

# NLSDeconv: an efficient cell-type deconvolution method for spatial transcriptomics data

Yunlu Chen<sup>\*†</sup>, Feng Ruan<sup>\*</sup>, Ji-Ping Wang<sup>\*</sup>

June 13, 2024

## Abstract

**Summary:** Spatial transcriptomics (ST) allows gene expression profiling within intact tissue samples but lacks single-cell resolution. This necessitates computational deconvolution methods to estimate the contributions of distinct cell types. This paper introduces NLSDeconv, a novel cell-type deconvolution method based on non-negative least squares, along with an accompanying Python package. Benchmarking against 18 existing deconvolution methods on various ST datasets demonstrates NLSDeconv’s competitive statistical performance and superior computational efficiency.

**Availability and implementation:** NLSDeconv is freely available with tutorial at <https://github.com/tinachenc/NLSDeconv> as a Python package.

## 1 Introduction

Spatial transcriptomics (ST), crowned the Method of the Year in 2020 (Marx, 2021), has revolutionized biomedical research by allowing quantification of the mRNA expression of a large number of genes simultaneously in the spatial context of the tissue. The sequencing-based ST technology, while achieving its popularity for spatial profiling of gene expression across larger tissue areas with higher throughput than the image-based ST methods, lacks single-cell resolution. Consequently, the gene expression measured within a spot or a section of the tissue only reflects the average expression of a mixture of cell populations. Understanding the precise cell type composition within each spatial location is crucial for studying developmental biology and cancer biology. Thus, deconvoluting the cell type composition within each spot has become an imperative task in ST research.

Cell-type deconvolution in ST data often utilizes a set of reference gene expression profiles for known cell types, frequently obtained from scRNA-seq studies (Moses and Pachter, 2022). Many methods have been developed in recent years, including probabilistic-based methods (Andersson et al., 2020; Ma and Zhou, 2022; Kleshchevnikov et al., 2022; Lopez et al., 2022; Cable et al., 2022; Danaher et al., 2022); non-negative matrix factorization-based methods

---

<sup>\*</sup>Department of Statistics and Data Science, Northwestern University, Evanston, IL 60208, United States.

<sup>†</sup>Corresponding author. Department of Statistics and Data Science, Northwestern University, Evanston, IL 60208, United States. E-mail: [yunluchen2026@u.northwestern.edu](mailto:yunluchen2026@u.northwestern.edu)

(Rodriques et al., 2019; Dong and Yuan, 2021; Elosua-Bayes et al., 2021); and deep learning-based methods (Biancalani et al., 2021), etc. For excellent reviews of this topic, we refer to Chen et al. (2022); Li et al. (2023).

In this paper, we propose a novel method named NLSDeconv and a Python package based on the non-negative least squares method for cell-type deconvolution for ST data. We demonstrate the competitive performance of NLSDeconv relative to 18 existing methods by utilizing comprehensive benchmark datasets from both image-based and sequencing-based ST technologies, as well as the compelling efficiency of NLSDeconv.

## 2 Methods

Like many ST cell-type deconvolution methods, NLSDeconv takes two input datasets.

- An ST dataset containing RNA-seq read count for each gene at each spot within the tissue, denoted by  $\mathbf{Y}$  of size  $n_s \times n_g$  where  $n_s$  and  $n_g$  stand for the number of spots (spatial locations) and number of genes respectively.
- A reference single-cell RNA-seq (scRNA-seq) dataset containing scRNA-seq read count for each gene within each cell, denoted by  $\mathbf{X}$  of size  $n_c \times n_g$  where  $n_c$  stand for the number of cell *prototypes*. Each row of  $\mathbf{X}$  corresponds to a cell prototype, and each prototype belongs to one of  $K$  unique and known cell types, though multiple cell prototypes can belong to the same single cell type. Here, we assume  $\mathbf{X}$  and  $\mathbf{Y}$  share the same gene dimension  $n_g$ , but note that NLSDeconv can adjust for different dimensions through gene selection preprocessing.

The output of NLSDeconv is a cell-type deconvolution matrix  $\hat{\mathbf{P}}$  with each entry estimating the proportion of a cell *type* (not prototype) within a spot in the tissue.

The fundamental idea behind NLSDeconv is modeling the ST data’s gene read counts at a spot as a *weighted sum* of contributions from each cell prototype present. This is captured by a weight matrix  $\mathbf{M}$  of size  $n_s \times n_c$ , where each entry indicates the weight of cell prototypes at a spot. The relationship is formulated by a linear model:

$$\mathbf{Y} = \mathbf{MX} + \mathbf{E}. \tag{1}$$

where  $\mathbf{E}$  is a noise matrix of dimensions  $n_s \times n_g$ . This linear model captures how various cell prototypes contribute to the observed gene read counts across different spots, with  $\mathbf{M}_{ij} = 0$  indicating no contribution from the  $j^{\text{th}}$  cell prototype at the  $i^{\text{th}}$  spot.

In the field of ST cell-deconvolution, many existing linear models, such as those documented by Rodriques et al. (2019); Dong and Yuan (2021); Elosua-Bayes et al. (2021), typically regress  $\mathbf{Y}$  onto the cell type. In contrast, our approach regresses  $\mathbf{Y}$  onto the cell prototype  $\mathbf{X}$ . Our method (see below) essentially builds on top of least squares objectives to solve the linear model, which differs significantly from the method in (Biancalani et al., 2021). That method employs Kullback–Leibler divergence and cosine distance to learn the matrix  $\mathbf{M}$ , even though it similarly regresses  $\mathbf{Y}$  on the cell prototypes  $\mathbf{X}$ .

To estimate the weight matrix  $\mathbf{M}$  in our method, we employ Non-negative Least Squares (NLS) (Hastie et al., 2009). Our NLS estimate  $\widehat{\mathbf{M}}^{\text{NLS}}$  of  $\mathbf{M}$  is defined by:

$$\widehat{\mathbf{M}}^{\text{NLS}} = \underset{\mathbf{M} \geq 0}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{Y} - \mathbf{M}\mathbf{X}\|_F^2 + \frac{\lambda}{2} \|\mathbf{M}\|_F^2. \quad (2)$$

The Frobenius norm  $\|\cdot\|_F$  measures model fidelity,  $\lambda > 0$  is a ridge regularization parameter, and the constraint  $\mathbf{M} \geq 0$  requires all entries of the matrix  $\mathbf{M}$  to be nonnegative. The NLS objective is convex, allowing projected gradient descent to converge to its global minimum (Boyd and Vandenberghe, 2004). In practice, we typically set  $\lambda = 0$  when  $n_s \leq n_c$  and a small positive value,  $\lambda = 0.1$ , when  $n_s > n_c$ .

The method then calculates the cell-type deconvolution matrix  $\widehat{\mathbf{P}}^{\text{NLS}}$  using the weight matrix estimator  $\widehat{\mathbf{M}}^{\text{NLS}}$ . The proportion of cell type  $k \in \{1, \dots, K\}$  at spot  $i$  is calculated as follows:

$$\widehat{\mathbf{P}}_{ik}^{\text{NLS}} = \frac{\sum_{j \in \text{cell type } k} \widehat{\mathbf{M}}_{ij}^{\text{NLS}}}{\sum_j \widehat{\mathbf{M}}_{ij}^{\text{NLS}}}, \quad (3)$$

where the numerator is the sum of weights of cell type  $k$  at spot  $i$ , and the denominator is the total sum of weights across all cell types at that spot. This results in  $\widehat{\mathbf{P}}^{\text{NLS}}$ , an estimator for the cell-type deconvolution matrix.

In practice, computing  $\widehat{\mathbf{M}}^{\text{NLS}}$ , and consequentially  $\widehat{\mathbf{P}}^{\text{NLS}}$  for large datasets can be challenging due to the complexities of solving the Non-negative Least Squares (NLS) minimization as described in equation (2). To enhance computational efficiency, we propose a variant of the NLS method. This approach starts with the Ordinary Least Squares (OLS) estimator:

$$\widehat{\mathbf{M}}^{\text{OLS}} = \mathbf{Y}\mathbf{X}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} = \underset{\mathbf{M}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{Y} - \mathbf{M}\mathbf{X}\|_F^2. \quad (4)$$

However,  $\widehat{\mathbf{M}}^{\text{OLS}}$  does not guarantee nonnegative entries, so we apply the soft thresholding operator, defined by  $(z)_+ = \max\{z, 0\}$  for  $z \in \mathbb{R}$ , to ensure nonnegativity (Hastie et al., 2009). Specifically, we estimate the proportion of cell type  $k$  at a spot  $i$  by (cf. equation (3))

$$\widehat{\mathbf{P}}_{ik}^{\text{SLS}} = \frac{(\sum_{j \in \text{cell type } k} \widehat{\mathbf{M}}_{ij}^{\text{OLS}})_+}{\sum_k (\sum_{j \in \text{cell type } k} \widehat{\mathbf{M}}_{ij}^{\text{OLS}})_+}. \quad (5)$$

The soft thresholding ensures all entries of the cell-type deconvolution matrix  $\widehat{\mathbf{P}}^{\text{SLS}}$  to be nonnegative.

In conclusion, we have developed two distinct cell-type deconvolution matrix estimators,  $\widehat{\mathbf{P}}^{\text{NLS}}$  and  $\widehat{\mathbf{P}}^{\text{SLS}}$ . These two methods offer complementary approaches that are tailored to different computational budgets. In Section 4 below, we report our findings on the computational and statistical performance of these two estimators on benchmark ST datasets.

### 3 Software

We have developed a Python package NLSDeconv, which takes input data of all acceptable formats through *scanpy.read* function, e.g. h5ad, csv, text, h5, etc. The input data contains

(i) the scRNA-seq reference data file, including the read count and metadata with a key indicating cell type, and (ii) the ST data file, including the read count and metadata with spatial location  $x$  and  $y$  coordinates as columns.

**Preprocess** NLSDeconv provides a data preprocessing function step for scRNA-seq data. The class *Preprocessing()* requires scRNA-seq read count, ST read count, and the cell-type key for scRNA-seq data. It performs the three ordered steps: (i) normalize each cell by total counts over all genes (*cellcount\_norm*) (default option is True), and (ii) remove cells with cell types observed less than a number (*cellcount\_min*) (default number is 2), (iii) select a certain number of genes that can most characterize each cell type using differential expression analysis (*gene\_top*) (default number is 200). The outputs are the preprocessed ST data and scRNA-seq data.

**Deconvolution** Two functions of cell-type deconvolution are provided. The class *Deconv()* requires scRNA-seq read count, ST read count, and the cell-type key for scRNA-seq data.

- *.SLS()* is the command for performing SLS cell-type deconvolution. The outputs are the estimated cell-type composition matrix  $\hat{\mathbf{P}}^{\text{SLS}}$ , algorithm running time, and list of cell types corresponding to the column name of the resulting matrix.
- *.NLS()* is the command for performing NLS cell-type deconvolution. The required argument is learning rate (*lr*) (default is 0.1). Optional arguments are: ridge regularization parameter (*reg*), whether to use least square estimator as a warm start (*warm\_start*), number of epochs (*num\_epochs*), device for running the algorithm (*device*). The outputs are the estimated cell-type composition matrix  $\hat{\mathbf{P}}^{\text{NLS}}$ , algorithm running time, and list of cell types corresponding to the column name of the resulting matrix.

**Visualization** We provide two functions for the visualization of deconvolution results. The required inputs are ST data and the two outputs of the previous deconvolution step: cell-type composition matrix  $\hat{\mathbf{P}}^{\text{SLS}}$  or  $\hat{\mathbf{P}}^{\text{NLS}}$  and the list of cell types corresponding to the column name of the resulting matrix. There are optional arguments relating to display, allowing users to adjust their figures flexibly. See details in our document and tutorials.

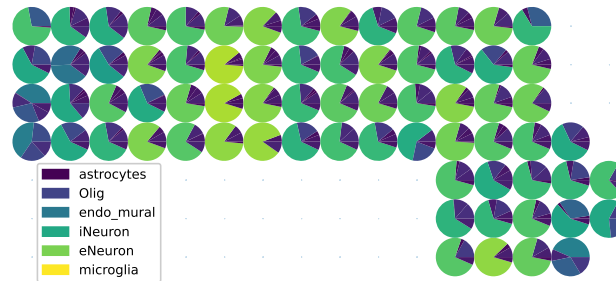
- *overall\_plt()* is the command for a spatial scatter pie plot displaying inferred cell-type composition on each location.
- *separate\_plt()* is the command for spatial proportion plots of given cell types.

In Figure 1, we show example visualization results of SLS on the seqFISH+(10000) dataset.

## 4 Results

We assess the performance of proposed methods by comparing them with 18 existing methods on diverse benchmarking datasets reported in the recent review paper (Li et al. (2023)).

(a)



(b)

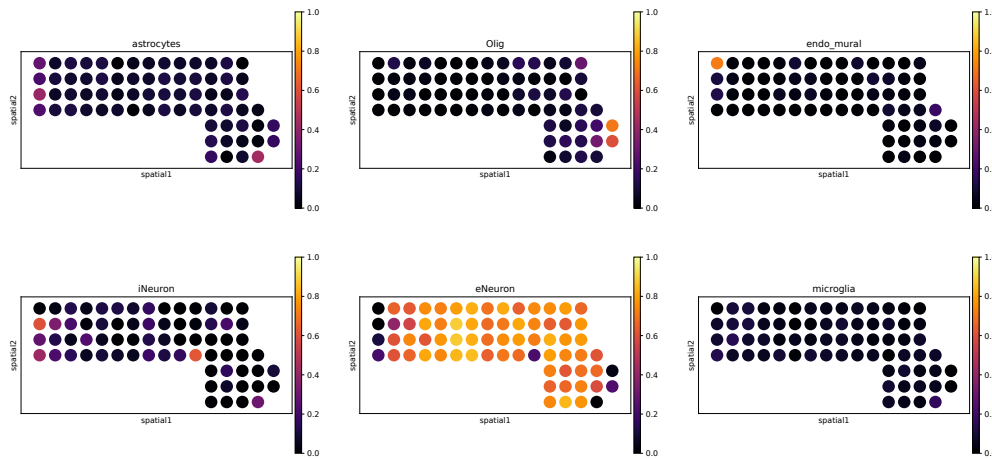


Figure 1: Example visualization results of SLS on the seqFISH+ (10000) dataset. (a) A spatial scatter pie plot displays inferred cell-type composition on spatial location (produced by the command `overall_plt()`). (b) The proportion of each cell type is displayed on spatial location (produced by the command `separate_plt()`).

The comprehensive datasets represent two image-based platforms, seqFISH+, MERFISH, and four sequencing-based platforms: Spatial Transcriptomics (ST), 10X Visium (Visium), Slide-seqV2, and stereo-seq. For seqFISH+ and MERFISH datasets, as the gene expression profile is at the single-cell level and cell type labels are known, we follow the exact approach by Li et al. (2023) to bin the neighboring cells into spots with different resolutions and use root-mean-square error (RMSE), Jensen-Shannon Divergence (JSD) to gauge the accuracy of deconvoluted cell type proportion. For the sequencing-based data sets where the true label is unknown, we also follow Li et al. (2023) to use Pearson Correlation Coefficient (PCC) between the deconvoluted cell type proportions within each spot and the expression of the corresponding marker genes of each cell type as the performance metric. Performance of other methods are ported from the Source Data file of Li et al. (2023).

For SLS, it is tuning-free. For NLS, we set  $\lambda = 0.1$  when  $n_s > n_c$ , and  $\lambda = 0$  when  $n_s \leq n_c$ . We set the learning rate to be 0.01, the number of epochs to be  $10^3$ , and use a warm start. We apply the codes provided by Li et al. (2023) for performance measures to avoid coding bias. Table 1 and 2 show the performance of our methods in comparison to other models on the image-based datasets with true labels. In general, our NLS has the lowest RMSE and JSD, followed by our SLS method, which can be viewed as a fast approximation version of NLS. For the sequencing-based datasets without true labels, to our surprise, the PCC measure suggests that SLS performs even better than NLS, ranking first overall out of all methods (Table 3).

One appealing feature of the SLS method is its computing efficiency. For a comparison, we pick three methods that were shown relatively more efficient in Li et al. (2023), including Tangram (Biancalani et al., 2021), RCTD (Cable et al., 2022) and SpatialDecon (Danaher et al., 2022). We consider benchmarking on two example datasets: seqFISH+ (10000), which represents datasets with large gene numbers and small spot numbers, and MERFISH (20), which represents datasets with large spot numbers and small gene numbers. We run SLS and the other three methods on a cluster node with Intel(R) Xeon(R) Gold 6338 CPU @ 2.0GHz and 256GB DDR4 2666MHz Memory with allocation of 1 CPU core and 100GB memory. We run NLS on Google Colab T4 GPU. The computing time is shown in Table 4. We note that Tangram is also computationally efficient. Nevertheless, its performance is less competitive. In conclusion, we recommend SLS for users with limited computation resources and NLS for those with high accuracy standards and a demand for code flexibility.

## Acknowledgments

Yunlu Chen thanks Hangyu Lin and Chenhao Zhang for helpful discussion on this project.

	seqFISH+			MERFISH			mean
	(10000)	(6000)	(3000)	(100)	(50)	(20)	
SLS	<b>0.109</b>	0.113	<b>0.126*</b>	0.133	0.181	0.250	<b>0.152</b>
NLS	<b>0.100</b>	<b>0.110</b>	<b>0.126*</b>	0.109	0.165	0.243	<b>0.142</b>
Berglund	0.272	0.267	0.280	0.177	0.177	0.286	0.243
CARD	0.126	0.129	0.141	0.188	0.188	0.264	0.172
Cell2location	0.242	0.243	0.240	0.109	<b>0.109</b>	0.247	0.198
DestVI	0.272	0.282	0.300	0.117	0.117	0.219	0.218
DSTG	0.227	0.260	0.292	0.231	0.231	0.311	0.259
NMFreg	0.296	0.291	0.291	0.180	0.180	0.310	0.258
RCTD	0.125	0.130	0.131	0.184	0.184	0.278	0.172
SD2	0.179	0.206	0.209	0.210	0.210	0.345	0.227
SpatialDecon	0.202	0.204	0.201	0.217	0.217	0.306	0.225
SpatialDWLS	0.112	<b>0.109</b>	<b>0.121</b>	0.213	0.213	0.525	0.215
SpiceMix	0.189	0.249	0.457	0.390	0.390	0.391	0.345
SPOTlight	0.262	0.257	0.256	0.169	0.169	0.318	0.239
STdeconvolve	0.343	0.325	0.326	0.220	0.220	0.247	0.280
stereoscope	0.223	0.226	0.228	0.193	0.193	0.274	0.223
STRIDE	0.260	0.273	0.232	0.124	0.124	0.372	0.231
Tangram	0.251	0.251	0.259	0.177	0.177	0.278	0.232
SpaOTsc	0.226	0.223	0.226	<b>0.104</b>	<b>0.115</b>	<b>0.151</b>	0.174
novoSpaRc	0.227	0.227	0.225	<b>0.096</b>	0.124	<b>0.162</b>	0.177

Table 1: RMSE of SLS and NLS methods compared with 18 existing methods on image-based datasets seqFISH+ and MERFISH. The numbers in the parenthesis of seqFISH+ indicate the numbers of genes that were randomly chosen in the dataset (We used the fixed random datasets given in [Li et al. \(2023\)](#)); those of MERFISH indicate the binning sizes (in  $\mu\text{m}$ ) for generating true labels. The last column is the mean of RMSE on all datasets. The top two methods are labeled in bold font. \* means equal RMSE when rounding to three decimal places.



	seqFISH+			MERFISH			mean
	(10000)	(6000)	(3000)	(100)	(50)	(20)	
SLS	0.117	0.097	<b>0.118*</b>	0.093	0.185	0.296	<b>0.151</b>
NLS	<b>0.090</b>	<b>0.086</b>	<b>0.118*</b>	<b>0.078</b>	0.172	0.270	<b>0.136</b>
Berglund	0.380	0.373	0.425	0.169	0.169	0.172	0.281
CARD	0.142	0.140	0.157	0.225	0.225	0.319	0.201
Cell2location	0.324	0.331	0.319	<b>0.082</b>	<b>0.082</b>	0.300	0.240
DestVI	0.443	0.474	0.519	0.089	0.089	<b>0.172</b>	0.298
DSTG	0.310	0.363	0.429	0.258	0.258	0.353	0.329
NMFreg	0.489	0.478	0.479	0.185	0.185	0.457	0.379
RCTD	0.143	0.145	0.140	0.156	0.156	0.242	0.164
SD2	0.216	0.261	0.277	0.221	0.221	0.528	0.288
SpatialDecon	0.275	0.279	0.270	0.236	0.236	0.282	0.263
SpatialDWLS	<b>0.064</b>	<b>0.073</b>	<b>0.096</b>	0.215	0.215	1.000	0.277
SpiceMix	0.241	0.352	0.827	0.687	0.687	0.473	0.545
SPOTlight	0.387	0.394	0.386	0.177	0.177	0.498	0.336
STdeconvolve	0.544	0.477	0.516	0.267	0.267	<b>0.065</b>	0.356
stereoscope	0.270	0.260	0.263	0.189	0.189	0.213	0.231
STRIDE	0.363	0.401	0.315	0.088	<b>0.088</b>	0.659	0.319
Tangram	0.364	0.356	0.372	0.141	0.141	0.319	0.282
SpaOTsc	0.648	0.638	0.654	0.167	0.268	0.462	0.473
novoSpaRc	0.647	0.626	0.633	0.233	0.352	0.491	0.497

Table 2: JSD of SLS and NLS methods compared with 18 existing methods on image-based datasets seqFISH+ and MERFISH. The numbers in the parenthesis of seqFISH+ indicate the numbers of genes that were randomly chosen in the dataset (We used the fixed random datasets given in [Li et al. \(2023\)](#)); those of MERFISH indicate the binning sizes (in  $\mu\text{m}$ ) for generating true labels. The last column shows the mean JSD on all datasets. The top two methods are labeled in bold font. \* means equal RMSE when rounding to three decimal places.



	ST	Visium	Slide-seqV2		stereo-seq		mean
			(11)	(17)	(gene)	(spot)	
SLS	<b>0.241</b>	0.732	<b>0.345</b>	<b>0.333</b>	0.392	0.303	<b>0.391</b>
NLS	0.144	0.499	0.226	0.190	0.522	0.463	0.341
Berglund	0.063	0.553	0.195	0.195	0.520	0.026	0.259
CARD	0.117	0.730	0.250	0.200	0.233	<b>0.471</b>	0.333
Cell2location	0.209	<b>0.793</b>	0.305	0.325	0.593	0.018	0.374
DestVI	0.081	0.642	-0.055	-0.070	0.537	0.449	0.264
DSTG	0.038	0.661	0.110	0.103	0.380	0.236	0.255
NMFreg	-0.103	0.496	0.190	0.243	0.533	0.400	0.293
novoSpaRc	-0.245	0.069	0.180	0.285	0.100	0.335	0.121
RCTD	0.160	0.011	0.305	0.320	<b>0.613</b>	0.462	0.312
SD2	0.033	0.585	0.075	0.100	0.410	0.149	0.225
SpaOTsc	-0.009	0.590	0.150	0.098	0.233	0.263	0.221
SpatialDecon	<b>0.295</b>	0.018	<b>0.320</b>	0.325	0.450	<b>0.506</b>	0.319
SpatialDWLS	-0.103	-0.008	-0.060	-0.025	-0.070	-0.021	-0.048
SpiceMix	-0.055	0.540	-0.065	-0.040	0.573	0.273	0.204
SPOTlight	0.125	0.750	0.270	0.233	0.360	0.433	0.362
STdeconvolve	-0.030	<b>0.760</b>	0.300	0.213	<b>0.603</b>	0.133	0.330
stereoscope	0.146	0.668	0.260	0.255	0.527	0.436	0.382
STRIDE	-0.062	0.642	0.235	0.205	0.543	0.434	0.333
Tangram	0.071	0.743	0.290	<b>0.365</b>	0.510	0.360	<b>0.390</b>

Table 3: Mean PCC of pairs of deconvoluted proportion of cell types and their corresponding marker genes from SLS and NLS methods in comparison with 18 existing methods on sequencing-based datasets ST, Visium, Slide-seqV2, and stereo-seq. The numbers in the parenthesis of Slide-seqV2 indicate the cell-type numbers of the dataset after the sub-cell types are integrated and the original dataset; the names in the parenthesis of stereo-seq indicate the original dataset and integrated subcellular-resolution dataset. The last column is the overall mean of PCC on all datasets. The top two methods are labeled in bold font.

	seqFISH+ (10000)	MERFISH (20)
SLS	<b>2.1</b>	<b>6.5</b>
NLS	48.9	<b>17.2</b>
Tangram	<b>4.0</b>	56.3
RCTD	87.0	1178.5
SpatialDecon	10.4	92.4

Table 4: Running time (seconds) on seqFISH+(10000) and MERFISH(20) datasets. We show the top two methods in bold.

## References

- ANDERSSON, A., BERGENSTRÅHLE, J., ASP, M., BERGENSTRÅHLE, L., JUREK, A., FERNÁNDEZ NAVARRO, J. and LUNDEBERG, J. (2020). Single-cell and spatial transcriptomics enables probabilistic inference of cell type topography. *Communications biology*, **3** 565.
- BIANCALANI, T., SCALIA, G., BUFFONI, L., AVASTHI, R., LU, Z., SANGER, A., TOKCAN, N., VANDERBURG, C. R., SEGERSTOLPE, Å., ZHANG, M. ET AL. (2021). Deep learning and alignment of spatially resolved single-cell transcriptomes with tangram. *Nature methods*, **18** 1352–1362.
- BOYD, S. P. and VANDENBERGHE, L. (2004). *Convex optimization*. Cambridge university press.
- CABLE, D. M., MURRAY, E., ZOU, L. S., GOEVA, A., MACOSKO, E. Z., CHEN, F. and IRIZARRY, R. A. (2022). Robust decomposition of cell type mixtures in spatial transcriptomics. *Nature biotechnology*, **40** 517–526.
- CHEN, J., LIU, W., LUO, T., YU, Z., JIANG, M., WEN, J., GUPTA, G. P., GIUSTI, P., ZHU, H., YANG, Y. ET AL. (2022). A comprehensive comparison on cell-type composition inference for spatial transcriptomics data. *Briefings in Bioinformatics*, **23** bbac245.
- DANAHER, P., KIM, Y., NELSON, B., GRISWOLD, M., YANG, Z., PIAZZA, E. and BEECHEM, J. M. (2022). Advances in mixed cell deconvolution enable quantification of cell types in spatial transcriptomic data. *Nature communications*, **13** 385.
- DONG, R. and YUAN, G.-C. (2021). Spatialdwl: accurate deconvolution of spatial transcriptomic data. *Genome biology*, **22** 145.
- ELOSUA-BAYES, M., NIETO, P., MEREU, E., GUT, I. and HEYN, H. (2021). Spotlight: seeded nmf regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic acids research*, **49** e50–e50.
- HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. H. and FRIEDMAN, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer.
- KLESHCHEVNIKOV, V., SHMATKO, A., DANN, E., AIVAZIDIS, A., KING, H. W., LI, T., ELEMENTAITE, R., LOMAKIN, A., KEDLIAN, V., GAYOSO, A. ET AL. (2022). Cell2location maps fine-grained cell types in spatial transcriptomics. *Nature biotechnology*, **40** 661–671.
- LI, H., ZHOU, J., LI, Z., CHEN, S., LIAO, X., ZHANG, B., ZHANG, R., WANG, Y., SUN, S. and GAO, X. (2023). A comprehensive benchmarking with practical guidelines for cellular deconvolution of spatial transcriptomics. *Nature Communications*, **14** 1548.
- LOPEZ, R., LI, B., KEREN-SHAUL, H., BOYEAU, P., KEDMI, M., PILZER, D., JELINSKI, A., YOFE, I., DAVID, E., WAGNER, A. ET AL. (2022). Destvi identifies continuums of cell types in spatial transcriptomics data. *Nature biotechnology*, **40** 1360–1369.

- MA, Y. and ZHOU, X. (2022). Spatially informed cell-type deconvolution for spatial transcriptomics. *Nature biotechnology*, **40** 1349–1359.
- MARX, V. (2021). Method of the year: spatially resolved transcriptomics. *Nature methods*, **18** 9–14.
- MOSES, L. and PACTER, L. (2022). Museum of spatial transcriptomics. *Nature methods*, **19** 534–546.
- RODRIGUES, S. G., STICKELS, R. R., GOEVA, A., MARTIN, C. A., MURRAY, E., VANDERBURG, C. R., WELCH, J., CHEN, L. M., CHEN, F. and MACOSKO, E. Z. (2019). Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science*, **363** 1463–1467.