  Hence:

$$\frac{\partial}{\partial \eta_i} A(\eta) = -\psi\left(\sum_{i=1}^k (\eta_i + 1)\right) + \psi(\eta_i + 1)$$

From B.1, we can write:

$$\mathbb{E}[\log x_i] = \frac{\partial}{\partial \eta_i} A(\eta) = -\psi\left(\sum_{i=1}^k (\eta_i + 1)\right) + \psi(\eta_i + 1) = \psi(\alpha_i) - \psi\left(\sum_{i=1}^k \alpha_i\right)$$

B.3 Dirichlet Proof

Let $X \sim Dir(\alpha)$ with dimension k, $f_X(x)$ is the corresponding density function

$$\begin{aligned} \mathbb{E}\left[\log f_X(x)\right] &= \mathbb{E}\left[\log\left(\frac{\Gamma\left(\sum_{i=1}^k \alpha_i\right)}{\prod_{i=1}^k \Gamma(\alpha_i)} \prod_{i=1}^k x_i^{\alpha_i - 1}\right)\right] \\ &= \log\Gamma\left(\sum_{i=1}^k \alpha_i\right) - \sum_{i=1}^k \log\Gamma(\alpha_i) + \mathbb{E}\left[\sum_{i=1}^k (\alpha_i - 1)\log x_i\right] \\ &= \log\Gamma\left(\sum_{i=1}^k \alpha_i\right) - \sum_{i=1}^k \log\Gamma(\alpha_i) + \sum_{i=1}^k (\alpha_i - 1)\cdot\left(\psi(\alpha_i) - \psi\left(\sum_{j=1}^k \alpha_j\right)\right) \end{aligned}$$

Note that third line above uses result from B.3.

B.4 Evidence-Lower Bound

After incorporating the isoform data, the ELBO becomes:

$$\begin{aligned}
\text{ELBO} &= \int_{\Theta} q(\theta, z, \beta) \cdot \log \frac{p(\theta, z, \beta, w, i|\alpha, f)}{q(\theta, z, \beta)} d\Theta \\
&= \mathbb{E}_q \left[\log p(\theta, z, \beta, w, i|\alpha, f)\right] - \mathbb{E}_q \left[\log q(\theta, z, \beta)\right] \\
&= \mathbb{E}_q[\log p(\beta)] + \mathbb{E}_q[\log p(\theta|\alpha)] + \mathbb{E}_q[\log p(z|\theta)] + \mathbb{E}_q[\log p(w|z, \beta)] \\
&\quad + \mathbb{E}_q[\log p(i|z, w, f)] - \mathbb{E}_q[\log q(\theta)] - \mathbb{E}_q[\log q(z)] - \mathbb{E}_q[\log q(\beta)]
\end{aligned}$$

By using result in B.2, each term in the ELBO function is shown below:

$$\mathbb{E}_q \left[\log p(\beta)\right] = \sum_{k=1}^{K} \left[\log \Gamma\left(\sum_{g=1}^{G} b_k \eta_{kg}\right) - \sum_{g=1}^{G} \log \Gamma(b_k \eta_{kg}) + \sum_{g=1}^{G} (b_k \eta_{kg} - 1) \cdot \left(\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G} \tau_{kj}\right)\right)\right]$$

$$\mathbb{E}_q \left[\log p(\theta|\alpha)\right] = \sum_{d=1}^{D} \left[\log \Gamma\left(\sum_{k=1}^{K} \alpha_k\right) - \sum_{k=1}^{K} \log \Gamma(\alpha_k) + \sum_{k=1}^{K} (\alpha_k - 1) \cdot \left(\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right)\right]$$

$$\mathbb{E}_q \left[\log q(\theta)\right] = \sum_{d=1}^{D} \left[\log \Gamma\left(\sum_{k=1}^{K} \gamma_{dk}\right) - \sum_{k=1}^{K} \log \Gamma(\gamma_{dk}) + \sum_{k=1}^{K} (\gamma_{dk} - 1) \cdot \left(\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right)\right]$$

$$\mathbb{E}_q \left[\log q(\beta)\right] = \sum_{k=1}^{K} \left[\log \Gamma\left(\sum_{g=1}^{G} \tau_{kg}\right) - \sum_{g=1}^{G} \log \Gamma(\tau_{kg}) + \sum_{g=1}^{G} (\tau_{kg} - 1) \cdot \left(\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G} \tau_{kj}\right)\right)\right]$$

For $\mathbb{E}_q \left[\log p(z|\theta)\right]$ (also see B.3):

$$\begin{aligned}
\mathbb{E}_q \left[\log p(z|\theta)\right] &= \mathbb{E}_q \left[\log \left(\prod_{d=1}^{D} \prod_{m=1}^{M_d} \prod_{k=1}^{K} p(z_{dm} = k \mid \theta)^{\mathbb{I}(z_{dm}=k)}\right)\right] \\
&= \sum_{d=1}^{D} \sum_{m=1}^{M_d} \sum_{k=1}^{K} \mathbb{E}_q \left[\mathbb{I}(z_{dm} = k) \log \theta_{dk}\right] \\
&= \sum_{d=1}^{D} \sum_{m=1}^{M_d} \sum_{k=1}^{K} \phi_{dmk} \left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right] \quad \text{(parameter independence)} \\
&= \sum_{d=1}^{D} \sum_{m=1}^{M_d} \sum_{k=1}^{K} \sum_{g=1}^{G} \sum_{i=1}^{T_g} \phi_{dg_ik} \, \mathbb{I}(w_{dm} = g) \, \mathbb{I}(i_{dm} = g_i) \left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right] \\
&= \sum_{d=1}^{D} \sum_{k=1}^{K} \sum_{g=1}^{G} \sum_{i=1}^{T_g} \phi_{dg_ik} \left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right] \underbrace{\sum_{m=1}^{M_d} \mathbb{I}(w_{dm} = g) \, \mathbb{I}(i_{dm} = g_i)}_{= \, \text{count}_{dg_i}} \\
&= \sum_{d=1}^{D} \sum_{g=1}^{G} \sum_{i=1}^{T_g} \sum_{k=1}^{K} \phi_{dg_ik} \left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right] \text{count}_{dg_i} \\
&= \sum_{d=1}^{D} \sum_{g=1}^{G} \sum_{i=1}^{T_g} \text{count}_{dg_i} \sum_{k=1}^{K} \phi_{dg_ik} \left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K} \gamma_{dj}\right)\right]
\end{aligned}$$

In line 3 ϕ_{dmk} represents the likelihood (as proportion) that molecule m from pixel d belongs to cell type k. In line 4, this term is equivalently written as ϕ_{dg_ik}, the likelihood (as proportion) that molecules belonging to isoform g_i are from cell type k. In practice, ϕ_{dg_ik} is the way to model ϕ, because for molecules belonging to the same gene in the same pixel, there are no extra information to say that these molecules have different cell type decompositions. $\mathbb{I}(w_{dm} = g)$ is an indicator variable representing whether molecule m from pixel d belongs to gene g and $\mathbb{I}(i_{dm} = g_i)$ is an indicator variable representing whether molecule m from pixel d belongs to isoform g_i. C_{dg_i} represents the count of isoform g_i in pixel d in spatial data.

For $\mathbb{E}_q\left[\log p(w|z,\beta)\right]$ (also see B.3):

$$
\begin{aligned}
\mathbb{E}_q[\log p(w|z,\beta)] &= \mathbb{E}_q\left[\log\left(\prod_{d=1}^{D}\prod_{m=1}^{M_d}\prod_{k=1}^{K}\prod_{g=1}^{G} p(w_{dm}=g|z_{dm}=k,\beta)^{\mathbb{I}(z_{dm}=k)\mathbb{I}(w_{dm}=g)}\right)\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\mathbb{E}_q\left[\mathbb{I}(z_{dm}=k)\mathbb{I}(w_{dm}=g)\log p(w_{dm}^{g}=1|z_{dm}=k,\beta)\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\phi_{dmk}\mathbb{I}(w_{dm}=g)\left[\psi(\tau_{kg})-\psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)\left[\psi(\tau_{kg})-\psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right] \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\left[\psi(\tau_{kg})-\psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right]\sum_{m=1}^{M_d}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i) \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\left[\psi(\tau_{kg})-\psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right]\text{count}_{dg_i} \\
&= \sum_{d=1}^{D}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\sum_{k=1}^{K}\phi_{dg_ik}\left[\psi(\tau_{kg})-\psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right]
\end{aligned}
$$

For $\mathbb{E}_q[\log p(i|z,w,f)]$

$$\begin{aligned}
\mathbb{E}_q[\log p(i|z,w,f)] &= \mathbb{E}_q\left[\log\left(\prod_{d=1}^{D}\prod_{m=1}^{M_d}\prod_{k=1}^{K}\prod_{g=1}^{G}\prod_{i=1}^{T_g} p(i_{dm}=g_i|z_{dm}=k,w_{dm}=g)^{\mathbb{I}(z_{dm}=k)\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)}\right)\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\mathbb{E}_q[\mathbb{I}(z_{dm}=k)\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)\log f_{kg}(i)] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dmk}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)\log f_{kg}(i) \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)\log f_{kg}(i) \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\log f_{kg}(i)\sum_{m=1}^{M_d}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i) \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\log f_{kg}(i)\,\mathrm{count}_{dg_i} \\
&= \sum_{d=1}^{D}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\mathrm{count}_{dg_i}\sum_{k=1}^{K}\phi_{dg_ik}\log f_{kg}(i)
\end{aligned}$$

Finally for $\mathbb{E}_q[q(z)]$:

$$\begin{aligned}
\mathbb{E}_q\left[q(z)\right] &= \mathbb{E}_q\left[\log\left(\prod_{d=1}^{D}\prod_{m=1}^{M_d}\prod_{k=1}^{K} q(z_{dm}=k)^{\mathbb{I}(z_{dm}=k)}\right)\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\mathbb{E}_q\left[\mathbb{I}(z_{dm}=k)\log\phi_{dmk}\right] \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\phi_{dmk}\log\phi_{dmk} \\
&= \sum_{d=1}^{D}\sum_{m=1}^{M_d}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_n}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i)\phi_{dg_ik}\log\phi_{dg_ik} \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_n}\phi_{dg_ik}\log\phi_{dg_ik}\sum_{m=1}^{M_d}\mathbb{I}(w_{dm}=g)\mathbb{I}(i_{dm}=g_i) \\
&= \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\phi_{dg_ik}\log\phi_{dg_ik}\,\mathrm{count}_{dg_i} \\
&= \sum_{d=1}^{D}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\mathrm{count}_{dg_i}\sum_{k=1}^{K}\phi_{dg_ik}\log\phi_{dg_ik}
\end{aligned}$$

B.5 Parameter Update

For $f_{kg}(i)$:

$$\text{ELBO}[f] = \sum_{d=1}^{D}\sum_{k=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g} \phi_{dg_ik}\,\text{count}_{dg_i}\log\left[f_{kg}(i)\right] + \sum_{k=1}^{K}\sum_{g=1}^{G}\lambda_{kg}\left(\sum_{i=1}^{T_g} f_{kg}(i) - 1\right)$$

$$\frac{\partial \text{ELBO}[f]}{\partial f_{kg}(i)} = \sum_{d=1}^{D}\phi_{dg_ik}\,\text{count}_{dg_i}\,\frac{1}{f_{kg}(i)} + \lambda_{kg}$$

Solve $\frac{\partial \text{ELBO}[f]}{\partial f_{kg}(i)} = 0$:

$$-\phi_{dg_ik}\,\text{count}_{dg_i} = f_{kg}(i)\lambda_{dg}$$

$$f_{kg}(i) \propto \sum_{d=1}^{D}\phi_{dg_ik}\,\text{count}_{dg_i}$$

For ϕ_{dg_ik}:

$$\text{ELBO}[\phi] = \mathbb{E}_q[\log p(w|z,\beta)] + \mathbb{E}_q[\log p(z|\theta)] + \mathbb{E}_q[\log p(i|z,w,f) - \mathbb{E}_q(q(z))] + \sum_{d=1}^{D}\sum_{g=1}^{G}\lambda_{dg}\left(\sum_{i=1}^{T_g}\phi_{dg_ik} - 1\right)$$

$$\frac{\partial \text{ELBO}[\phi]}{\partial \phi_{dg_ik}} = \left[\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\tau_{kj}\right) + \psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right) - \log\phi_{dg_ik} - 1 + \log f_{kg}(i)\right]\text{count}_{dg_i}$$

$$+ \lambda_{dg} = 0$$

$$\log\phi_{dg_ik} + 1 = \left[\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\tau_{kj}\right) + \psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right) + \log f_{kg}(i)\right] - \frac{\lambda_{dg}}{\text{count}_{dg_i}}$$

$$\phi_{dg_ik} \propto f_{kg}(i)\left[\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\tau_{kj}\right) + \psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right)\right]$$

For τ_{kg}:

$$\text{ELBO}[\tau] = \sum_{k=1}^{K}\sum_{g=1}^{G}\left\{(b_k\eta_{kg} - 1)\cdot\left(\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right) + \sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik}\left[\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\gamma_{kj}\right)\right]\right\}$$

$$- \sum_{k=1}^{K}\left[\log\Gamma\left(\sum_{g=1}^{G}\tau_{kg}\right) - \sum_{g=1}^{G}\log\Gamma(\tau_{kg})) + \sum_{g=1}^{G}(\tau_{kg} - 1)\cdot\left(\psi(\tau_{kg}) - \psi\left(\sum_{j=1}^{G}\tau_{kj}\right)\right)\right]$$

$$\begin{aligned}\frac{\partial \text{ELBO}[\tau]}{\partial \tau_{kg}} &= (b_k\eta_{kg} - 1)\psi'(\tau_{kg}) - \sum_{i=1}^{N}(b_k\eta_{ki} - 1)\psi'\left(\sum_{j=1}^{G}\tau_{kj}\right) \\ &+ \psi'(\tau_{kg})\sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik} - \sum_{i=1}^{N}\sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik}\psi'\left(\sum_{j=1}^{G}\tau_{dj}\right) \\ &- \psi\left(\sum_{i=1}^{N}\tau_{ki}\right) + \psi(\tau_{kg}) - \psi(\tau_{kg}) + \psi\left(\sum_{i=1}^{N}\tau_{ki}\right) - (\tau_{kg} - 1)\psi'(\tau_{kg}) + \sum_{i=1}^{N}(\tau_{ki} - 1)\psi'\left(\sum_{j=1}^{G}\tau_{kj}\right) \\ &= \psi'(\tau_{kg})\left(b_k\eta_{kg} + \sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik} - \tau_{kg}\right) - \\ &\psi'\left(\sum_{j=1}^{G}\tau_{kj}\right)\sum_{i=1}^{N}\left(b_k\eta_{kg} + \sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik} - \tau_{ki}\right)\end{aligned}$$

Hence, solving $\frac{\partial \text{ELBO}[\tau]}{\partial \tau_{kg}} = 0$ gives

$$\tau_{kg} = b_k\eta_{kg} + \sum_{d=1}^{D}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik}$$

For γ_{dk}:

$$\begin{aligned}\text{ELBO}[\gamma] = &\sum_{d=1}^{D}\sum_{k=1}^{K}\left\{(\alpha_k - 1)\cdot\left(\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right)\right) + \sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik}\left[\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right)\right]\right\} \\ &- \sum_{d=1}^{D}\left[\log\Gamma\left(\sum_{k=1}^{K}\gamma_{dk}\right) - \sum_{k=1}^{K}\log\Gamma(\gamma_{dk})) + \sum_{k=1}^{K}(\gamma_{dk} - 1)\cdot\left(\psi(\gamma_{dk}) - \psi\left(\sum_{j=1}^{K}\gamma_{dj}\right)\right)\right]\end{aligned}$$

$$\begin{aligned}\frac{\partial \text{ELBO}[\gamma]}{\partial \gamma_{dk}} &= (\alpha_k - 1)\psi'(\gamma_{dk}) - \sum_{i=1}^{K}(\alpha_i - 1)\psi'\left(\sum_{j=1}^{K}\gamma_{dj}\right) \\ &+ \psi'(\gamma_{dk})\sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik} - \sum_{i=1}^{K}\sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ii}\psi'\left(\sum_{j=1}^{K}\gamma_{dj}\right) \\ &- \psi\left(\sum_{i=1}^{K}\gamma_{di}\right) + \psi(\gamma_{dk}) - \psi(\gamma_{dk}) + \psi\left(\sum_{i=1}^{K}\gamma_{di}\right) - (\gamma_{dk} - 1)\psi'(\gamma_{dk}) + \sum_{i=1}^{K}(\gamma_{di} - 1)\psi'\left(\sum_{j=1}^{K}\gamma_{dj}\right) \\ &= \psi'(\gamma_{dk})\left(\alpha_k + \sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik} - \gamma_{dk}\right) - \\ &\psi'\left(\sum_{j=1}^{K}\gamma_{dj}\right)\sum_{i=1}^{K}\left(\alpha_i + \sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ii} - \gamma_{di}\right)\end{aligned}$$

Hence, solving $\frac{\partial \text{ELBO}[\gamma]}{\partial \gamma_{dk}} = 0$ gives

$$\gamma_{dk} = \alpha_k + \sum_{g=1}^{G}\sum_{i=1}^{T_g}\text{count}_{dg_i}\,\phi_{dg_ik}$$

B.6 Newton-Raphson Method

Finally, update for α will be based on Newton-Raphson method [3]. It is an optimization technique which finds the maximum of a function $f(\alpha)$ by iterating:

$$\alpha_{i+1} = \alpha_i - H(\alpha_i)^{-1} g(\alpha_i)$$

Where $H(\alpha_i)$ and $g(\alpha_i))$ are the Hessian matrix and gradient vector respectively. If the Hessian matrix can be expressed as $H(\alpha_i) = diag(h) + \mathbf{1}z\mathbf{1}^\top$, where $diag(h)$ is a diagonal matrix with the diagonal elements being vector h, then the update formula can be simplified to:

$$(H(\alpha_i)^{-1} g(\alpha_i))_j = \frac{g_j - c}{h_j}$$

$$\text{where } c = \frac{\sum_{m=1}^{K} g_m/h_m}{z^{-1} + \sum_{m=1}^{K} h_m^{-1}}$$

References

[1] Dylan M. Cable, Evan Murray, Luli S. Zou, Aleksandrina Goeva, Evan Z. Macosko, Fei Chen, Rafael A. Irizarry, Dylan M. Cable, Evan Murray, Luli S. Zou, Aleksandrina Goeva, Evan Z. Macosko, Fei Chen, and Rafael A. Irizarry. Robust decomposition of cell type mixtures in spatial transcriptomics. *Nature Biotechnology 2021 40:4*, 40(4), 2021–02–18. ISSN 1546-1696. doi: 10.1038/s41587-021-00830-w. URL https://www.nature.com/articles/s41587-021-00830-w.

[2] Brendan F. Miller, Feiyang Huang, Lyla Atta, Arpan Sahoo, Jean Fan, Brendan F. Miller, Feiyang Huang, Lyla Atta, Arpan Sahoo, and Jean Fan. Reference-free cell type deconvolution of multi-cellular pixel-resolution spatially resolved transcriptomics data. *Nature Communications 2022 13:1*, 13(1), 2022–04–29. ISSN 2041-1723. doi: 10.1038/s41467-022-30033-z. URL https://www.nature.com/articles/s41467-022-30033-z.

[3] David Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 14, 2001.

[4] Kevin Lebrigand, Joseph Bergenstråhle, Kim Thrane, Annelie Mollbrink, Konstantinos Meletis, Pascal Barbry, Rainer Waldmann, and Joakim Lundeberg. The spatial landscape of gene expression isoforms in tissue sections. *Nucleic Acids Research*, 51(8), 2023/05/08. ISSN 0305-1048. doi: 10.1093/nar/gkad169. URL https://academic.oup.com/nar/article/51/8/e47/7079641495908890.