Sample-wise unsupervised deconvolution of complex tissues

Lulu Chen[1], Chia-Hsiang Lin[2,*], Chiung-Ting Wu[1,*], Robert Clarke[3], Guoqiang Yu[1], Jennifer E. Van Eyk[4], David M. Herrington[5], and Yue Wang[1,#]

* Equal contribution

[1]Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Arlington, VA 22203, USA; [2]Department of Electrical Engineering, National Cheng Kung University, Tainan City, Taiwan 70101; [3]The Hormel Institute, University of Minnesota, Austin, MN 55912; [4]Advanced Clinical Biosystems Research Institute, Cedars Sinai Medical Center, Los Angeles, CA 90048, USA; [5]Department of Internal Medicine, Wake Forest University, Winston-Salem, NC 27157, USA

Article Format:	bioRxiv
Running title:	Sample-wise unsupervised deconvolution
Open source software:	R code is available at https://github.com/Lululuella/swCAM

[#]Author for correspondence: Yue Wang, Ph.D.
Virginia Polytechnic Institute and State University
900 N. Glebe Road, Arlington, VA 22203
E-mail: yuewang@vt.edu

Abstract

We report a sample-wise fully unsupervised deconvolution method, namely sample-wise Convex Analysis of Mixtures (swCAM), that can estimate constituent proportions and subtype-specific expressions in individual samples using tissue-level bulk data (Chen 2019). The swCAM software tool enables statistically-principled subtype-level downstream analyses, such as detecting subtype-specific differentially expressed genes (sDEG) and differential dependency networks (DDN) (Zhang, Li et al. 2009, Chen, Lu et al. 2020). Significantly different from population-level deconvolution, individual-level deconvolution is mathematically an underdetermined problem because there are more variables than observations. We therefore extend the existing CAM framework by adding an extra term of between-sample variations and formulate swCAM as a nuclear-norm regularized low-rank matrix factorization problem (Wang, Hoffman et al. 2016). We determine hyperparameter value by random entry exclusion based cross-validation scheme and obtain swCAM solution using a modified efficient alternating direction method of multipliers (ADMM). Experimental results on realistic simulation data sets show that swCAM can accurately perform sample-wise unsupervised deconvolution of complex tissues and successfully recover subtype-specific correlation networks that are otherwise unobtainable using existing methods.

Introduction

Decades of research on molecule regulatory mechanisms have provided a rich framework with which we can extract molecule expression patterns to gain insight into the organization and structure of the large biological networks (Zhang and Horvath 2005, Zhang, Li et al. 2009, Tian, Zhang et al. 2014). However, most discoveries are concluded from measured molecule expressions in heterogeneous tissues, in which the underlying changes of constituent components could obscure molecule regulations and corrupt network inference that only occur in particular tissue subtypes. The inference of subtype-specific molecule expression patterns becomes an essential problem for understanding complex molecule functions and the role of each subtype during the dynamic biological process, such as cell fate specification (Sonawane, Platig et al. , Chasman and Roy 2017, Gal, London et al. 2017). The ability to obtain sample-wise expression variation in each subtype is critical prior to infer subtype-specific molecular networks (Shen-Orr, Tibshirani et al. 2010, Junttila and de Sauvage 2013). Single-cell expression profiling techniques have become popular to investigate cell-type-specific network but may lose critical information of cell-cell interactions and is prone to cell-cycle/state confounders (Buettner, Natarajan et al. 2015, Gal, London et al. 2017).

While the current CAM tool can dissect mixed signals of multiple samples into the 'averaged' expression profiles of subtypes, many subsequent molecular analyses of complex tissues require sample-specific signal deconvolution where each sample is a mixture of 'individualized' subtype-specific expression profiles. Here we propose a new algorithm called swCAM as an extension of CAM to solve sample-specific Blind Source Separation (sBSS) problem. The sBSS problem, because the number of variables is much larger than the number of observations, is ill-posed and underdetermined. As a result, simple yet highly regularized approaches often become the methods of choice (Hastie, Tibshirani et al. 2001). The sBSS problems have received increasing interest in hyperspectral imagery area where the spatially smooth and variation sparsity regularization can be exploited to unmix spectral signals (Thouvenin, Dobigeon et al. 2016). In the context of biological process, transcriptional regulatory networks connect regulatory proteins, such as transcription factors (TFs) and signaling proteins, to target genes and thus form co-expressed gene sets as function modules in each subtype. Based on such underlying cellular mechanisms, we impose and exploit the low-rank assumption on between-sample variations of molecule expressions in each

subtype. While estimating subtype-specific signals from a single mixture for each gene independently is an underdetermined problem, the low-rank assumption can aggregate information from gene sets within function modules to help find a biologically plausible solution.

General rank minimization is a challenging nonconvex optimization problem for which all existing finite-time algorithms have at least doubly exponential running times in both theory and practice (Recht, Fazel et al. 2010). Minimizing the nuclear norm, or the sum of the singular values of the matrix, over the affine subset, have multiple advantages. The nuclear norm is a convex function and can be optimized efficiently, thus can provide the best convex approximation of the rank function over the unit ball of matrices with norm less than one (Recht, Fazel et al. 2010, Candes, Sing-Long et al. 2013). Nuclear norm regularization has been successfully applied in many practical applications with low-rank modeling, such as image denoising (Candes, Sing-Long et al. 2013) and matrix completion (Cai, Candès et al. 2010). swCAM will adopt nuclear norm regularization to optimize the estimation of between-sample variations in each subtype to recover sample-specific signals for each subtype. This Chapter introduces mathematical modeling of sample-specific deconvolution and optimization solver used in swCAM algorithm, followed by validation in simulations and discussion on further possible improvement by sparsity regularization.

Method

Problem formulation and nuclear norm regularization

A fundamental assumption for the conventional linear mixing model, $\boldsymbol{X} = \boldsymbol{AS} + \boldsymbol{E}$, is that all the mixture samples share a common source matrix $\boldsymbol{S}$. However, each sample may have its 'individualized' sample-specific sources as the sampled realizations in additional sample-specific subtype proportions (**Fig. 1**):

$$\boldsymbol{S}_i = \overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i, i = 1, \dots, M.$$

The associated sample-specific BSS (sBSS) model is given by

$$\boldsymbol{x}_i = \boldsymbol{a}_i(\bar{\boldsymbol{S}} + \Delta\boldsymbol{S}_i) + \boldsymbol{n}_i \in \mathbb{R}^L, \forall i = 1, \dots, M \tag{1}$$

where $\boldsymbol{a}_i$ is the row vector in proportion matrix $\boldsymbol{A}$, and $\boldsymbol{n}_i$ is noise term. $\Delta\boldsymbol{S}_i$ is expected to have small-valued entries so that source matrices among different samples are similar.

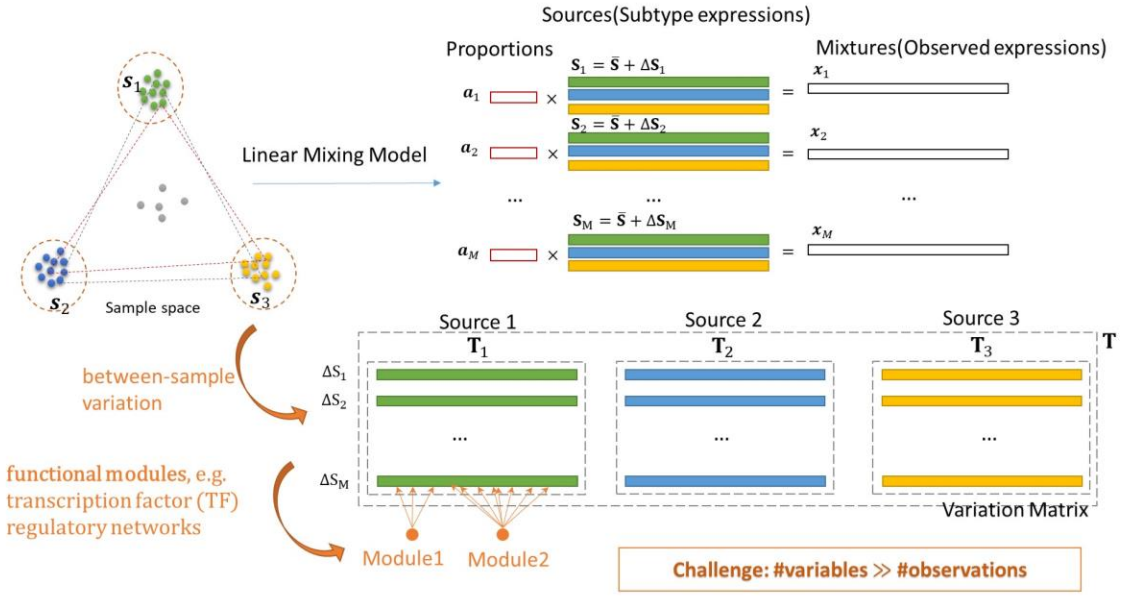

Fig. 1. Sample-specific deconvolution problem formulation and the assumption of hidden low-rank pattern in each source. (For convenient illustration, $\boldsymbol{T}$ matrix in all figures are the transposed version of those in the text and equations.)

If some expression units (molecular features) are highly correlated in a particular source, the rank of the matrix consisting of $\Delta\boldsymbol{S}_i$ entries in this source will be low, leading to the following swCAM objective function:

$$\min_{\boldsymbol{A},\{\Delta\boldsymbol{S}_i\}_{i=1}^M} \frac{1}{2}\sum_{i=1}^{M} \|\boldsymbol{x}_i - \boldsymbol{a}_i(\bar{\boldsymbol{S}} + \Delta\boldsymbol{S}_i)\|_2^2 + \lambda \sum_{k=1}^{K} \|\boldsymbol{T}_k\|_* \tag{2}$$

$$s.t.\ \boldsymbol{a}_i \geqslant \boldsymbol{0}_K, \boldsymbol{a}_i\boldsymbol{1}_K = 1, \bar{\boldsymbol{S}} + \Delta\boldsymbol{S}_i \geqslant \boldsymbol{0}_{K\times L},$$

$$\boldsymbol{T}_k = [\Delta\boldsymbol{S}_1^T(k), \dots, \Delta\boldsymbol{S}_M^T(k)] \in \mathbb{R}^{L\times M},\ k = 1, \dots, K,$$

where $\boldsymbol{T}_k$ consists of the kth column in all $\Delta\boldsymbol{S}_i, i = 1, \dots, M$, representing between-sample variation in source k, namely the variation matrix for the kth subtype. $\|\boldsymbol{T}_k\|_*$ is the nuclear norm of $\boldsymbol{T}_k$; and $\lambda > 0$ is the regularization parameter of nuclear norms. The hyperparameter λ actually

can be different among different source k and thus the regularizer in (2) can be replaced by $\sum_{k=1}^{K}\lambda_k\|\boldsymbol{T}_k\|_*$ if necessary. Please see supplementary information for the reason to select subtype-specific regularization terms.

Optimization of swCAM objective function

The objective function in (2) is bi-convex w.r.t. the two block-wise variables, i.e. $\boldsymbol{A} \triangleq [\boldsymbol{a}_1^T, \dots, \boldsymbol{a}_M^T]^T$ and $\boldsymbol{T} \triangleq [\boldsymbol{T}_1^T, \dots, \boldsymbol{T}_M^T]^T \in \mathbb{R}^{KL\times M}$. Accordingly, we can solve (2) by alternatively solving the following two convex subproblems until convergence:

$$\boldsymbol{T}^{p+1} \in \underset{\Delta\boldsymbol{S}_i \geqslant -\boldsymbol{S},\forall \boldsymbol{i}}{\operatorname{argmin}} \mathcal{J}(\boldsymbol{A}^p, \boldsymbol{T}) \tag{3}$$

$$\boldsymbol{A}^{p+1} \in \underset{\boldsymbol{A}\geqslant 0_{M\times K}, \boldsymbol{A}\boldsymbol{1}_K=\boldsymbol{1}_M}{\operatorname{argmin}} \mathcal{J}(\boldsymbol{A}, \boldsymbol{T}^{p+1}) \tag{4}$$

where

$$\mathcal{J}(\boldsymbol{A},\boldsymbol{T}) \triangleq \frac{1}{2}\sum_{i=1}^{M}\|\boldsymbol{x}_i - \boldsymbol{a}_i(\overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i)\|_2^2 + \lambda\sum_{k=1}^{K}\|\boldsymbol{T}_k\|_*$$

CAM-estimated subtype-specific expression matrix serves as the initial reference $\overline{\boldsymbol{S}}$. Note that in (3) (4), we have implicitly used the following relationship for concise representation:

$$\boldsymbol{T} \triangleq [vec(\Delta\boldsymbol{S}_1^T), \dots, vec(\Delta\boldsymbol{S}_M^T)],$$

where (4) can be decoupled w.r.t each row of $\boldsymbol{A}$:

$$\boldsymbol{a}_i^{p+1} \in \underset{\boldsymbol{a}_i\geqslant \boldsymbol{0}_K, \boldsymbol{a}_i\boldsymbol{1}_K=1}{\operatorname{argmin}} \frac{1}{2}\left\|\boldsymbol{x}_i - \boldsymbol{a}_i\left(\overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i^{p+1}\right)\right\|_2^2$$

which can be solved using quadratic programming. If *a prior* proportion matrix or CAM-estimated proportion matrix has already been of high quality, we can skip the alternative optimization on $\boldsymbol{A}$ matrix, and obtain $\boldsymbol{T}$ matrix by optimizing the subproblem (3) only once.

To solve (3), we notice that the main bottleneck is its huge dimension of variables (typically, L is several ten thousand), preventing conventional convex solvers from being readily applicable here.

We propose to solve (3) by adapting the alternating direction method of multipliers (ADMM), which has been widely applied to many large-scale problems in areas such as statistical learning, image processing and computational biology (Boyd, Parikh et al. 2011).

ADMM naturally allows decoupling the non-smooth regularization term from the smooth loss term, which is computationally advantageous. Specifically, we reformulate (3) in the form that the primal variable can be "split" into several parts, with the associated objective function "separable" across this splitting (Boyd, Parikh et al. 2011). We will use the following definitions:

$$\boldsymbol{T} \triangleq [vec(\Delta \boldsymbol{S}_1^T), \dots, vec(\Delta \boldsymbol{S}_M^T)] = \begin{bmatrix} \boldsymbol{T}_1 \\ \dots \\ \boldsymbol{T}_K \end{bmatrix} \in \mathbb{R}^{KL \times M}$$

$$\boldsymbol{S} \triangleq [vec(\boldsymbol{S}_1^T), \dots, vec(\boldsymbol{S}_M^T)] \in \mathbb{R}^{KL \times M}$$

$$\boldsymbol{V} \triangleq \boldsymbol{X}^T \in \mathbb{R}^{L \times M}$$

$$\boldsymbol{W} \triangleq \begin{bmatrix} \boldsymbol{T} \\ \boldsymbol{S} \end{bmatrix} \in \mathbb{R}^{2KL \times M}$$

$$\boldsymbol{C}_1 \triangleq \begin{bmatrix} \boldsymbol{I}_{KL} \\ \boldsymbol{I}_{KL} \end{bmatrix} \in \mathbb{R}^{2KL \times KL}$$

$$\boldsymbol{C}_2 \triangleq -\boldsymbol{I}_{2KL} \in \mathbb{R}^{2KL \times 2KL}$$

$$\boldsymbol{C}_3 \triangleq \begin{bmatrix} \boldsymbol{1}_M^T \otimes vec(\overline{\boldsymbol{S}^T}) \\ \boldsymbol{0}_{KL \times M} \end{bmatrix} \in \mathbb{R}^{2KL \times M}$$

$$\boldsymbol{B}_0 \triangleq [\boldsymbol{0}_{KL \times KL}, \boldsymbol{I}_{KL}] \in \mathbb{R}^{KL \times 2KL}$$

$$\boldsymbol{B}_k \triangleq \left[\boldsymbol{0}_{L \times (k-1)L}, \boldsymbol{I}_L, \boldsymbol{0}_{L \times (K-k)L}, \boldsymbol{0}_{L \times KL}\right] \in \mathbb{R}^{L \times 2KL}, k = 0, \dots, K$$

Then we can simplify (3) as the equivalent form:

$$\min_{\boldsymbol{U} \in \mathbb{R}^{KL \times M}, \boldsymbol{W} \in \mathbb{R}^{2KL \times M}} \frac{1}{2} \|\mathcal{A}(\boldsymbol{U}) - \boldsymbol{V}\|_F^2 + \lambda \sum_{k=1}^{K} \|\boldsymbol{B}_k \boldsymbol{W}\|_* + I_+(\boldsymbol{B}_0 \boldsymbol{W}) \tag{5}$$

$$s.t. \, \boldsymbol{C}_1 \boldsymbol{U} + \boldsymbol{C}_2 \boldsymbol{W} = \boldsymbol{C}_3,$$

where $I_+(\cdot)$ is the indicator function for the non-negative orthant; $I_+(\boldsymbol{B}_0\boldsymbol{W}) = I_+(\boldsymbol{S}) = 0$ if $\boldsymbol{S} \geqslant \boldsymbol{0}_{KL\times M}$ ($I_+(\boldsymbol{U}) = +\infty$, otherwise). The linear transformation in the first term is $\mathcal{A}(\boldsymbol{U}) = \mathcal{A}([\boldsymbol{u}_1, \dots, \boldsymbol{u}_M]) = [\boldsymbol{H}_1\boldsymbol{u}_1, \dots, \boldsymbol{H}_M\boldsymbol{u}_M]$ with $\boldsymbol{H}_i = \left[\boldsymbol{a}_i^p \otimes I_L\right], i = 1, \dots, M$. Note that (5) has been with the ADMM form w.r.t. the two split block variables $\boldsymbol{U}$ and $\boldsymbol{W}$, and, as (5) is solved, the solution of (3) can be obtained by $\boldsymbol{T}^{p+1} = [\,\boldsymbol{I}_{KL}, \boldsymbol{0}_{KL\times KL}]\boldsymbol{W}^*$.

Given a penalty parameter $\gamma > 0$ (empirically, $\gamma \coloneqq 1$ generally guarantees good convergence speed), the augmented Lagrangian (ignoring some irrelevant terms) of problem (5) is defined by

$$\mathcal{L}(\boldsymbol{U}, \boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{2}\|\mathcal{A}(\boldsymbol{U}) - \boldsymbol{V}\|_F^2 + \lambda \sum_{k=1}^{K} \|\boldsymbol{B}_k\boldsymbol{W}\|_* + I_+(\boldsymbol{B}_0\boldsymbol{W}) + \frac{\gamma}{2}\|\boldsymbol{C}_1\boldsymbol{U} + \boldsymbol{C}_2\boldsymbol{W} - \boldsymbol{C}_3 - \boldsymbol{Z}\|_F^2$$

where "$-\gamma\boldsymbol{Z}$"$\in \mathbb{R}^{2KL\times M}$ is the dual variable (or Lagrange multiplier) associated with the constraint $\boldsymbol{C}_1\boldsymbol{U} + \boldsymbol{C}_2\boldsymbol{W} = \boldsymbol{C}_3$. Then, ADMM solves (5) via the following iterative procedure:

$$\boldsymbol{U}^{q+1} \epsilon \underset{\boldsymbol{U}\in\mathbb{R}^{KL\times M}}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{U}, \boldsymbol{W}^q, \boldsymbol{Z}^q) \tag{6a}$$

$$\boldsymbol{W}^{q+1} \epsilon \underset{\boldsymbol{W}\in\mathbb{R}^{2KL\times M}}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{U}^{q+1}, \boldsymbol{W}, \boldsymbol{Z}^q) \tag{6b}$$

$$\boldsymbol{Z}^{q+1} = \boldsymbol{Z}^q - (\boldsymbol{C}_1\boldsymbol{U}^{q+1} + \boldsymbol{C}_2\boldsymbol{W}^{q+1} - \boldsymbol{C}_3) \tag{6c}$$

where $\boldsymbol{W}^0$ can be initialized by $[\boldsymbol{T}_0^T, \boldsymbol{U}_0^T]^T$ with $\boldsymbol{T}_0 = \boldsymbol{0}_{KL\times M}$ and $\boldsymbol{U}_0 = \boldsymbol{1}_M^T \otimes vec(\overline{\boldsymbol{S}}^T)$; $\boldsymbol{Z}^0$ can be simply initialized by $\boldsymbol{0}_{2KL\times M}$. As we will show, both (6a) and (6b) can be solved with closed-form expressions, thanks to the decomposability of ADMM.

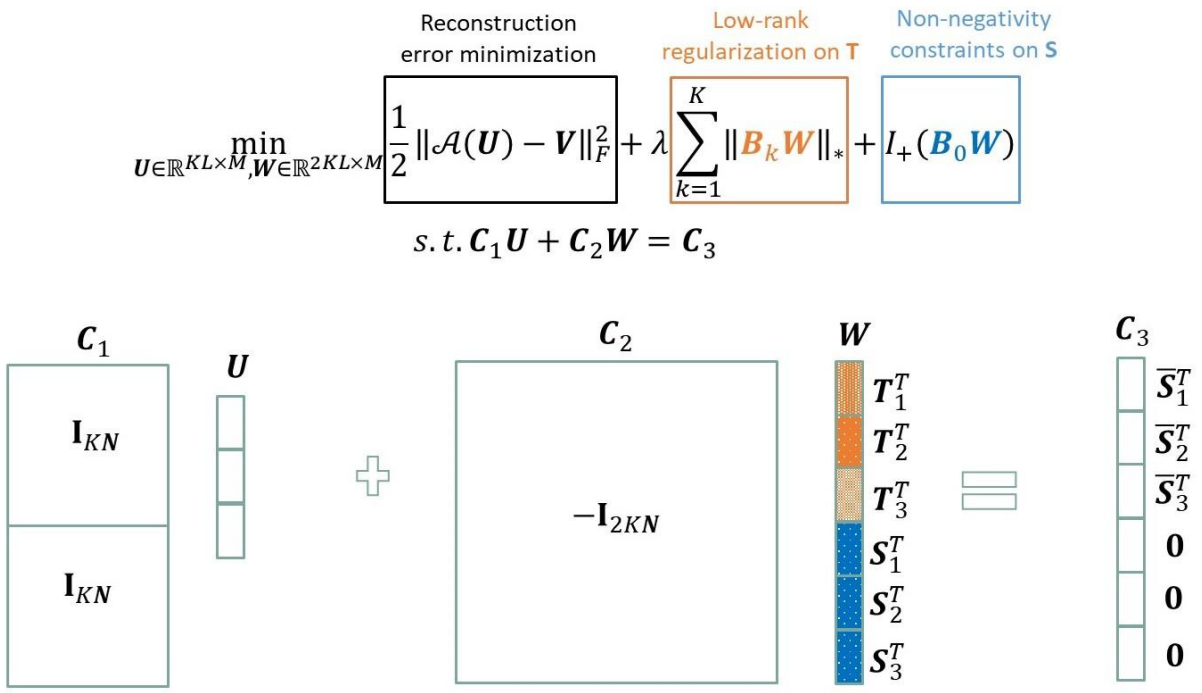

Fig. 2. The objective function of swCAM for sample-specific deconvolution problem and its reformulation by ADMM. (For convenient illustration, $\boldsymbol{T}$ matrix in all figures are the transposed version of those in the text and equations.)

Notice that (6a) is a column-wise separable optimization problem, so we can decouple w.r.t each column of $\boldsymbol{U}$:

$$\boldsymbol{u}_i^{q+1} \in \underset{\boldsymbol{u}_i\in\mathbb{R}^{KL}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{H}_i\boldsymbol{u}_i-\boldsymbol{v}_i\|_2^2+\frac{\gamma}{2}\left\|\boldsymbol{C}_1\boldsymbol{u}_i+\boldsymbol{y}_i^q\right\|_F^2 \tag{7}$$

where $\left[\boldsymbol{y}_1^q,\ldots,\boldsymbol{y}_M^q\right]\triangleq\boldsymbol{C}_2\boldsymbol{W}^q-\boldsymbol{C}_3-\boldsymbol{Z}^q$. The subproblem (7) is an unconstrained quadratic problem, which can be solved by

$$\boldsymbol{u}_i^{q+1}=(\boldsymbol{H}_i^T\boldsymbol{H}_i+\gamma\boldsymbol{C}_1^T\boldsymbol{C}_1)^{-1}\left(\boldsymbol{H}_i^T\boldsymbol{v}_i-\gamma\boldsymbol{C}_1^T\boldsymbol{y}_i^q\right). \tag{8}$$

The matrix inversion can speed up by

$$(\boldsymbol{H}_i^T\boldsymbol{H}_i+\gamma\boldsymbol{C}_1^T\boldsymbol{C}_1)^{-1}=\left(\left(\boldsymbol{a}_i^p\right)^T\boldsymbol{a}_i^p+2\gamma\boldsymbol{I}_K\right)^{-1}\otimes\boldsymbol{I}_L.$$

The right term in (8) can also be simplified as

$$\boldsymbol{H}_i^T\boldsymbol{v}_i - \gamma\boldsymbol{C}_1^T\boldsymbol{y}_i^q = \left(\boldsymbol{a}_i^p\right)^T \otimes \boldsymbol{x}_i^T - \gamma\left(\overline{\boldsymbol{y}_i^q} + \underline{\boldsymbol{y}_i^q}\right),$$

where $\boldsymbol{y}_i^q = \left[\left(\overline{\boldsymbol{y}_i^q}\right)^T, \left(\underline{\boldsymbol{y}_i^q}\right)^T\right]^T$ with $\overline{\boldsymbol{y}_i^q} \in \mathbb{R}^{KL}$ and $\underline{\boldsymbol{y}_i^q} \in \mathbb{R}^{KL}$ being the first and second half vector of $\boldsymbol{y}_i^q$, respectively.

Finally, the column vectors of $\boldsymbol{U}^{q+1}$ in (6a) can be computed fast by

$$\boldsymbol{u}_i^{q+1} = vec\left\{devec\left\{\left(\boldsymbol{a}_i^p\right)^T \otimes \boldsymbol{x}_i^T - \gamma\left(\overline{\boldsymbol{y}_i^q} + \underline{\boldsymbol{y}_i^q}\right)|L, K\right\}\left(\left(\boldsymbol{a}_i^p\right)^T\boldsymbol{a}_i^p + 2\gamma\boldsymbol{I}_K\right)^{-1}\right\} \tag{9}$$

To solve (4.6b), we remove some irrelevant terms from its objective function:

$$\min_{\boldsymbol{W}\in\mathbb{R}^{2KL\times M}} \lambda\sum_{k=1}^{K}\|\boldsymbol{B}_k\boldsymbol{W}\|_* + I_+(\boldsymbol{B}_0\boldsymbol{W}) + \frac{\gamma}{2}\|\boldsymbol{C}_1\boldsymbol{U}^{q+1} + \boldsymbol{C}_2\boldsymbol{W} - \boldsymbol{C}_3 - \boldsymbol{Z}^q\|_F^2, \tag{10}$$

And then, by defining $\boldsymbol{U}_k^{q+1} \in \mathbb{R}^{L\times M}, k = 1, \dots, K$ as block matrices from top to bottom in $\boldsymbol{U}^{q+1} \in \mathbb{R}^{KL\times M}$, $\boldsymbol{Z}_k \in \mathbb{R}^{L\times M}, k = 1, \dots, K$ and $\boldsymbol{Z}_0 \in \mathbb{R}^{KL\times M}$ as block matrices from top to bottom in $\boldsymbol{Z} \in \mathbb{R}^{2KL\times M}$, respectively (i.e., $\boldsymbol{Z} \triangleq [\boldsymbol{Z}_1^T, \dots, \boldsymbol{Z}_K^T, \boldsymbol{Z}_0^T]^T$), we decouple the objective function (10) as functions of $\boldsymbol{T}_k, k = 1, \dots, K$ and $\boldsymbol{S}$:

$$\min_{\boldsymbol{W}\in\mathbb{R}^{2KL\times M}} \sum_{k=1}^{K}\left\{\lambda\|\boldsymbol{T}_k\|_* + \frac{\gamma}{2}\left\|\boldsymbol{U}_k^{q+1} - \boldsymbol{T}_k - \boldsymbol{1}_M^T\otimes\bar{\boldsymbol{s}}_k - \boldsymbol{Z}_k^q\right\|_F^2\right\} + \left\{I_+(\boldsymbol{S}) + \frac{\gamma}{2}\left\|\boldsymbol{U}^{q+1} - \boldsymbol{S} - \boldsymbol{Z}_0^q\right\|_F^2\right\}$$

Therefore, $\boldsymbol{W}^{q+1}$ can be solved by the proximal point algorithm (PPA) (Parikh and Boyd 2014). Specifically, we have

$$\boldsymbol{W}^{q+1} = \left[\left(\boldsymbol{T}_1^{q+1}\right)^T, \dots, \left(\boldsymbol{T}_K^{q+1}\right)^T, (\boldsymbol{S}^{q+1})^T\right]^T$$

in which

$$\boldsymbol{T}_k^{q+1} \in \underset{\boldsymbol{T}\in\mathbb{R}^{KL\times M}}{\operatorname{argmin}} \lambda\|\boldsymbol{T}_k\|_* + \frac{\gamma}{2}\left\|\boldsymbol{U}_k^{q+1} - \boldsymbol{T}_k - \boldsymbol{1}_M^T\otimes\bar{\boldsymbol{s}}_k - \boldsymbol{Z}_k^q\right\|_F^2 \tag{11a}$$

$$\boldsymbol{S}^{q+1} \in \underset{\boldsymbol{T}\in\mathbb{R}^{KL\times M}}{\operatorname{argmin}} I_+(\boldsymbol{S}) + \frac{\gamma}{2}\left\|\boldsymbol{U}^{q+1} - \boldsymbol{S} - \boldsymbol{Z}_0^q\right\|_F^2 \tag{11b}$$

Note that (4.11a) and (4.11b) are exactly the proximal operators of $\|\boldsymbol{T}_k\|_*$ and $I_+(\boldsymbol{S})$, respectively (Parikh and Boyd 2014), and their closed-form solutions are given by

$$\boldsymbol{T}_k^{q+1} = \sum_{\ell=1}^{r} \left(\sigma_{k\ell} - \frac{\lambda}{\gamma}\right)_+ \boldsymbol{\mu}_{k\ell}\boldsymbol{\nu}_{k\ell}^T, k = 1, \dots, K, \tag{12}$$

$$\boldsymbol{S}^{q+1} = \left[\boldsymbol{U}^{q+1} - \boldsymbol{Z}_0^q\right]_+, \tag{13}$$

where the singular value decomposition (SVD) of is performed ahead of the computation of (12), i.e. $\boldsymbol{U}_k^{q+1} - \boldsymbol{T}_k - \mathbf{1}_M^T \otimes \bar{\boldsymbol{s}}_k - \boldsymbol{Z}_k^q = \sum_{\ell=1}^{r} \sigma_{k\ell}\boldsymbol{\mu}_{k\ell}\boldsymbol{\nu}_{k\ell}^T$.

A reasonable termination criterion is that the primal residual, $\varepsilon^{pri} = \|\boldsymbol{C}_1\boldsymbol{U} + \boldsymbol{C}_2\boldsymbol{W} - \boldsymbol{C}_3\|_2$, and dual residual, $\varepsilon^{dual} = \|\gamma\boldsymbol{C}_1^T\boldsymbol{C}_2(\boldsymbol{W}^{q+1} - \boldsymbol{W}^q)\|_2$, are smaller than a predefined tolerance.

Model parameter tuning

In noisy scenarios, the penalty parameter λ setting is critical to determine how much variation is persevered as patterns of interest or ignored as noise. An extremely large λ will coerce the individual variation to be zero. Decreasing λ will allow more subtype-specific patterns to be detected until overfitting.

Cross-validation is a popular strategy in parameter tuning for the balance of underfitting and overfitting. One round of cross-validation excludes a certain portion of samples and uses the model learned from other samples to predict the excluded ones. Then every model is assessed by summarizing prediction performances across multiple rounds. However, our sample-specific deconvolution estimates the individual expression of each sample in each subtype, which cannot be used to predict the excluded samples directly. Thus, we proposed to randomly exclude entries rather than samples in $\boldsymbol{X}$ matrix (**Fig. 3**), similar to the strategy used in missing value imputation. The foundation of success is that the low-rank patterns in $\boldsymbol{T}_k$ matrix are detectable by only a portion of $\boldsymbol{X}$ entries and able to predict the excluded $\boldsymbol{X}$ entries.

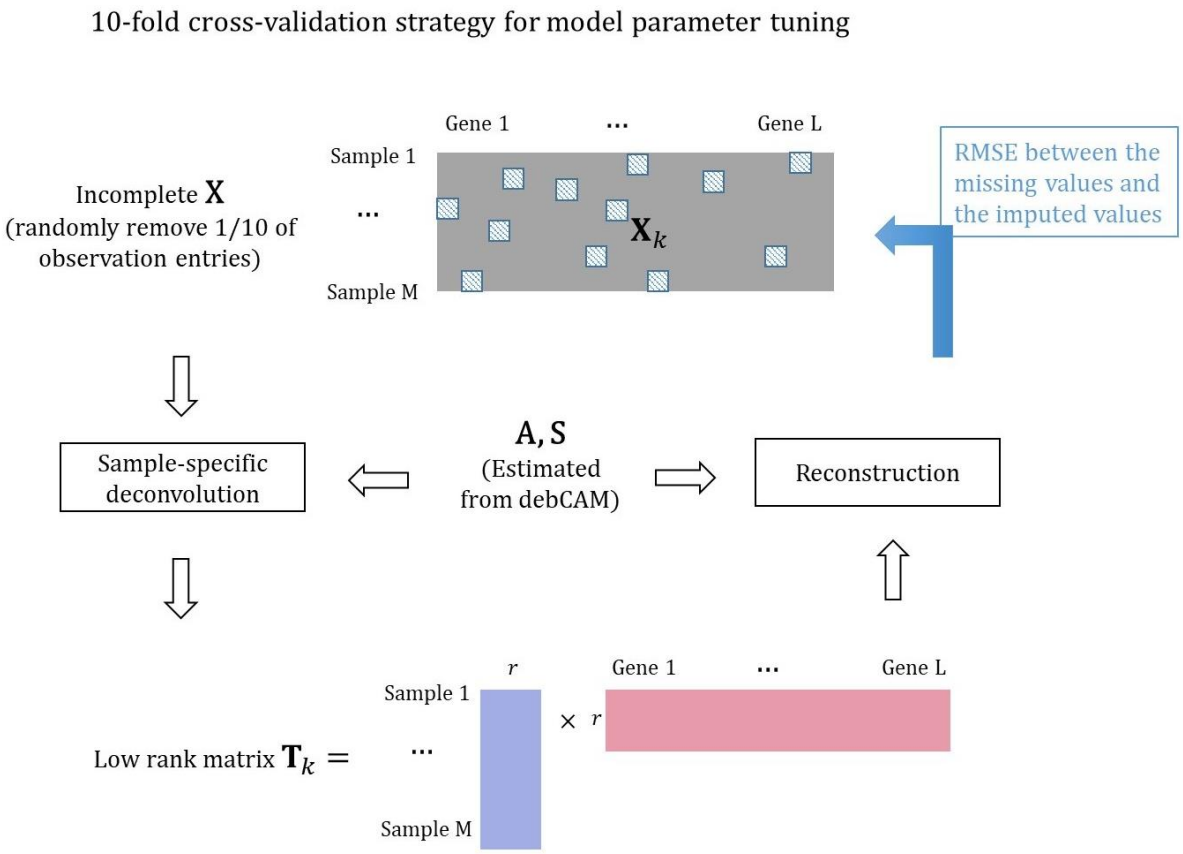

Fig. 3. 10-fold cross-validation strategy for model parameter tuning. A part of entries is randomly removed before applying swCAM. The removed entries are reconstructed by estimated $\boldsymbol{T}$ matrix and compared to observed expressions for computing RMSE to decide the optimal parameter λ.

Specifically, we fix the $\boldsymbol{A}$ and $\overline{\boldsymbol{S}}$ at the initialization values (from CAM-estimation or *a priori* knowledge) and randomly remove entries in $\boldsymbol{X}$ matrix, leading to the objective function w.r.t $\Delta\boldsymbol{S}_i, i = 1, \dots, M$:

$$\min_{\{\Delta\boldsymbol{S}_i\}_{i=1}^{M}} \frac{1}{2}\sum_{i=1}^{M}\left\|P_{\Omega_i}(\boldsymbol{x}_i) - P_{\Omega_i}\left(\boldsymbol{a}_i(\overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i)\right)\right\|_2^2 + \lambda\sum_{k=1}^{K}\|\boldsymbol{T}_k\|_* \tag{14}$$

$$s.t. \overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i \succcurlyeq \boldsymbol{0}_{K\times L},$$

$$\boldsymbol{T}_k = [\Delta\boldsymbol{S}_1^T(k), \dots, \Delta\boldsymbol{S}_M^T(k)] \in \mathbb{R}^{L\times M}, k = 1, \dots, K,$$

where $P_{\Omega_i}(\boldsymbol{x}_i) \in \mathbb{R}^L$ denote a vector with the entries in Ω_i left alone, and all other entries set to zero. The workflow of our proposed 10-fold cross-validation strategy is:

(1) Randomly split all entries into 10 folds;

(2) Remove one fold of entries and use the remaining 9 folds of entries to solve (14) with different λ values $[\lambda_1, \lambda_2, \ldots]$;

(3) Use estimated $\Delta \boldsymbol{S}_i(\lambda_\theta), i = 1, \ldots, M,\ \theta = 1,2, \ldots$, together with fixed $\boldsymbol{A}$ and $\overline{\boldsymbol{S}}$ matrix to reconstruct $\boldsymbol{X}$ matrix and only record the reconstructed values for the removed entries in $\boldsymbol{X}$;

(4) Repeat Step (2)-(3) and obtained a reconstructed $\widetilde{\boldsymbol{X}}(\lambda_\theta)$ matrix in which all entry values are reconstructed when their original values are absent in optimization processes with $\lambda = \lambda_\theta$.

(5) Calculate Root Mean Square Error (RMSE) by

$$RMSE(\lambda_\theta) = \sqrt{\frac{1}{ML}\sum_{i=1}^{M}\sum_{j=1}^{L}\left(\boldsymbol{X}_{ij} - \widetilde{\boldsymbol{X}}_{ij}(\lambda_\theta)\right)^2} \tag{15}$$

(6) Choose the λ_θ yielding the minimum RMSE.

Warm start can be used in Step (2) with the decreasing parameter $\lambda_1 > \lambda_2 > \cdots$, which use the estimation with λ_θ as the initialization of next optimization with $\lambda_{\theta+1}$.

The optimization problem (14) can be solved using a similar ADMM algorithm in (5-13) that have solved (3). The only modification is that (7) becomes

$$\boldsymbol{u}_i^{q+1} \in \underset{\boldsymbol{u}_i \in \mathbb{R}^{KL}}{\operatorname{argmin}} \frac{1}{2}\left\|P'_{\Omega_i}(\boldsymbol{H}_i\boldsymbol{u}_i) - P'_{\Omega_i}(\boldsymbol{v}_i)\right\|_2^2 + \frac{\gamma}{2}\left\|\boldsymbol{C}_1\boldsymbol{u}_i + \boldsymbol{y}_i^q\right\|_F^2 \tag{16}$$

where $P'_{\Omega_i}(\cdot) = \left[\boldsymbol{1}_K^T \otimes P_{\Omega_i}(\cdot)^T\right]^T \in \mathbb{R}^{KL}$ makes all excluded-entry related variables be optimized only by the second term, which is still an unconstrained quadratic problem that can be solved easily. The remaining variables unrelated to excluded entries can still be optimized following (8-9).

Sparsity regularization

In addition to low-rank assumption, we could also reasonably assume only limited genes are involved in functional modules and thus impose a row-sparsity regularization by $\ell_{2,1}$-norm minimization. The alternative swCAM formulation will be:

$$\min_{\boldsymbol{A},\{\Delta\boldsymbol{S}_i\}_{i=1}^M} \frac{1}{2}\sum_{i=1}^{M}\|\boldsymbol{x}_i - \boldsymbol{a}_i(\overline{\boldsymbol{S}} + \Delta\boldsymbol{S}_i)\|_2^2 + \lambda\sum_{k=1}^{K}\|\boldsymbol{T}_k\|_* + \delta\|\boldsymbol{T}\|_{2,1} \tag{17}$$

where $\delta > 0$ is the regularization parameter of $\ell_{2,1}$ norm of $\boldsymbol{T}$, defined as

$$\|\boldsymbol{T}\|_{2,1} \triangleq \sum_{i=1}^{KL} \|\boldsymbol{t}_i\|_2$$

accounting for the row-sparsity of $\boldsymbol{T}$. If necessary, the parameter δ actually can be varied for different rows based on the character of each gene, such as mean-variance trend. The supplementary information gives more details on the optimization of (17) by ADMM method. The ℓ_1 or ℓ_2-norm minimization, as common-used sparsity regularization methods, could impose the entry sparsity in $\boldsymbol{T}$ matrix. We also provide ADMM optimization for sample-specific deconvolution with ℓ_1 or ℓ_2-norm minimization, which could be useful in other sBSS problems.

Results

As swCAM focuses on subtype-specific variation estimation, simulating biological variance within each subtype and technical variance for each observation is important for validating swCAM performance. We conduct two sets of simulations. The first is in an ideal scenario where the variance is not related to mean value. The second is more realistic where genes with larger mean usually have larger variance.

Validation on ideal simulations

In the first simulations, we design twelve function modules, with four in each of three subtypes. The observations for 300 genes in 50 samples were simulated with subtype-specific expression baseline, $\bar{\mathbf{S}}$, sampled from the purified cell populations in real benchmark microarray gene expression data GSE19380 (Kuhn, Thu et al. 2011). $\boldsymbol{a_i}, i = 1, \dots, M$, are drawn randomly from a flat Dirichlet distribution. Between-sample variation, $\Delta \boldsymbol{S}_i(k,j), i = 1, \dots, M$, for the *k*th subtype and *j*th gene was drawn from normal distribution $\mathcal{N}(0, \sigma_{kj}^{(s)})$ if the *j*th gene was involved in a function module in the *k*th subtype; otherwise zero (Fig. 4a). The genes in the same function module has pairwise correlation coefficient equal to one, thus generating a highly correlated gene set in each module. $\sigma_{kj}^{(s)}$ are drawn from uniform distribution $U[50, 300]$. The technical noise, $\boldsymbol{n}_i, i = 1, \dots, M$, was drawn from zero-mean normal distribution with the variance $\sigma_{ij}^{(n)}$=10.

The twelve functional modules can be recognized in the variation matrix from swCAM when λ falls into a certain range (Fig. 4b~4i). Increasing the penalty parameter of the nuclear norm will filter more noise but at the cost of the possibility of missing the true variation signal. RMSE derived by 10-fold cross-validation strategy is relatively small when $\lambda = 1{\sim}50$ and reach the minimum at $\lambda = 5$ (Fig. 5a). The estimated variation matrix looks quite similar when $1 \leq \lambda \leq 50$ (Fig. 4e~4g), with 12 clear patterns and some artifacts. The artifacts are formed when the signal variation in one subtype spreads to other subtypes for the same genes, which are much lower than detected true signals if λ is not extremely small. (As shown in the supplementary information, the nuclear norm minimization for each subtype's variation matrix is a good option to reduce artifacts compared to other regularization terms.)

It is interesting to find $\lambda = 5$ is also the point where both primal and dual residuals surge in ADMM algorithm (Fig. 5c~5f). It is because larger λ tends to train an over-simplified model and thus approach the optimum solution more easily in ADMM.

The recovery of sample-specific signals in a subtype is also affected by the mixing proportions of this subtype within the sample. When a subtype accounts for a very small portion in a certain sample, its true signal in this sample will be very weak and thus underestimated (green points in Fig. 6). On the contrary, the major subtype in a sample can be estimated very well by CAM-SS (red points in Fig. 6).

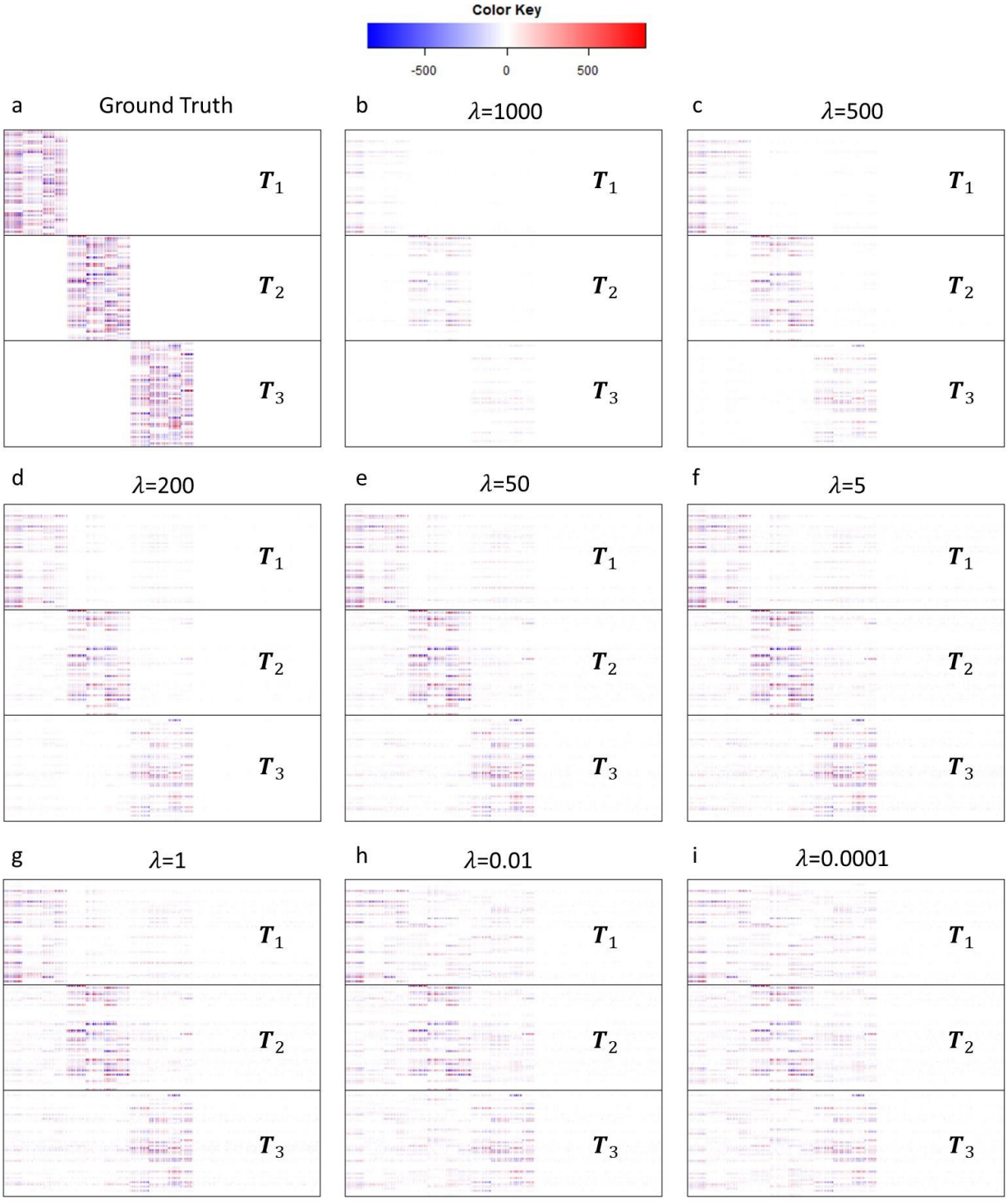

Fig. 4. Heatmap of estimated *T* matrix with varied λ parameters compared to ground truth in the ideal simulation. Increasing the penalty parameter of the nuclear norm will filter more noise but at the cost of the possibility of missing true signal variation.

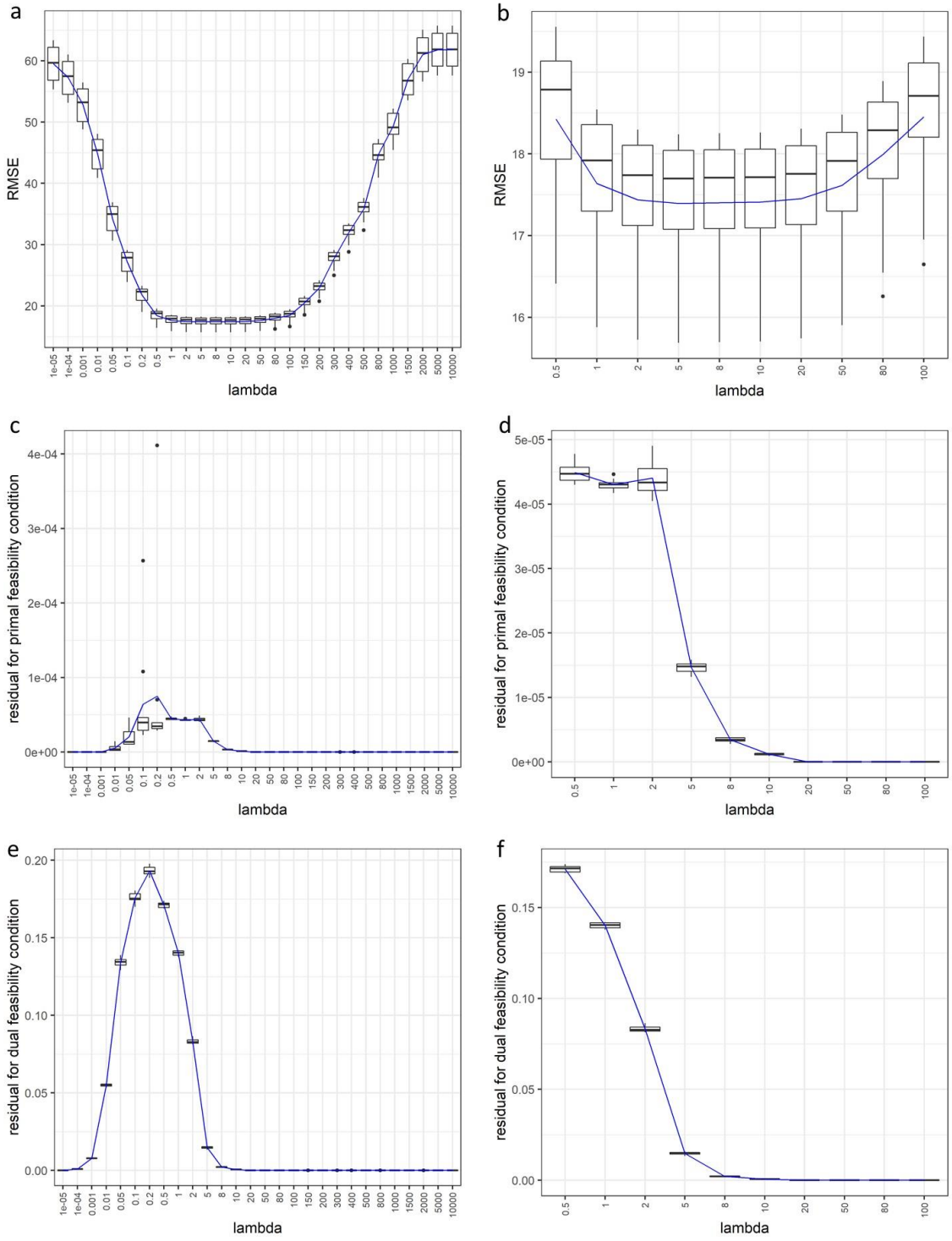

Fig. 5. 10-fold cross-validation results under different λ parameter in the ideal simulation. (a) RMSE; (c) Residuals for primal feasibility condition; (e) Residuals for dual feasibility condition; (b), (d), (f) are zoomed curves of (a), (c), (e).

Fig. 6. Estimated ***T*** matrix versus ground truth when λ=5 in the ideal simulation. The mixing proportions associated with estimated entries are colored to show the sample-specific expression estimations for high-proportion subtypes can be estimated more accurately than those for low-proportion ones

Validation on realistic simulations

Mean-variance trend is widely existing in molecular expression data. In our second simulation, all settings are the same as above except that the variance of subtype-specific expression, $\sigma_{kj}^{(s)}$, and the technical variance of observations, $\sigma_{ij}^{(n)}$, are proportional to the subtype-specific expression mean and mixed expression level, respectively. The coefficient of variation (CV), as the ratio of the standard deviation to the mean, is drawn from uniform distribution $U[0.15, 0.3]$ and $U[0.02, 0.05]$, respectively.

10-fold cross-validation strategy still obtains the minimum RMSE at $\lambda = 5$ (Fig. 8a~8b) when both primal and dual residuals also surge (Fig. 8c~8f). However, the estimated variation matrix by

swCAM is blurred by artifacts trained from noise (Fig. 7). Some high-expressed genes have relatively large variance, which could be falsely modeled as subtype-specific signal variations. As shown in Fig. 9, the entries with zero value in Ground Truth variation matrix could be overestimated.

Though the absolute expression values estimated by swCAM could deviate from Ground Truth, we can still clearly detect 12 functional modules defined by the Weighted Gene Correlation Network Analysis (WGCNA) (Zhang and Horvath 2005, Langfelder and Horvath 2008) on the estimated sample-specific expressions (Fig. 10). WGCNA constructs weighted networks based on correlation patterns among genes across samples and thus detects function modules of highly-correlated gene sets. In Fig. 10, the second and third subtype finds the exact four true modules with very few genes are missed. The first subtype detects an extra false module, but it is a less significant pattern compared to other modules and can be undetectable with stricter tree height cut threshold. More importantly, without swCAM based deconvolution (Fig. 10d), WGCNA on mixture expression profiles can find none of the true modules, but three false modules that are related to the mixing process of three subtypes.

Incorporation of L21-norm regularization

In the above simulations, the deconvoluted sample-specific signals contain artifacts trained from signals of other subtypes and artifacts trained from noise (Fig. 4 and Fig. 7). We can use a $\ell_{2,1}$-norm regularization to enforce the sparsity of genes that have signal variation across samples. It is supposed to reduce artifacts while it also follows the assumption that genes contributing to source variation in hidden modules are limited. Figure 11 shows the alleviated artifacts with $\lambda = 5$ and $\delta = 10$, 1, or 0.1. The true function modules are correctly detected with $\lambda = 5$ and $\delta = 1$ or 0.1, where the false module in the first subtype is suppressed when $\delta = 1$ (Fig. 12).

Increasing the penalty parameter δ will force more genes to have zero variance, which suppresses the artifacts and false function modules but brings the risk of missing the true signals. It is critical to propose a parameter tuning method for δ. However, the cross-validation strategy with randomly excluding entries for tuning parameter λ is based on the low-rank assumption, where the hidden low-rank patterns can be trained from a part of entries and then used to reconstruct the remaining entries. This strategy is not applicable to δ selection, which needs further study.

Fig. 7. Heatmap of estimated $\boldsymbol{T}$ matrix scaled by associated means compared to ground truth in the realistic simulation with varied λ parameters. Increasing the penalty parameter of the nuclear norm will filter more noise but at the cost of the possibility of missing true variation signal.

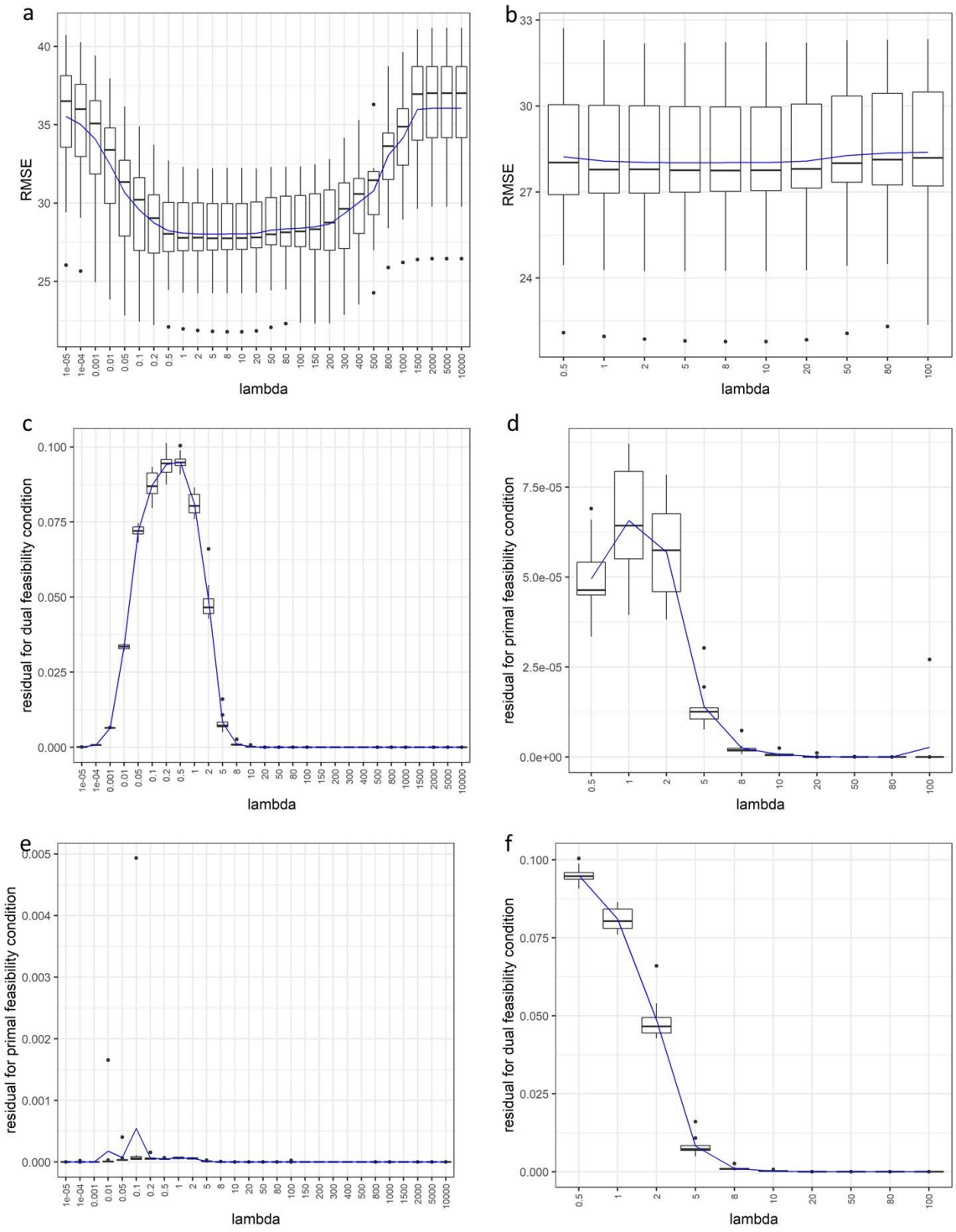

Fig. 8. 10-fold cross-validation results under different λ parameter in the realistic simulation. (a) RMSE; (c) Residuals for primal feasibility condition; (e) Residuals for dual feasibility condition; (b), (d), (f) are zoomed curves of (a), (c), (e).

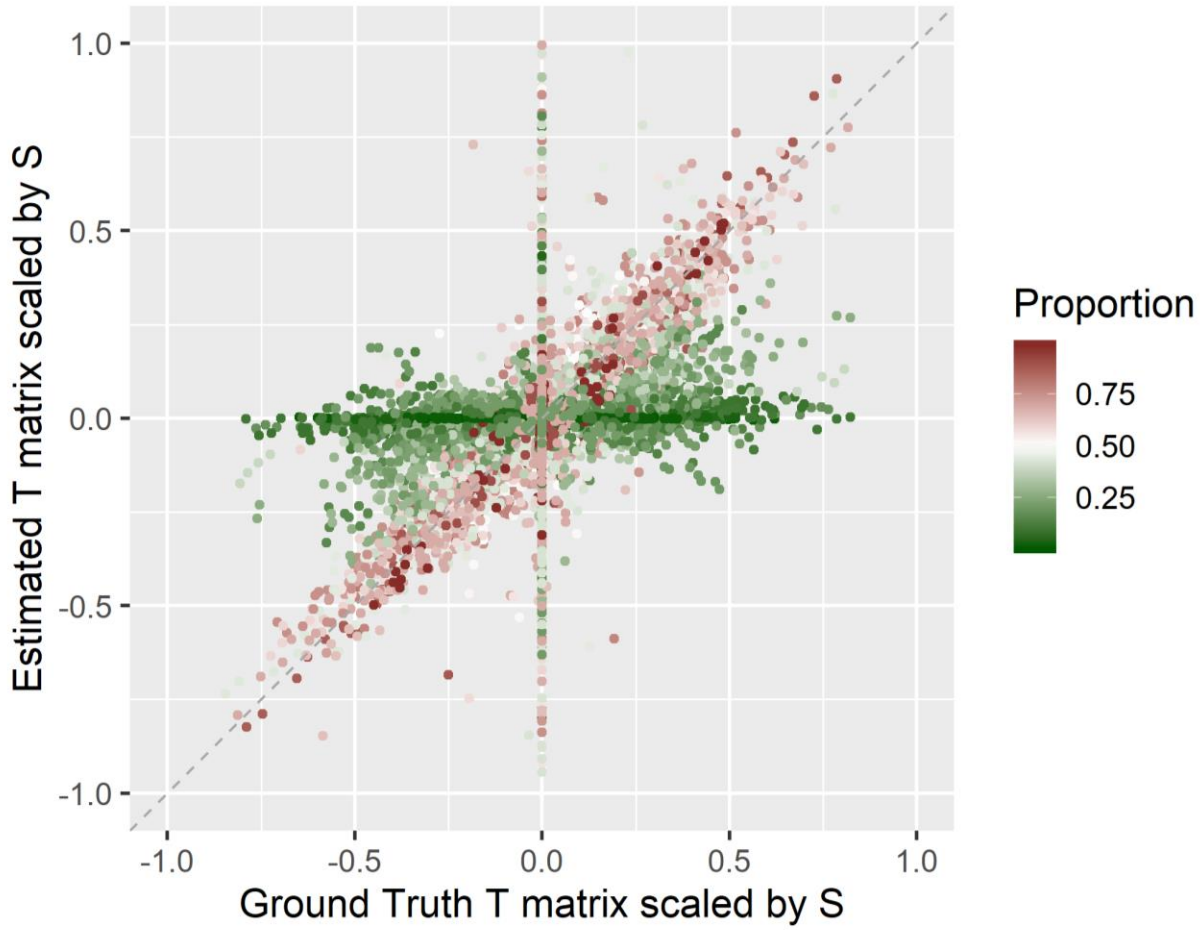

Fig. 9. Estimated ***T*** matrix scaled by associated means versus ground truth in the realistic simulation (λ=5). The mixing proportions associated with estimated entries are colored to show the sample-specific expression estimations for high-proportion subtypes can be estimated more accurately than those for low-proportion ones.

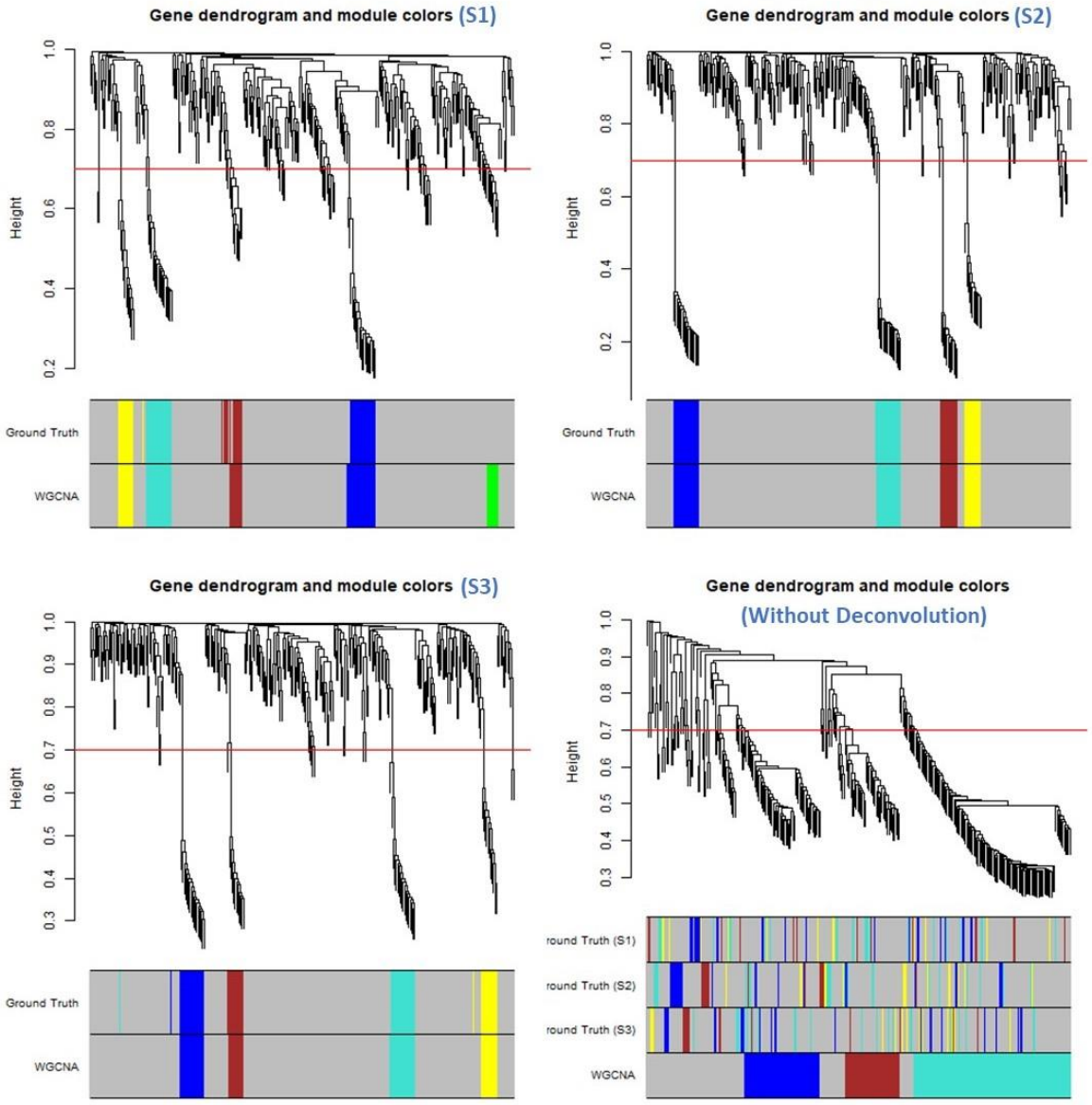

Fig. 10. Gene co-expressed function modules detected by WGCNA on swCAM estimated sample-specific expression for each subtype (a~c) or on originally observed expressions without deconvoluton (d). (Network interconnectedness is measured by topological overlap; cutHeight = 0.7; minSize = 8.)

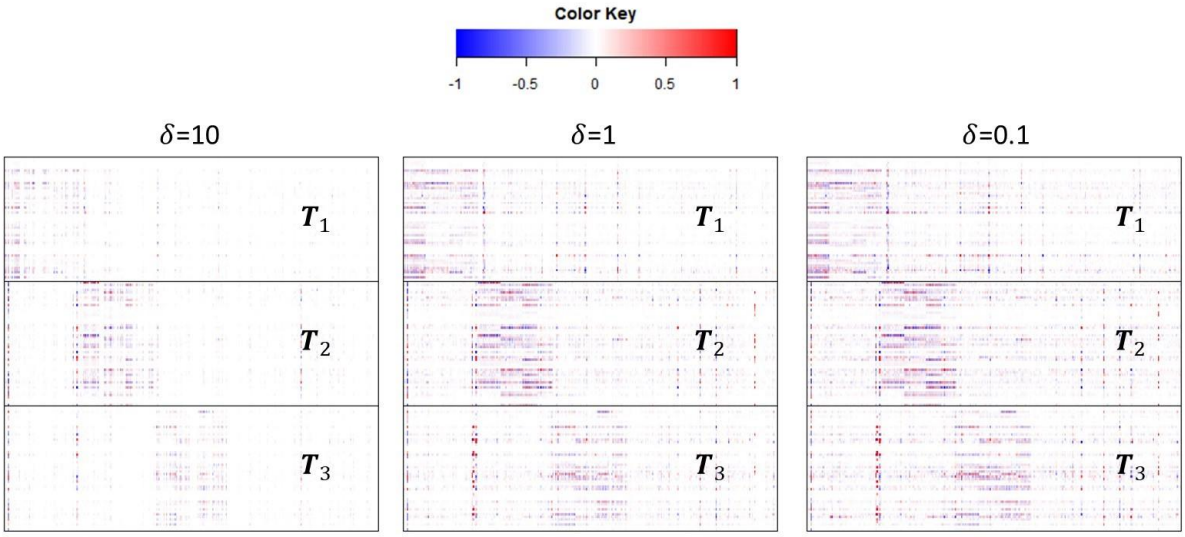

Fig. 11. Heatmap of estimated T matrix scaled by associated means compared to ground truth in the realistic simulation with $\lambda = 5$ and varied δ. Increasing the penalty of L21 norm will enforce more zero columns in ΔS_k matrix.

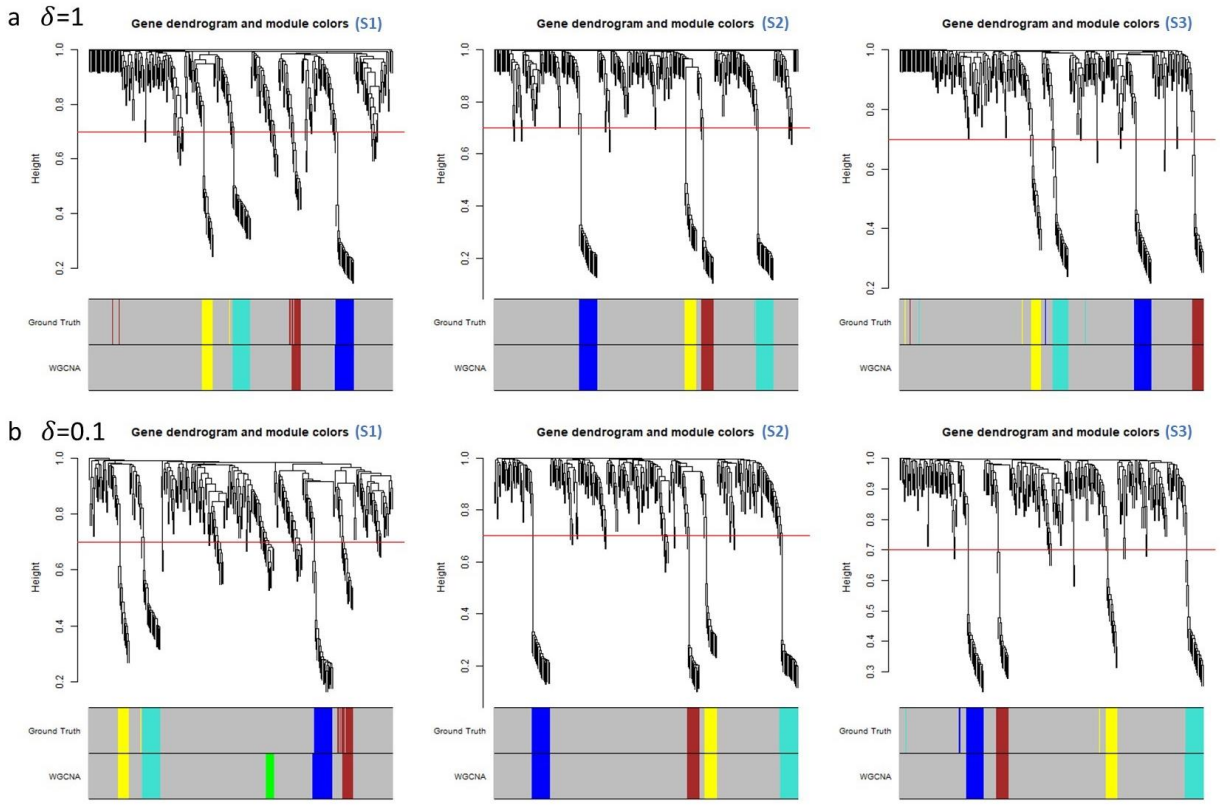

Fig. 12. Gene co-expressed function modules detected by WGCNA on swCAM estimated sample-specific expression for each subtype with λ=5 and δ=1 or 0.1. (Network interconnectedness is measured by topological overlap; cutHeight = 0.7; minSize = 8.)

Discussion

Most existing tissue deconvolution methods ignore the expression variability of subtypes across individual samples. swCAM will significantly expand the utility of CAM by producing subtype-specific expression profiles in each sample. The success of swCAM depends on the low-rank assumption, which takes advantage of biologically expected cooperation among genes and thus sheds light on solving the seemingly underdetermined sample-specific deconvolution problem. The low-rank assumption holds naturally in molecule expression data when there exist activated functional modules required by particular biological processes or pathways in different subtypes. The detection of such subtype-specific associations or networks is one of the major targets in the analysis of molecule expression profiles. After our sample-specific deconvolution by swCAM, conventional network analysis methods can be applied directly to the estimated sample-subtype-specific signals to construct subtype-specific networks, e.g. weighted correlation network analysis (WGCNA (Zhang and Horvath 2005, Langfelder and Horvath 2008)) and differential dependency network analysis (DDN (Zhang, Li et al. 2009, Zhang, Tian et al. 2011, Tian, Zhang et al. 2014, Tian, Zhang et al. 2015)).

The cross-validation strategy of excluding entries randomly is inspired by the similar ideas in matrix imputation methods that commonly assume the matrix to be recovered has a low rank. Our results consistently show a U-curve over parameter λ, demonstrating the feasibility of the proposed cross-validation strategy. Meanwhile, CAM is not sensitive to the choice of λ, as the U-curve has a wide platform where the recovered sample-subtype-specific signals are similar and detected modules are close.

It is also reasonable to assume that genes involved in biological associations or networks are sparse. Therefore, it deserves our further study to use $\ell_{2,1}$-norm regularization for reducing artifacts and improving function module detection.

When group information is available, we can also apply basic CAM algorithm to each group to obtain group-wise expression profiles of subtypes. Compared to sample-specific deconvolution, group-specific deconvolution aims at a lower resolution of underlying subtype signals and thus could obtain more robust results. If grouping is fine enough, group-specific deconvolution can also

acquire signal variation in each subtype and thus help detect function modules and construct biological networks.

Though swCAM can solve a seemingly underdetermined problem theoretically based on a low-rank assumption. It still needs improvement and validations. First, the improvement of swCAM by sparsity regularization. The sparsity assumption is practically reasonable, and we already show some preliminary results after imposing $\ell_{2,1}$ norm regularization. However, introducing one more regularization term will increase the difficulty of parameter tuning. Besides, the current cross-validation strategy with matrix entry sampling is not applicable to selecting the coefficient of $\ell_{2,1}$ norm term. Therefore, the integration of sparsity regularization still needs our further study. Second, the improvement of function module detection based on swCAM estimated sample-specific signals in each subtype. Recovering the exact values of sample-specific signals is impossible unless there are more strong assumptions. Luckily, our goal is to detect function module or networks from the between-sample variations in each subtype. Thus, increasing the accuracy of estimated intercorrelations among molecules can be regarded as our target of further efforts. Third, the validation of Validate swCAM in real data analysis. We have demonstrated the capacity of swCAM to estimate sample-specific signals in each subtype using simulations where the between-sample variation matrices are low-rank. Validation of swCAM in real molecule expression data would be difficult, as the benchmark datasets with true subtype-specific signals are unavailable. One possible direction is to verify the constructed subtype-specific networks through biological experiments.

Conclusion

We propose a sample-specific deconvolution algorithm to estimate simple-specific molecule expressions for each subtype, from which between-sample variation can be used to detect biological associations and construct networks in each subtype. The contributions of this work include: We formulate the objective function for swCAM with a penalty term to minimize the nuclear norm of between-sample variation matrix in each subtype, based on our expectation on the existence of subtype-specific networks. We design an efficient method based on ADMM to solve swCAM's optimization problem in large-scale biological data. We design a 10-fold cross-validation strategy to select the coefficient of nuclear norm term, and demonstrate its feasibility in

simulations where a U-curve of RMSE is obtained to determine the optimal selection. We validate swCAM in simulations to demonstrate sample-specific signals can be well estimated when low-rank assumption holds. Even though artificial signal variances exist in swCAM estimations, the intercorrelations among genes can still be well preserved for function module detection and biological network construction. We propose to use extra $\ell_{2,1}$ norm regularization to enforce the sparsity of genes involved in networks and thus reduce the artifacts trained from noise or from signals of other subtypes.

ACKNOWLEDGMENTS

This work has been supported by the National Institutes of Health under Grants HL111362-05A1, HL133932, NS115658-01, and the Department of Defence under Grant W81XWH-18-1-0723 (BC171885P1).

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

Reference

Boyd, S., N. Parikh, E. Chu, B. Peleato and J. Eckstein (2011). "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers." Found. Trends Mach. Learn. **3**(1): 1-122.

Buettner, F., K. N. Natarajan, F. P. Casale, V. Proserpio, A. Scialdone, F. J. Theis, S. A. Teichmann, J. C. Marioni and O. Stegle (2015). "Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells." Nat Biotechnol **33**(2): 155-160.

Cai, J.-F., E. J. Candès and Z. Shen (2010). "A Singular Value Thresholding Algorithm for Matrix Completion." SIAM Journal on Optimization **20**(4): 1956-1982.

Candes, E. J., C. A. Sing-Long and J. D. Trzasko (2013). "Unbiased Risk Estimates for Singular Value Thresholding and Spectral Estimators." Trans. Sig. Proc. **61**(19): 4643-4657.

Chasman, D. and S. Roy (2017). "Inference of cell type specific regulatory networks on mammalian lineages." Current Opinion in Systems Biology **2**(Supplement C): 130-139.

Chen, L. (2019). Mathematical Modeling and Deconvolution for Molecular Characterization of Tissue Heterogeneity. Ph.D. Doctoral Dissertation, Virginia Polytechnic Institute and State University.

Chen, L., Y. Lu, C.-T. Wu, R. Clarke, G. Yu, J. E. Van Eyk, D. Herrington and Y. Wang (2020). "Data-driven detection of subtype-specific differentially expressed genes." Scientific Reports.

Gal, E., M. London, A. Globerson, S. Ramaswamy, M. W. Reimann, E. Muller, H. Markram and I. Segev (2017). "Rich cell-type-specific network topology in neocortical microcircuitry." Nature Neuroscience **20**: 1004.

Hastie, T., R. Tibshirani and J. Friedman (2001). The Elements of Statistical Learning. New York, NY, USA, Springer New York Inc.

Junttila, M. R. and F. J. de Sauvage (2013). "Influence of tumour micro-environment heterogeneity on therapeutic response." Nature **501**: 346.

Kuhn, A., D. Thu, H. J. Waldvogel, R. L. Faull and R. Luthi-Carter (2011). "Population-specific expression analysis (PSEA) reveals molecular changes in diseased brain." Nat Methods **8**(11): 945-947.

Langfelder, P. and S. Horvath (2008). "WGCNA: an R package for weighted correlation network analysis." BMC Bioinformatics **9**: 559.

Parikh, N. and S. Boyd (2014). "Proximal Algorithms." Foundations and Trends® in Optimization **1**(3): 127-239.

Recht, B., M. Fazel and P. A. Parrilo (2010). "Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization." SIAM Review **52**(3): 471-501.

Shen-Orr, S. S., R. Tibshirani, P. Khatri, D. L. Bodian, F. Staedtler, N. M. Perry, T. Hastie, M. M. Sarwal, M. M. Davis and A. J. Butte (2010). "Cell type-specific gene expression differences in complex tissues." Nat Methods **7**(4): 287-289.

Sonawane, A. R., J. Platig, M. Fagny, C.-Y. Chen, J. N. Paulson, C. M. Lopes-Ramos, D. L. DeMeo, J. Quackenbush, K. Glass and M. L. Kuijjer "Understanding Tissue-Specific Gene Regulation." Cell Reports **21**(4): 1077-1088.

Thouvenin, P. A., N. Dobigeon and J. Y. Tourneret (2016). "Hyperspectral Unmixing With Spectral Variability Using a Perturbed Linear Mixing Model." IEEE Transactions on Signal Processing **64**(2): 525-538.

Tian, Y., B. Zhang, E. P. Hoffman, R. Clarke, Z. Zhang, I.-M. Shih, J. Xuan, D. M. Herrington and Y. Wang (2014). "Knowledge-fused differential dependency network models for detecting significant rewiring in biological networks." BMC Systems Biology **8**(1): 87.

Tian, Y., B. Zhang, E. P. Hoffman, R. Clarke, Z. Zhang, M. Shih Ie, J. Xuan, D. M. Herrington and Y. Wang (2015). "KDDN: an open-source Cytoscape app for constructing differential dependency networks with significant rewiring." Bioinformatics **31**(2): 287-289.

Wang, N., E. P. Hoffman, L. Chen, L. Chen, Z. Zhang, C. Liu, G. Yu, D. M. Herrington, R. Clarke and Y. Wang (2016). "Mathematical modelling of transcriptional heterogeneity identifies novel markers and subpopulations in complex tissues." Scientific Reports **6**: 18909.

Zhang, B. and S. Horvath (2005). "A general framework for weighted gene co-expression network analysis." Stat Appl Genet Mol Biol **4**: Article17.

Zhang, B., H. Li, R. B. Riggins, M. Zhan, J. Xuan, Z. Zhang, E. P. Hoffman, R. Clarke and Y. Wang (2009). "Differential dependency network analysis to identify condition-specific topological changes in biological networks." Bioinformatics **25**(4): 526-532.

Zhang, B., Y. Tian, L. Jin, H. Li, M. Shih Ie, S. Madhavan, R. Clarke, E. P. Hoffman, J. Xuan, L. Hilakivi-Clarke and Y. Wang (2011). "DDN: a caBIG(R) analytical tool for differential network analysis." Bioinformatics **27**(7): 1036-1038.