Accurate estimation of rare cell type fractions from tissue omics data via hierarchical deconvolution

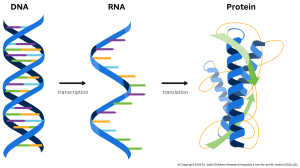
Penghui Huang

Department of Biostatistics
University of Pittsburgh

2023 ICSA Applied Statistics Symposium
6/14/2023

Biology background

- The central dogma
  - transcription: making an RNA copy of a gene's DNA sequence
  - quantity of RNA transcript → how vigorously a gene is expressed
- Tissue omics data
  - gene expression at tissue level
  - mixture of multiple cell types
- Single-cell data
  - gene expression at cell level
  - mostly with cell type annotations


DNA
RNA
Protein
transcription
translation
© Copyright 2022 St. Jude Children's Research Hospital, a not-for-profit, section 501(c)(3)

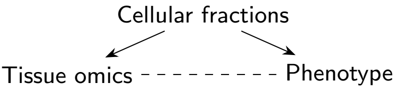
@St.Jude Children's Research Hospital

Cell Type A
Cell Type B
Bulk RNA Seq
SCRNA Seq

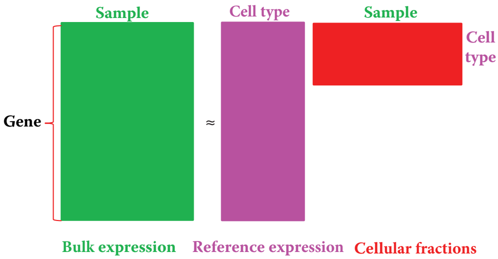
@Honeycomb Biotechnologies

Motivation

- Cell type fractions
  - can confound tissue-level analyses (Jaffe et al., *Genome biology*, 2014).



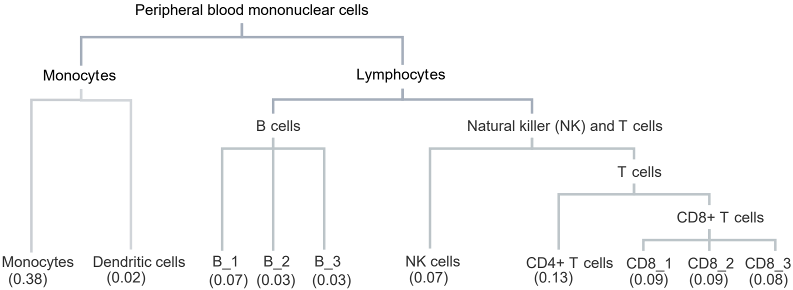
  - can be measured by biochem methods e.g. flow cytometry and immunohistochemistry, yet they are both labor-intensive and costly.
- Cellular deconvolution – *in silico* flow cytometry
  - assumes the tissue level gene expression is the sum of cell type gene expression weighted by their fractions



Motivation

- All good? Not really.
  - Cell types of low abundance yet important in biology.
  - Collinearity in reference matrix because of cell types that share the same origin in differentiation.

- Hierarchical cell type tree
  - Available from biology, hierarchical clustering, etc.
  - Tree guided top-down deconvolution approaches: HEpiDISH and MuSiC.



Penalty and constraints

- Incorporate hierarchical cell type tree via penalty terms: the fraction of the parent cell type ≈ the sum of fractions of children cell types

  $$\boldsymbol{p}_{il} \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} \boldsymbol{p}_{i(l+1)} = \boldsymbol{B}_{l,(l+1)} \boldsymbol{p}_{i(l+1)},$$

  layer l: monocytes and lymphocytes; layer $l+1$: lymphocytes $\rightarrow$ B cells and T cells.
- Cellular fractions
  - Non-negativity.
  - Sum-to-1 constraints for each sample's cellular fractions: it works better if fractions are normalized after estimation.

Hierarchical deconvolution (HiDecon)

Estimate cellular fractions with constraints from “parent” and “children” cell types

$$\underset{\substack{p_{i1},\dots,p_{iL} \\ p_{il} \geq 0,\ \|p_{iL}\|_1 = 1}}{\text{argmin}} \left\{ \sum_{l=1}^{L} \frac{1}{m_l} \|\boldsymbol{x}_{il} - \boldsymbol{A}_l \boldsymbol{S}_l \boldsymbol{p}_{il}\|_2^2 + \lambda \sum_{l=1}^{L-1} \frac{\|\boldsymbol{p}_{il} - \boldsymbol{B}_{l,(l+1)} \boldsymbol{p}_{i(l+1)}\|_2^2}{K_l} \right\},$$

- p: cellular fractions (to be estimated)
- i: bulk sample
- $l = 1, \dots, L$: layer in the hierarchical tree
- m_l: number of marker genes in layer l
- $\boldsymbol{x}$: bulk data
- $\boldsymbol{A}$: signature matrix
- $\boldsymbol{S}$: cell size (a known diagonal matrix)
- λ: tuning parameter
- $\boldsymbol{B}_{l,(l+1)}$: mapping matrix between layer l and $l+1$
- K_l: number of cell types in layer l

Estimation algorithm

- The previous objective function has multiple vectors to be optimized with respect to. It is not a standard optimization problem.
- Rewrite the model with all vectors stacked up.

$$\operatorname*{argmin}_{p_i \in \mathbb{R}^K_{\geq 0}} f(\boldsymbol{p}_i), \quad f(\boldsymbol{p}_i) = \frac{1}{2}\|\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{A}}\boldsymbol{p}_i\|_2^2 + \frac{\lambda}{2}\|\tilde{\boldsymbol{B}}\boldsymbol{p}_i\|_2^2,$$

where $K = \sum_{l=1}^{L} K_l$, $\tilde{x}_i = (m_1^{-1/2} x_{i1}^\top, \ldots, m_L^{-1/2} x_{iL}^\top)^\top \in \mathbb{R}^m_{\geq 0}$, and $\tilde{\boldsymbol{A}} = \bigoplus_{l=1}^{L} (m_l^{-1/2} A_l S_l) \in \mathbb{R}^{m \times K}_{\geq 0}$ for $m = \sum_{l=1}^{L} m_l$. The matrix $\tilde{B} \in \mathbb{R}^{(K-K_L)\times K}_{\geq 0}$ is an upper-triangular difference operator taking the form

$$\tilde{B} = \left(\bigoplus_{l=1}^{L-1} K_l^{-1/2} I_{K_l}, 0_{(K-K_L)\times K_L} \right) - \left(0_{(K-K_L)\times K_1}, \bigoplus_{l=1}^{L-1} K_l^{-1/2} B_{l,(l+1)} \right).$$

Estimation algorithm

- Convex optimization with non-negativity constraint.
- We employ coordinate-wise descent algorithm for the optimization purpose.
- Convexity ensures convergence and the existence of Hessian ensures fast convergence.

```
Data: $\boldsymbol{b} = \tilde{\boldsymbol{A}}^T \tilde{\boldsymbol{x}}_i$, $\boldsymbol{H} = \tilde{\boldsymbol{A}}^\top \tilde{\boldsymbol{A}} + \lambda \tilde{\boldsymbol{B}}^\top \tilde{\boldsymbol{B}}$, and $\epsilon > 0$
Result: $\boldsymbol{p}_i$
Initialize $\boldsymbol{p}_i = \boldsymbol{H}^{-1}\boldsymbol{b}$;
if $\boldsymbol{p}_i \geq 0$ then
    return $\boldsymbol{p}_i$;
else
    $\boldsymbol{p}_{ik} = \max(0, \boldsymbol{p}_{ik})$, $k \in \{1, \cdots, K\}$;
    repeat
        $\boldsymbol{p}_{ik} = \max[0, \{\boldsymbol{b}_k - \boldsymbol{p}_{i(-k)}^\top \boldsymbol{H}_{k(-k)}\}/\boldsymbol{H}_{kk}]$, $k \in \{1, \ldots, K\}$;
    until $|(\boldsymbol{H}\boldsymbol{p}_i - \boldsymbol{b})_k| \leq \epsilon$ OR $\{(\boldsymbol{H}\boldsymbol{p}_i - \boldsymbol{b})_k \geq 0$ AND $\boldsymbol{p}_{ik} = 0\}$ for all
     $k \in \{1, \ldots, K\}$ //Karush–Kuhn–Tucker (KKT) conditions;
    return $\boldsymbol{p}_i$;
end
```

Evaluation metrics

- Mean absolute error:

$$\mathrm{MAE}\left(\boldsymbol{P}, \hat{\boldsymbol{P}}\right) = avg\left(|\boldsymbol{P} - \hat{\boldsymbol{P}}|\right)$$

- Lin's concordance correlation coefficient:

$$\mathrm{CCC}\left(\boldsymbol{P}_{k*}, \hat{\boldsymbol{P}}_{k*}\right) = \frac{2cov\left(\boldsymbol{P}_{k*}, \hat{\boldsymbol{P}}_{k*}\right)}{\sigma^2_{\boldsymbol{P}_{k*}} + \sigma^2_{\hat{\boldsymbol{P}}_{k*}} + \left(avg\left(\boldsymbol{P}_{k*} - \hat{\boldsymbol{P}}_{k*}\right)\right)^2}$$

It is a more comprehensive concordance evaluation statistic than correlation coefficient.

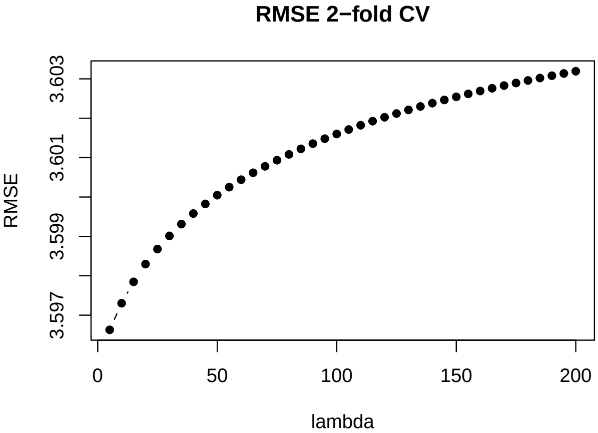
Tuning parameter selection

$$f(\boldsymbol{p}_i) = \frac{1}{2}\|\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{A}}\boldsymbol{p}_i\|_2^2 + \frac{\lambda}{2}\|\tilde{\boldsymbol{B}}\boldsymbol{p}_i\|_2^2$$

- Although HiDecon is a regression type problem, our focus is different from a typical regression.
  - Regression is more focused on how well we can predict (fit) the response.
  - Cellular deconvolution is focused on the precision of cellular fraction (regression coefficient) estimates.
- If we use cross-validation, where we use predicted bulk data for performance evaluation, the evaluation process is evaluating K-dimensional cellular fraction using the low dimensional representation (response/bulk) as the surrogate for it.
- Highly correlated genes are equally split into training and test sets. Thus, training set and test set carry similar gene information. The CV RMSE will be increasing.

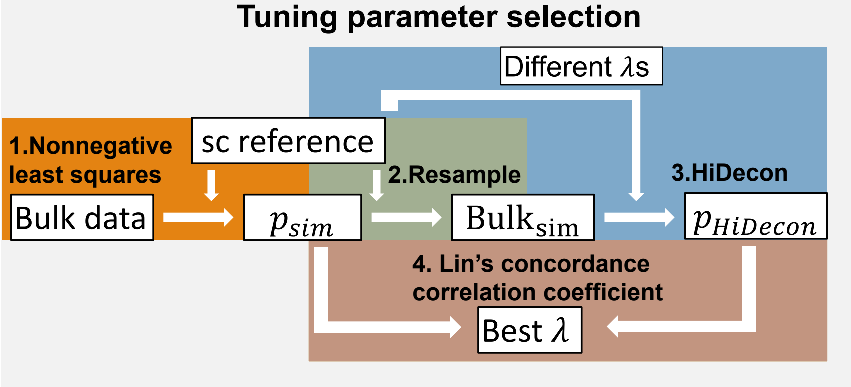
Tuning parameter selection

- In our simulation, we used 2-fold cross-validation and calculated the prediction RMSE under different parameters.
- CV fails to select the proper parameter.



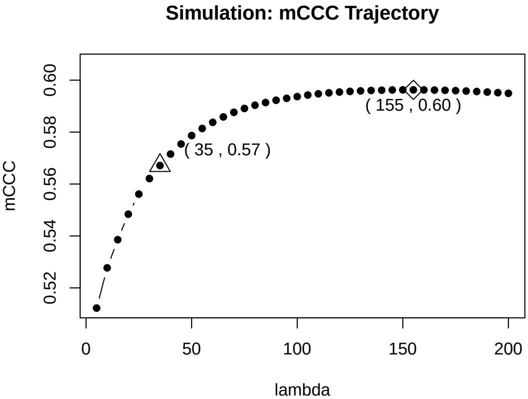
Tuning parameter selection

- Intuition: imitate bulk data structure by resampling from single cell reference using the initial fraction guess.


Tuning parameter selection
Different λs
1.Nonnegative least squares
sc reference
2.Resample
3.HiDecon
Bulk data
p_{sim}
Bulk_{sim}
$p_{HiDecon}$
4. Lin's concordance correlation coefficient
Best λ

Tuning parameter selection

- In our simulation, we tested HiDecon under a series of parameters and compare fraction estimates with ground truth to find the best parameter (diamond).
- We use our proposed method to choose the best parameter (triangle).



Simulation: benchmarking

- Data source: real COVID-19 scRNA-seq PBMC data (Ren et al., *Cell*, 2021).
- Simulated bulk data: averaged across single cells within sample (126 samples).
- Reference: all individual level single cell data used for simulating bulk data.

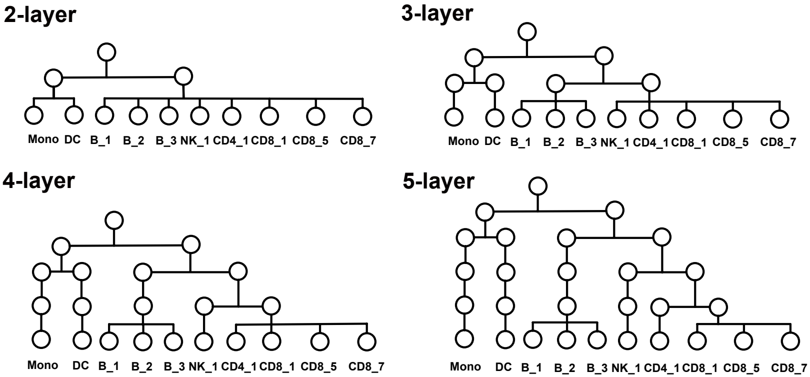
Simulation: benchmarking

- First 10 columns show CCC for each cell type
- Mean abundances are presented in parentheses (can be only of 2%)
- NA denotes all zero estimates

	Mono	DC	B_1	B_2	B_3	NK_1	CD4_1	CD8_1	CD8_5	CD8_7	Mean	MAE
	(0.38)	(0.02)	(0.07)	(0.03)	(0.03)	(0.07)	(0.13)	(0.09)	(0.09)	(0.08)	CCC	
HiDecon	0.85	0.25	**0.80**	**0.54**	**0.50**	0.34	**0.62**	0.66	0.52	0.64	**0.57**	**0.05**
CIBERSORT	0.88	**0.31**	0.66	0.31	0.37	0.01	0.42	**0.76**	0.25	0.25	0.42	0.07
dtangle	0.35	0.08	0.56	0.18	0.30	**0.47**	0.51	0.51	**0.60**	**0.69**	0.43	0.06
MuSiC	**0.89**	0.01	NA	NA	0.16	NA	NA	0.35	0.22	0.39	0.20	0.08

Simulation: more tree info, better performance

- Use previous COVID-19 simulation data
- More information: trees are extended more to get to the leaf cell types



Simulation: more tree info, better performance

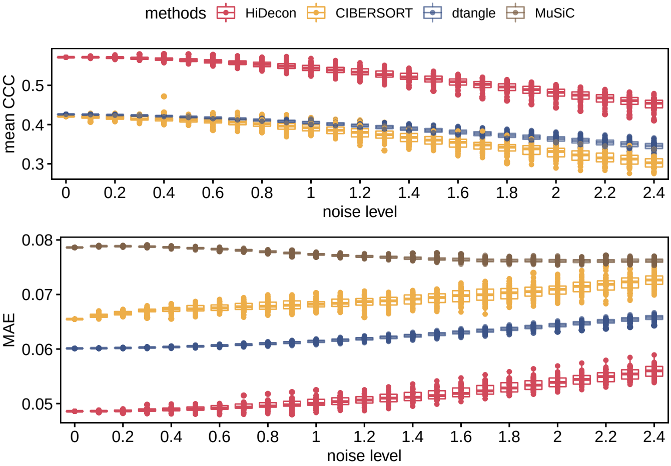
- Increasing mean CCC and decreasing MAE.
- Appreciable performance boost in NK_1, CD4_1 and CD8_1.

	Mono	DC	B_1	B_2	B_3	**NK_1**
	(0.38)	(0.02)	(0.07)	(0.03)	(0.03)	(0.07)
2layer	0.76	0.19	0.83	0.51	0.51	0.04
3layer	0.82	0.23	0.80	0.55	0.50	0.05
4layer	0.83	0.25	0.80	0.55	0.50	0.31
5layer	0.85	0.25	0.80	0.54	0.50	0.34

	CD4_1	**CD8_1**	CD8_5	CD8_7	Mean	MAE
	(0.13)	(0.09)	(0.09)	(0.08)	CCC	
2layer	0.21	0.45	0.49	0.55	0.45	0.063
3layer	0.24	0.48	0.49	0.57	0.47	0.060
4layer	0.25	0.47	0.52	0.63	0.51	0.057
5layer	0.62	0.66	0.52	0.64	0.57	0.049

Simulation: robustness evaluation

- Increasing normal noise added to simulated bulk data with standard deviation ranging from 0 to that of bulk data
- 50 replication per noise level setting.



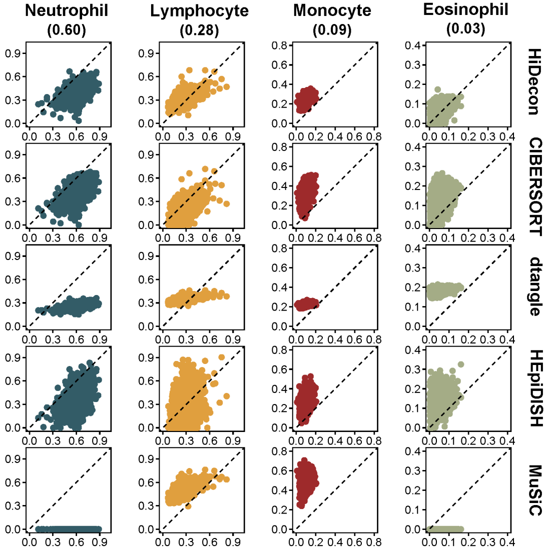
Real data benchmarking

- Using FHS blood microarray data and cell counts data as ground truth
- Reference: LM22 microarray dataset
- First 4 columns show CCC for each cell type
- Mean abundances are presented in parentheses

	Neutrophil (0.60)	Lymphocyte (0.28)	Monocyte (0.09)	Eosinophil (0.03)	Mean CCC	MAE
HiDecon	0.13	**0.57**	**0.04**	**0.28**	**0.26**	**0.10**
CIBERSORT	**0.15**	0.31	0.02	0.06	0.13	0.15
dtangle	0.02	0.17	0.01	0.01	0.05	0.17
HEpiDISH	0.12	0.25	0.03	0.04	0.11	0.17
MuSiC	NA	0.08	0.00	NA	0.02	0.32

Real data benchmarking

- x-axis: ground truth, y-axis: estimates



Summary

- We developed HiDecon to incorporate a hierarchical cell type tree to facilitate the estimation of related and rare cell types
- HiDecon can incorporate complex tree structure
- There's no universally best deconvolution method. When you are in trouble, you may try HiDecon!
- Links
  - bioRxiv: doi.org/10.1101/2023.03.15.532820
  - GitHub: github.com/randel/HiDecon

Acknowledgements

- Faculty member
  - Jiebiao Wang (Biostatistics)
  - Chris McKennan (Statistics)
- PhD student
  - Manqi Cai (Biostatistics)

Thank you!

Questions or suggestions?
huangpenghui@pitt.edu

References I

Jaffe, A. E., & Irizarry, R. A. (2014). Accounting for cellular heterogeneity is critical in epigenome-wide association studies. *Genome biology*, *15*(2), 1–9.

Ren, X., Wen, W., Fan, X., Hou, W., Su, B., Cai, P., Li, J., Liu, Y., Tang, F., Zhang, F., et al. (2021). Covid-19 immune features revealed by a large-scale single-cell transcriptome atlas. *Cell*, *184*(7), 1895–1913.