Adaptive Regularized Tri-Factor Non-Negative Matrix Factorization for Cell Type Deconvolution

Tianyi Liu[1] Chuwen Liu[1] Quefeng Li[1,*] Xiaojing Zheng[1,2,*] Fei Zou[1,3,*]

[1]Department of Biostatistics, The University of North Carolina at Chapel Hill

[2]Department of Pediatrics, The University of North Carolina at Chapel Hill

[3]Department of Genetics, The University of North Carolina at Chapel Hill

Abstract

Accurate deconvolution of cell types from bulk gene expression is crucial for understanding cellular compositions and uncovering cell-type specific differential expression and physiological states of diseased tissues. Existing deconvolution methods have limitations, such as requiring complete cellular gene expression signatures or neglecting partial biological information. Moreover, these methods often overlook varying cell-type mRNA amounts, leading to biased proportion estimates. Additionally, they do not effectively utilize valuable reference information from external studies, such as means and ranges of population cell-type proportions. To address these challenges, we introduce an Adaptive Regularized Tri-factor non-negative matrix factorization approach for deconvolution (ARTdeConv). We rigorously establish the numerical convergence of our algorithm. Through benchmark simulations, we demonstrate the superior performance of ARTdeConv compared to state-of-the-art semi-reference-based and reference-free methods. In a real-world application, our method accurately estimates cell proportions, as evidenced by the nearly perfect Pearson's correlation between ARTdeConv estimates and flow cytometry measurements in a dataset from a trivalent influenza vaccine study. Moreover, our analysis of ARTdeConv estimates in COVID-19 patients reveals patterns consistent with important immunological phenomena observed in other studies. The proposed method, ARTdeConv, is implemented as an R package and can be accessed on GitHub for researchers and practitioners.

Keywords: Cell-type deconvolution, Convergence analysis, Multiplicative update algorithm, Non-negative matrix factorization, RNA sequencing, Single cell data

1. Introduction

Heterogeneity in cell-type proportions exists across biological samples, and neglecting this heterogeneity in bulk gene expression can introduce bias into subsequent analyses, such

*Co-corresponding authors

as differential expression analyses. Conversely, acknowledging and accounting for this heterogeneity has shown clear advantages, yielding more accurate survival time predictions and tumor type classifications (Elloumi et al., 2011; Avila Cobos et al., 2020).

Laboratory techniques such as flow cytometry or immunohistochemistry are available for physically sorting cells into cell types and quantifying their abundances. However, these methods are often limited by the availability of cell samples, the specificity of antibodies for separating cells, and the substantial labor and time investments required (Newman et al., 2015; Spitzer and Nolan, 2016). Cell type deconvolution (simply referred to as "deconvolution" in this work), a computational process aimed at digitally separating heterogeneous mixture signals into their constituent components, has been critical in expediting the estimation of cell-type proportions from bulk gene expression data, such as RNA sequencing (RNA-seq). In recent years, several deconvolution methods have emerged, with extensive applications in the field of computational biology. These methods can generally be categorized into two groups: reference-based and reference-free, depending on whether they require individual cell-type gene expression signatures, referred to as a signature matrix, as prior knowledge (Li and Wu, 2019; Avila Cobos et al., 2020; Jin and Liu, 2021).

Gene signatures can be derived from either single-cell RNA-seq (scRNA-seq) data or sorted bulk RNA profiles of individual cell types and used as a reference (Chen et al., 2018). Although gene signatures have been successfully established for some cell types, acquiring those for other cell types might be labor-intensive or even unfeasible (Morris and Kopetz, 2016; Wigerblad et al., 2022). Neglecting to incorporate gene signatures for a prevalent cell type can induce biases in the proportional estimates of other cell types within reference-based deconvolution (Avila Cobos et al., 2020). On the other hand, reference-free techniques enable the unsupervised estimation of cell-type proportions but at the cost of disregarding information embedded in known gene signatures (Zaitsev et al., 2019; Chen et al., 2020). A deconvolution method that can utilize partial reference gene signatures presents a plausible compromise.

Additionally, external information on population cell-type proportions may also aid in estimating cell-type proportions. For example, in blood samples, the proportion means (or medians) and ranges of major cell types are available from complete blood count (CBC) tests. However, deconvolution techniques that effectively incorporate such information are currently lacking in the field.

During data pre-processing, information specific to the quantities of RNA molecules contained within an individual cell is often lost after library normalization. Zaitsev et al. (2019) found that failures to account for mRNA amounts produced biased cell-type proportion estimates. However, many deconvolution methods assume a generative model where the loss of this information cannot be accounted for (Avila Cobos et al., 2020). Therefore, incorporating these quantities into the generative model can be pivotal to deconvolution accuracy.

In recent years, several semi-reference-based methods have been developed to perform deconvolution using only partial references. EPIC addresses the lack of representation of cancer cells in common gene signature matrices for the deconvolution of the tumor microenvironment (TME) (Racle et al., 2017; Racle and Gfeller, 2020). Specifically, EPIC treats all types of cancer cells as a single super cell type in deconvolution, allowing for a detailed characterization of immune cell proportions in the TME while accounting for the fraction of cancer cells. This super-cell-type rationale can also be extended to the deconvolution of other tissue types, such as blood. Moreover, EPIC assigns weights

to each gene in the signature matrix based on its variability across cell types; genes with lower expression variance are given higher weights for deconvolution. EPIC also adjusts for differing mRNA amounts between cell types, which must be specified prior to deconvolution. However, this becomes infeasible if the tissue is complex, especially when there is a super cell type with multiple constituent cell types whose reference expression is missing. EPIC further requires that deconvolution be performed using signature genes not specific to the uncharacterized super cell type and estimates the proportions of cell types with reference using an inequality-constrained optimization procedure. Consequently, the estimated proportions of the cell type with a missing reference are biased toward zero (Tai et al., 2021).

Two other tools, quanTIseq and SECRET, have similar functions (i.e., they can accommodate one cell or super cell type with a missing reference), with the former essentially employing the same mathematical model as EPIC (Finotello et al., 2019; Lu et al., 2023). An innovation of quanTIseq over EPIC is its carefully defined set of reference gene signatures for immune cells and the calculation of cell-type mRNA amount scaling factors using housekeeping genes (Finotello et al., 2019), but it does not weigh the genes for deconvolution. These innovations also render quanTIseq unable to use customized signature matrices when needed. On the other hand, SECRET can incorporate customized gene signatures from scRNA-seq experiments and uses an outlier-insensitive L_1 norm on the residuals instead of EPIC's L_2 norm (Lu et al., 2023). However, it does not truly utilize any reference information about the uncharacterized cell type, making it prone to the same bias toward zero as occurs with EPIC.

BayICE is another semi-reference-based method serving the same purpose as the other methods above, but it adopts a hierarchical Bayes design with stochastic gene signature selection (Tai et al., 2021). Due to its probabilistic nature, BayICE can quantify the uncertainty of cellular abundance estimates and claims to mitigate the biases of EPIC. These benefits come at the expense of increased computational complexity, sensitivity to priors, and difficulty in interpreting the parameters involved. Moreover, it requires bulk samples of purified cells as reference inputs, which, like quanTIseq, precludes the use of other types of reference data, such as customized signature matrices.

In this article, we propose a novel semi-reference-based method called ARTdeConv (short for "Adaptive Regularized Tri-factor non-negative matrix factorization method for deConvolution"). It addresses the three outstanding issues discussed above: utilizing partial reference gene signatures, incorporating external information on cell-type proportions, and accounting for cell-type mRNA amounts. Compared to EPIC, ARTdeConv does not require the specificity of the gene signatures to cell types that are characterized. It also uses external proportion information to correct for EPIC's observed biases toward zero. Moreover, compared to EPIC and quanTIseq, ARTdeConv learns the relative cell-type mRNA abundances automatically without the need to use housekeeping genes or other information while maintaining the straightforward interpretation of the resulting estimates. It can also incorporate scRNA-seq data as SECRET does and uses a different optimization procedure. Our further contributions include deriving and implementing a multiplicative update (MU) algorithm for solving ARTdeConv, proving the algorithm's convergence, and demonstrating its merits through simulations and real data analysis. A schematic representation of ARTdeConv's workflow is shown in Figure 1. An R package implementing ARTdeConv is also available on GitHub.

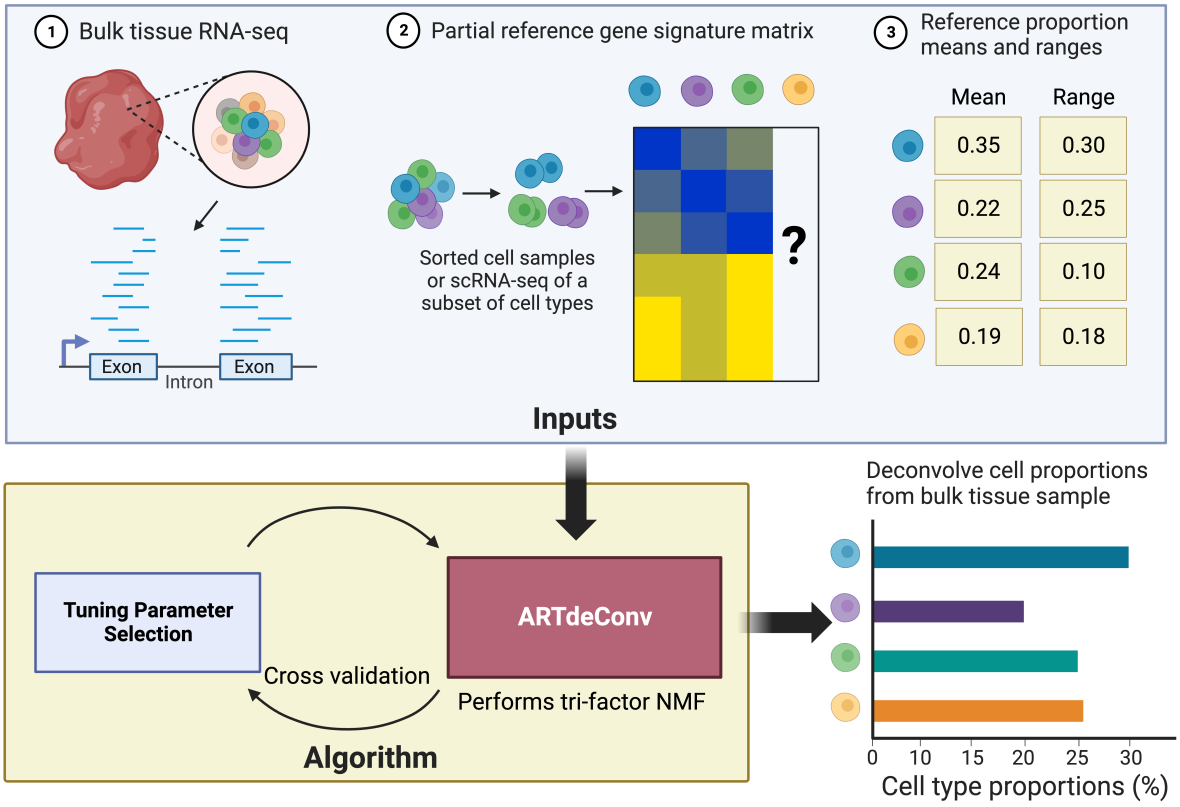

Figure 1: A schema of the ARTdeConv workflow. Specifically, ARTdeConv takes in three parts of information: the gene expression data of the bulk tissue to be deconvolved, the reference gene expression data with one cell type or super cell type uncharacterized, and the reference means/medians and ranges of the proportions of all cell types in deconvolution. It then passes these inputs to a cross validation algorithm to select the optimal hyperparameters. Lastly, these hyperparameters are passed along the rest of the inputs for a deconvolution run that estimate the proportions of each cell type or super cell type involved.

2. Method

2.1. Notation

Let $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote the set of non-negative and positive real numbers respectively. m, n, K are positive integers used to denote the number of genes, samples, and cell types with $K \leq \min(m, n)$ in deconvolution. Let a positive integer K_0 denote the number of cell types for which we have reference gene expression available. Unless otherwise mentioned, we set $K = K_0 + 1$. Let $\boldsymbol{Y} \in \mathbb{R}_+^{m \times n}$ be the matrix of bulk gene expression and $\boldsymbol{\Theta} \in \mathbb{R}_+^{m \times K}$ be the full gene signature matrix. Let $\boldsymbol{s} \in \mathbb{R}_{++}^K$ be a vector of relative cell-type mRNA amounts, and $\boldsymbol{P} \in \mathbb{R}_+^{K \times n}$ the proportion matrix. Let $\boldsymbol{\epsilon} \in \mathbb{R}^{m \times n}$ be the random error matrix. Unless otherwise specified, we denote $\boldsymbol{\theta}_k$ as the k-th column of $\boldsymbol{\Theta}$. Let $\boldsymbol{A}$ be a genereic $m \times n$ matrix. We use $\boldsymbol{A}^\top$ to denote the transpose of a matrix. If $\boldsymbol{A}$ is further a square matrix ($m = n$), we denote its trace $\mathrm{tr}(\boldsymbol{A}) = \sum_{i=1}^n A_{ii}$. For any $m \times n$ matrix $\boldsymbol{A}$, its Frobenius norm is denoted as $\|\boldsymbol{A}\|_F = \sqrt{\mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{A})} = \sqrt{\sum_{i=1}^m \sum_{j=1}^n A_{ij}^2}$. For two matrices $\boldsymbol{A}, \boldsymbol{B}$ of the same dimensions, $\boldsymbol{A} \odot \boldsymbol{B}$ denotes the element-wise product of $\boldsymbol{A}$ and $\boldsymbol{B}$ and $\frac{\boldsymbol{A}}{\boldsymbol{B}}$ denotes their element-wise quotient.

2.2. Model And Problem Setup

We first propose the following tri-factor generative model

$$\boldsymbol{Y} = \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P} + \boldsymbol{\epsilon} \tag{1}$$

for ARTdeConv, which extends the canonical model $\boldsymbol{Y} = \boldsymbol{\Theta}\boldsymbol{P} + \boldsymbol{\epsilon}$ (Avila Cobos et al., 2020). The difference between them is $\mathrm{diag}(\boldsymbol{s})$, a diagonal matrix of cell-type mRNA amounts. In practice, normalization (e.g. by library sizes, etc.) is frequently employed to alleviate between-sample technical artifacts in sequencing data (Hafemeister and Satija, 2019; Lytal et al., 2020). During this process, information regarding the quantities of RNA molecules packed within different kinds of cells sometimes gets lost. Failures to recover these quantities lead to documented biases in estimating cell-type proportions (Racle et al., 2017; Zaitsev et al., 2019). Should this occurs, $\mathrm{diag}(\boldsymbol{s})$ can account for the RNA molecule quantities.Otherwise, $\mathrm{diag}(\boldsymbol{s})$ would be close to the identity matrix $\boldsymbol{I}_K$ (an option to fix $\mathrm{diag}(\boldsymbol{s}) = \boldsymbol{I}_K$ is given by ARTdeConv). The values in $\mathrm{diag}(\boldsymbol{s})$ should be interpreted in a relative sense. For example, s_2/s_1 is the ratio between the amounts of mRNA molecules packed by the second and the first cell type. We do not require any prior specification of $\mathrm{diag}(\boldsymbol{s})$ and notice that (1) is similar to the model of MuSiC, a reference-based method where each sample is accorded an additional scaling factor c_i to account for between-sample measurement variation in bulk gene expression (Wang et al., 2019). Moreover, this setup differs from that of EPIC, which requires $\boldsymbol{s}$ be manually specified (Racle et al., 2017).

For performing deconvolution based on (1), ARTdeConv assumes prior knowledge of a partial signature matrix $\boldsymbol{\Theta}_{K_0} \in \mathbb{R}_+^{m \times K_0}$ as well as $\boldsymbol{Y}$. Then, an objective function is defined:

$$f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) = \frac{1}{2mn}\|\boldsymbol{Y} - \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\|_F^2 + \frac{\alpha_1}{2}R_1 + \frac{\alpha_2}{2}R_2 + \frac{\beta}{2}R_3, \tag{2}$$

where R_1, R_2, and R_3 are regularizers to be explained later and α_1, α_2, and β are their tuning parameters. We discuss the selection of optimal tuning parameters in Section 2.4.

$\boldsymbol{\Theta}$, $\boldsymbol{s}$, and $\boldsymbol{P}$ are then estimated via

$$(\hat{\boldsymbol{\Theta}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{P}}) = \underset{\boldsymbol{\Theta}\in\mathbb{R}_+^{m\times K},\ \boldsymbol{s}\in\mathbb{R}_{++}^{K},\ \boldsymbol{P}\in\mathbb{R}_+^{K\times n}}{\operatorname{argmin}} f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}). \tag{3}$$

The main objective here is to obtain an estimate of the proportion matrix $\hat{\boldsymbol{P}}$. To follow a common practice of deconvolution and simplify the algorithm (Mohammadi et al., 2017), ARTdeConv does not directly constrain each column of $\boldsymbol{P}$ to sum to one during the estimation process. Instead, it obtains an unconstrained estimate $\hat{\boldsymbol{P}}$ and then re-normalizes its columns to have the unit sum.

Let $\boldsymbol{\Theta}_0 = \begin{bmatrix}\boldsymbol{\Theta}_{K_0} & \mathbf{0} & \cdots & \mathbf{0}\end{bmatrix} \in \mathbb{R}_+^{m\times K}$ and $\boldsymbol{\Delta}$ be an $m \times K$ matrix such that $\Delta_{jk} = I(k \leq K_0)$ for $1 \leq j \leq m$. The squared Frobenius distance between the estimated $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}_0$ is penalized through $R_1 = \|\boldsymbol{\Delta} \odot (\boldsymbol{\Theta} - \boldsymbol{\Theta}_0)\|_F^2$. Though we present a special case where $\boldsymbol{\Theta}_{K_0}$ occupies the first K_0 columns of $\boldsymbol{\Theta}$, by re-defining $\boldsymbol{\Delta}$ correspondingly, $\boldsymbol{\Theta}_{K_0}$ can occupy any K_0 columns, covering all structures of prior knowledge on the signature matrix. Same as other well established semi-reference-based methods (i.e., EPIC, quanTIseq, etc.), we set $K = K_0 + 1$. We also recommend that the unrepresented cell types in $\boldsymbol{\Theta}_{K_0}$ be grouped into a single artificial "cell type". Albeit that ARTdeConv technically allows $K > K_0 + 1$, we have found that the resulting estimates from ARTdeConv became less reliable as $K - K_0$ grew larger in preliminary analysis (results not shown). This makes ARTdeConv suitable for the situations where a major cell type is missing from the reference or that distinguishing the multiple cell types that are missing is not salient. We shall see such examples at work late in Sections 3.2 and 3.3.

On the other hand, R_1 is not a strictly convex function of $\boldsymbol{\Theta}$, making the guarantee of the numerical convergence of ARTdeConv's algorithm difficult (more details on this later). To atone for this, we make $R_2 = \|(\boldsymbol{J}_{m\times K} - \boldsymbol{\Delta}) \odot \boldsymbol{\Theta}\|_F^2$, where $\boldsymbol{J}_{m\times K}$ is an $m \times K$ matrix of 1s. It can be shown that any positive linear combination of R_1 and R_2 is strictly convex with respect to $\boldsymbol{\Theta}$. It is of notes that R_2 will force the estimated gene signatures of the uncharacterized close to zero. While this is the case for genes that are specific to the cell types in $\boldsymbol{\Theta}_{K_0}$, ARTdeConv does not restrict the gene signatures to only those genes. Thus, it is recommended to reduce the penalization effects R_2 by setting its tuning parameter α_2 to a very small value.

Information on the cell-type proportions such as means (or medians) and ranges in a population frequently exists in external data and is accessible online, for example through complete blood counts of leukocytes (Takami et al., 2021). Let m_k denote the reference mean (or median) of cell type k's population proportion and let r_k denote its range. To incorporate it into our deconvolution method, we propose adding another penalty $R_3 = \sum_{k=1}^{K}(1/r_k)\|\boldsymbol{p}_k^\top - m_k\mathbf{1}_n\|_2^2$, where $\boldsymbol{p}_k^\top$ is the k-th row of $\boldsymbol{P}$. In R_3, m_k acts as a pivot, and the deviation from which is penalized for the estimated $\boldsymbol{p}_k^\top$. Meanwhile, $1/r_k$ acts as a weight so the proportion estimates of cell types with wider ranges are less penalized for departing from their respective m_k. Letting $\boldsymbol{M}$ be a $K \times n$ matrix such that $\boldsymbol{M} = \begin{bmatrix}m_1\mathbf{1}_n & m_2\mathbf{1}_n & \cdots & m_K\mathbf{1}_n\end{bmatrix}^\top$ and $\boldsymbol{\rho} = \operatorname{diag}(r_1, r_2, \cdots, r_K)$, we can write R_3 in a matrix form $R_3 = \|\boldsymbol{\rho}^{-1/2}(\boldsymbol{P} - \boldsymbol{M})\|_F^2$.

2.3. Multiplicative Update Algorithm

A multiplicative update (MU) algorithm is proposed to solve (3). The MU algorithm was originally designed to solve the canonical bi-factor non-negative matrix factorization (NMF) problem (Lee and Seung, 2000) and was extensively studied (Lin, 2007). Technically, it

is similar to the majorization-minimization algorithm of Lange (2016). It can be readily extended to solving a multi-factor regularized NMF problem like the one for ARTdeConv.

Let t be a non-negative integer denoting the current number of iteration. The multiplicative update steps are derived by finding the gradients of a set of auxiliary functions for each row of $\boldsymbol{\Theta}^t$, each column of $\boldsymbol{P}^t$, and $\boldsymbol{s}^t$. These update steps are:

$$\boldsymbol{\Theta}^{t+1} = \boldsymbol{\Theta}^t \odot \frac{\boldsymbol{Y}(\boldsymbol{P}^t)^\top \mathrm{diag}(\boldsymbol{s}^t) + mn\alpha_1 \boldsymbol{\Delta} \odot \boldsymbol{\Theta}_0}{\boldsymbol{\Theta}^t \mathrm{diag}(\boldsymbol{s}^t)\boldsymbol{P}^t(\boldsymbol{P}^t)^\top \mathrm{diag}(\boldsymbol{s}^t) + mn\left\{\alpha_1 \boldsymbol{\Delta} \odot \boldsymbol{\Theta}^t + \alpha_2(\boldsymbol{J}_{m\times K} - \boldsymbol{\Delta}) \odot \boldsymbol{\Theta}^t\right\}}; \tag{4}$$

$$\boldsymbol{P}^{t+1} = \boldsymbol{P}^t \odot \frac{\mathrm{diag}(\boldsymbol{s}^t)(\boldsymbol{\Theta}^{t+1})^\top \boldsymbol{Y} + mn\beta\boldsymbol{\rho}^{-1}\boldsymbol{M}}{\mathrm{diag}(\boldsymbol{s}^t)(\boldsymbol{\Theta}^{t+1})^\top \boldsymbol{\Theta}^{t+1}\mathrm{diag}(\boldsymbol{s}^t)\boldsymbol{P}^t + mn\beta\boldsymbol{\rho}^{-1}\boldsymbol{P}^t}; \tag{5}$$

$$\boldsymbol{s}^{t+1} = \boldsymbol{s}^t \odot \frac{\boldsymbol{u}^t}{\boldsymbol{Z}^t\boldsymbol{s}^t}, \tag{6}$$

where $u_k^t = \mathrm{tr}\left\{\boldsymbol{Y}^\top \boldsymbol{\theta}_k^{t+1}\left(\boldsymbol{p}^{t+1}\right)_k^\top\right\}$ and $Z_{kl}^t = \left\{(\boldsymbol{\theta}^{t+1})_k^\top \boldsymbol{\theta}_l^{t+1}\right\}\left\{(\boldsymbol{p}^{t+1})_k^\top \boldsymbol{p}_l^{t+1}\right\}$ for $k, l = 1, 2, \cdots, K$.

Mathematical details for finding the auxiliary functions and the update steps can be found in Supplementary Material Section B. The pseudo-code of the MU algorithm is shown in Algorithm 1.

Algorithm 1 The MU Algorithm For Solving ARTdeConv

Initialize $\boldsymbol{\Theta}^0 \in \mathbb{R}_+^{m\times K}$ such that $R_1 = 0, \boldsymbol{P}^0 \in \mathbb{R}_+^{K\times n}, \boldsymbol{s}^0 \in \mathbb{R}_{++}^K$. Fix α_1, α_2, and $\beta > 0$.

Repeat

Update $\boldsymbol{\Theta}^t$ to $\boldsymbol{\Theta}^{t+1}$ via (4);

Update $\boldsymbol{P}^t$ to $\boldsymbol{P}^{t+1}$ via (5);

Update $\boldsymbol{s}^t$ to $\boldsymbol{s}^{t+1}$ via (6);

Until

$$\frac{|f(\boldsymbol{\Theta}^{t+1}, \boldsymbol{s}^{t+1}, \boldsymbol{P}^{t+1}) - f(\boldsymbol{\Theta}^t, \boldsymbol{s}^t, \boldsymbol{P}^t)|}{f(\boldsymbol{\Theta}^t, \boldsymbol{s}^t, \boldsymbol{P}^t)} < \delta,$$

where δ is a small tolerance parameter and f defined as in (2). Define the final outputs as $(\hat{\boldsymbol{\Theta}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{P}})$. Re-normalize $\hat{\boldsymbol{P}}$ by dividing each column of by its sum.

The MU algorithm has two advantages (under the assumptions discussed in Supplementary Material Section A). First, $\boldsymbol{\Theta}^t$, $\boldsymbol{P}^t$, and $\boldsymbol{s}^t$ remain non-negative throughout all iterations if their initial values are non-negative. It is recommended that $\boldsymbol{\Theta}^0$ is set to satisfy $R_1 = 0$, so any zero in the known partial signature matrix will remain zero. Second, the MU algorithm can be shown to achieve numerical convergence. If we enforce $\mathrm{diag}(\boldsymbol{s}^t) = \boldsymbol{I}_K$ and set $\alpha_1 = \alpha_2 = \beta = 0$, our MU algorithm is reduced to that of Lee and Seung (2000) for the canonical NMF. Due to the non-linearity of the objective function (2), it is not guaranteed that a global minimizer can be produced. Following the precedence of other NMF-based deconvolution software, we recommend restarting the MU algorithm multiple times and choosing the run with the smallest Frobenius norm of the residuals on the estimated bulk expression (Gaujoux and Seoighe, 2010, 2013). We set 10 restarts as the default in our implementation.

We now describe the default initialization procedure of ARTdeConv. First, the first K_0 columns of $\boldsymbol{\Theta}^0$ were used as $\boldsymbol{\Theta}_{K_0}^*$ to satisfy $R_1 = 0$. Each initial value in the last column

θ^0_{jK} is generated independently by $\theta^0_{jK} \sim |N(\bar{y}_j, \hat{\sigma}^2_j)|$, where $\bar{y}_j$ and $\hat{\sigma}^2_j$ are the mean and variance of the bulk expression values for gene j across all n samples. This differs from the procedure of SECRET (Lu et al., 2023), which initiates $\theta^0_{jK} = 0$. This is because the MU algorithm would produce $\theta^t_{jK} = 0$ for any $t = 1, 2, \ldots, T$ following this initiation. Then, for each cell type k of sample i, p^0_{ki} is generated from a $N(m_k, 0.1)$ distribution. After this, negative entries in $\boldsymbol{P}^0$ are corrected to 0.01. Lastly, $\boldsymbol{s}^0$ is initialized by solving

$$\boldsymbol{s}^0 = \underset{\boldsymbol{s} \in \mathbb{R}^K}{\operatorname{argmin}} \frac{1}{2} \|\boldsymbol{Y}^* - \boldsymbol{\Theta}^0 \operatorname{diag}(\boldsymbol{s}) \boldsymbol{P}^0\|_F^2.$$

Negative values in $\boldsymbol{s}^0$ will be replaced by the mean of positive values.

2.4. Tuning Parameter Selection

A grid search using a B-fold cross-validation (CV) is designed for selecting tuning parameters. Since R_2 is designed to make the objective function in (2) strictly convex, α_2 is advised to be fixed at a minuscule value. In the implementation of ARTdeConv, $\alpha_2 = 10^{-12}$ is the default value.

To begin, $\boldsymbol{Y}$ is randomly divided into B different folds, each containing $1/B$ columns of $\boldsymbol{Y}$. Two grids $\mathfrak{A}_1, \mathfrak{B} \subset \mathbb{R}_{++}$ containing respective candidates values for α_1 and β are declared. Next, for one $\alpha_1 \in \mathfrak{A}_1$, one $\beta \in \mathfrak{B}$ and a fold $b \in \{1, 2, \ldots, B\}$, the columns in the fold are held out as the test set $\boldsymbol{Y}^{(b)}_{\text{test}}$. The rest of the columns in $\boldsymbol{Y}$ are used as the training set $\boldsymbol{Y}^{(b)}_{\text{train}}$, upon which an ARTdeConv solution $(\tilde{\boldsymbol{\Theta}}^{(b)}, \tilde{\boldsymbol{s}}^{(b)}, \tilde{\boldsymbol{P}}^{(b)})$ is obtained.

Then, since it is assumed that n samples should share the same $\boldsymbol{\Theta}$ and $\boldsymbol{s}$ in (1), the estimated proportions $\hat{\boldsymbol{P}}^{(b)}$ on the test set $\boldsymbol{Y}^{(b)}_{\text{test}}$ is obtained by

$$\hat{\boldsymbol{P}}^{(b)} = \underset{\boldsymbol{P} \in \mathbb{R}_+^{K \times n/B}}{\operatorname{argmin}} \|\boldsymbol{Y}^{(b)}_{\text{test}} - \tilde{\boldsymbol{\Theta}}^{(b)} \operatorname{diag}(\tilde{\boldsymbol{s}})^{(b)} \boldsymbol{P}\|_F^2.$$

Computationally, this is accomplished by solving a non-negative least square problem using the R package `nnls`. The CV error given α_1 and β is

$$\operatorname{Err}(\alpha_1, \beta) = \frac{1}{B} \sum_{b=1}^{B} \|\boldsymbol{Y}^{(b)}_{\text{test}} - \tilde{\boldsymbol{\Theta}}^{(b)} \operatorname{diag}(\tilde{\boldsymbol{s}})^{(b)} \hat{\boldsymbol{P}}^{(b)}\|_F^2.$$

Finally, we select the best tuning parameters α_1^* and β^* that minimize $\operatorname{Err}(\alpha_1, \beta)$.

2.5. Convergence Analysis

In this section, we prove the numerical convergence of Algorithm 1. The main result depends on two reasonable assumptions in Supplementary Material Section A. Consequently, the following theorem holds for Algorithm 1 for solving the problem in (3).

Theorem 1 (Convergence of Algorithm 1 to a Stationary Point)**.** *Under the technical assumptions listed in Supplementary Material Section A, the sequence* $\{(\boldsymbol{\Theta}^t, \boldsymbol{s}^t, \boldsymbol{P}^t)\}_{t=1}^{\infty}$ *in Algorithm 1 converges to a stationary point of* $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$.

The proof of Theorem 1 relies on showing that it is a special case of the block successive upper-bound minimization (BSUM) algorithm given in Razaviyayn et al. (2013). Thus, the convergence is guaranteed by related BSUM theories. Doing so involved proving that the

sub-level set $\mathcal{X}^0 = \{(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) : f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) \leq f(\boldsymbol{\Theta}^0, \boldsymbol{s}^0, \boldsymbol{P}^0)\}$ is compact and the objective function in (3) is coercive, as well as demonstrating the majorization-minimization (MM) properties and strong convexity of the auxiliary functions used to derive the update steps in Algorithm 1. Details on the definitions of coercive functions and MM properties, as well as the proof of Theorem 1 are relegated to Supplementary Material Section C.

3. Results

3.1. Deconvolution Performance Benchmarks on Pseudo-Bulk Samples

To assess the deconvolution performance of ARTdeConv in comparison to alternative methods, we conducted a benchmarking simulation study by evaluating it against two semi-reference-based methods: EPIC and SECRET, and three reference-free methods: NMF (Gaujoux and Seoighe, 2010), debCAM (Chen et al., 2020), and LINSEED (Zaitsev et al., 2019). We deemed the NMF application here semi-supervised and called it "Semi-NMF" due to the prior knowledge on $\boldsymbol{\Theta}$, which was used as the initial values for the basis matrix in NMF. BayICE and quanTIseq were not included due to their required inputs' incompatibility with the simulation setup.

We generated a set of pseudo-bulk expression matrices using methodologies similar to those outlined in other previous studies (Avila Cobos et al., 2020) by the following formula

$$\boldsymbol{Y}^* = \boldsymbol{\Theta}^* \mathrm{diag}(\boldsymbol{s}^*) \boldsymbol{P}^* + \boldsymbol{\epsilon}. \tag{7}$$

Here, simulated cellular gene expression for $K = 5$ hypothetical cell types, CT1 - CT5, were constructed with assumed prior knowledge on $K_0 = 4$ of them (CT1 - CT4). We also directly simulated the true signature matrix $\boldsymbol{\Theta}^*$, as opposed to using aggregated purified bulk data or single cell data, for mimicking the widely common practice of using pre-constructed signature matrices in deconvolution applications (Newman et al., 2015; Monaco et al., 2019).

In total, $n = 200$ samples were simulated on $\boldsymbol{P}^*$. To investigate the effects of the true relative abundance of CT5 in bulk tissue samples on deconvolution results, three classes of $\boldsymbol{P}^*$ were generated, representing when CT5 was rare, uniform, and extra compared to other cell types in the tissue. Reference medians $(m_1, \ldots, m_5)$ and ranges $(r_1, \ldots, r_5)$ were obtained directly from each row of $\boldsymbol{P}^*$.

The expression values in the true signature matrix $\boldsymbol{\Theta}^*$ were simulated row-by-row (i.e., gene-by-gene), controlled by a parameter γ, which dictated how cell-type-specific each gene was on average. Each simulated gene was more likely to be cell-type-specific when $\gamma = 1$ compared to when $\gamma = 0$. The expression of $M = 2,000$ genes was first created in a matrix $\boldsymbol{\Theta}^*_{\text{full}}$. Then, from these genes, a subset of $m = 1,000$ highly cell-type-specific genes called marker genes were selected for ARTdeConv, Semi-NMF, EPIC, and SECRET. In addition, $\boldsymbol{s}^*$ (including s^*_5) was given to EPIC as for adjusting the mRNA amount per cell type. The method for the selection is described in details in Supplementary Material Section D.2. The expression of those selected genes was stored in $\boldsymbol{\Theta}^*$. The matrix $\boldsymbol{\Theta}^*_{K_0}$ consisting of the first four columns of $\boldsymbol{\Theta}^*$ was used as the partial signature matrix for ARTdeConv, EPIC, and SECRET, and the initial value in the basis matrix for Semi-NMF. Each gene was given the same unit weight in EPIC and SECRET during simulation, for all of the simulated gene expression were derived from the identical generative process.

On the other hand, debCAM and LINSEED searched for marker genes differently from $\boldsymbol{\Theta}^*_{\text{full}}$ by looking for simplicial vertices of the vector space spanned by normalized bulk gene expression (Zaitsev et al., 2019; Chen et al., 2020).

The matrix of errors $\boldsymbol{\epsilon}$ was first generated for all of the $M = 2,000$ simulated genes. Beginning with $(\boldsymbol{\Theta}^*_{\text{full}}, \boldsymbol{s}^*, \boldsymbol{P}^*)$, the error-free full bulk expression matrix was calculated as $\boldsymbol{\Theta}^*_{\text{full}}\text{diag}(\boldsymbol{s}^*)\boldsymbol{P}^*$. The relative strength of the errors (noises) to the mean expression of genes (signals) was controlled by another parameter σ. Two levels of noises with $\sigma = 0.1$ or 10 (low vs. high) were introduced to evaluate the robustness of the methods to added noises. The final bulk expression matrix of all simulated genes was calculated using $\boldsymbol{Y}^*_{\text{full}} = \boldsymbol{\Theta}^*_{\text{full}}\text{diag}(\boldsymbol{s}^*)\boldsymbol{P}^* + \boldsymbol{\epsilon}$. From this, a sub-matrix $\boldsymbol{Y}^*$ corresponding to the genes in $\boldsymbol{\Theta}^*$ was selected. A detailed description of the probability distributions and their parameters in the generation of $\boldsymbol{\Theta}^*$, $\boldsymbol{s}^*$, $\boldsymbol{P}^*$, and $\boldsymbol{\epsilon}$ can be found in Supplementary Material Section D.1.

With necessary parts generated, 100 simulations were conducted for each combination of γ, σ, and CT-5 abundance class, where the deconvolution performance of all benchmarked methods was reported. The tuning grid and algorithm parameters for ARTdeConv are described in Supplementary Material Section D.3. Due to a lack of identifiability, estimated proportions from Semi-NMF, debCAM, and LINSEED were manually matched to different cell types, whose details were relegated to Supplementary Material Section D.4.

Performance metrics for evaluating deconvolution results included the following: (a) $\frac{1}{K}\sum_{l=1}^{K}\text{CCC}(\boldsymbol{p}_l^{*\top}, \hat{\boldsymbol{p}}_l^{\top})$, (b) $\frac{1}{5n}\sum_{i=1}^{n}\sum_{k=1}^{5}|p^*_{ki} - \hat{p}_{ki}|$, (c) $\text{CCC}(\boldsymbol{p}_5^{*\top}, \hat{\boldsymbol{p}}_5^{\top})$, (d) $\frac{1}{n}\sum_{i=1}^{n}|p^*_{5i} - \hat{p}_{5i}|$, where CCC denotes the Canonical Correlation Coefficient (Lin, 1989; Huang et al., 2023) between two vectors. An advantage of CCC over Pearson's correlation is that CCC directly measures the agreement between two sets of values by penalizing deviations from the 45-degree line in a scatterplot. Among the four metrics, (a) and (b) delineated the overall deconvolution accuracy, while (c) and (d) described the deconvolution accuracy for CT5, the missing cell type.

Overall, ARTdeConv demonstrated superior performance compared to other semi-reference-based and reference-free methods in accurately recovering cell-type proportions, irrespective of the cell-type specificity of signature genes, the level of additive noise, and the relative abundance of the missing cell type compared to other cell types (Figures 2a and 3a). Notably, ARTdeConv also exhibited robust performance in estimating the proportions of CT5, especially when CT5 was relatively prevalent in the pseudo-bulk samples under high noise conditions, where other methods, except debCAM, showed reduced precision (Figures 2b and 3b).

Among the semi-reference-based methods, EPIC achieved high overall accuracy when CT5 was relatively rare in the pseudo-bulk samples. However, its performance in estimating the proportion of CT5 alone was surpassed by SECRET (Figure 2). While the overall performance of EPIC was comparable to that of SECRET when CT5 had similar abundance to other cell types, EPIC's accuracy diminished when CT5 was relatively common among the pseudo-bulks. This is consistent with previous findings that EPIC tends to underestimate the proportions of cell types not characterized in the reference (Tai et al., 2021). Although SECRET showed higher accuracy than EPIC in scenarios where CT5 was common, its overall performance remained inferior to ARTdeConv and did not significantly enhance the accuracy of CT5 estimates over EPIC, except when CT5 was relatively rare, where it performed slightly better than ARTdeConv (Figures 2 and 3).

Among the reference-free methods, debCAM demonstrated accuracy that exceeded those of semi-reference-based methods when CT5 was relatively common (Figures 2 and 3).

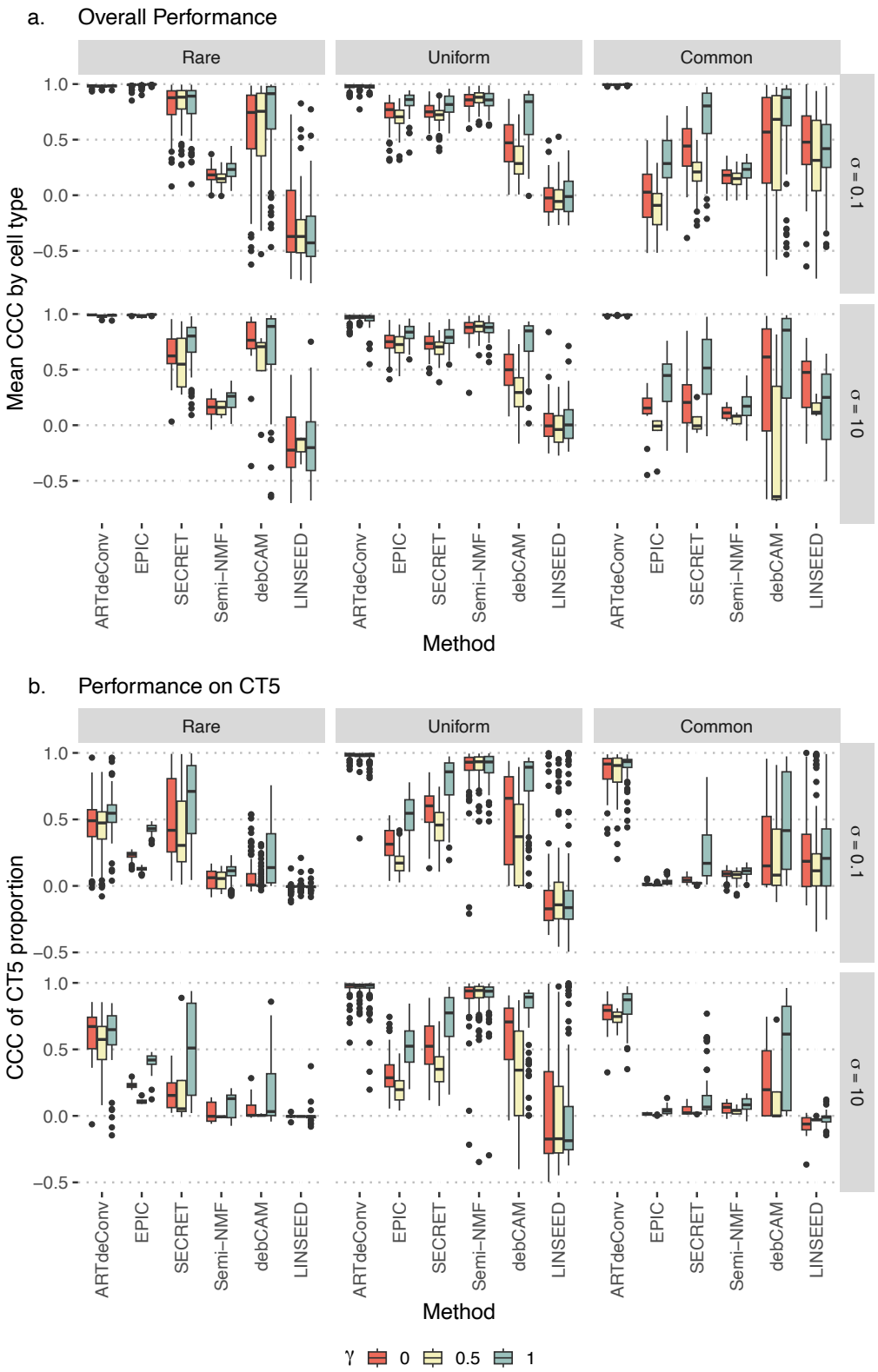

Figure 2: **a.** Mean Concordance Correlation Coefficient (CCC) by cell type between true proportions and estimated cell-type proportions from simulated pseudo-bulks. **b.** CCC between estimated and true proportions of CT5 in the pseudo-bulks for the benchmark simulations. Each column represents a case of the relative abundance of CT5 against other cell types in the simulated pseudo-bulks. Each row represents the level of noises controlled by σ. Colors represent different levels of cell-type specificity of genes in $\mathbf{\Theta}^*$ regulated by different γ.

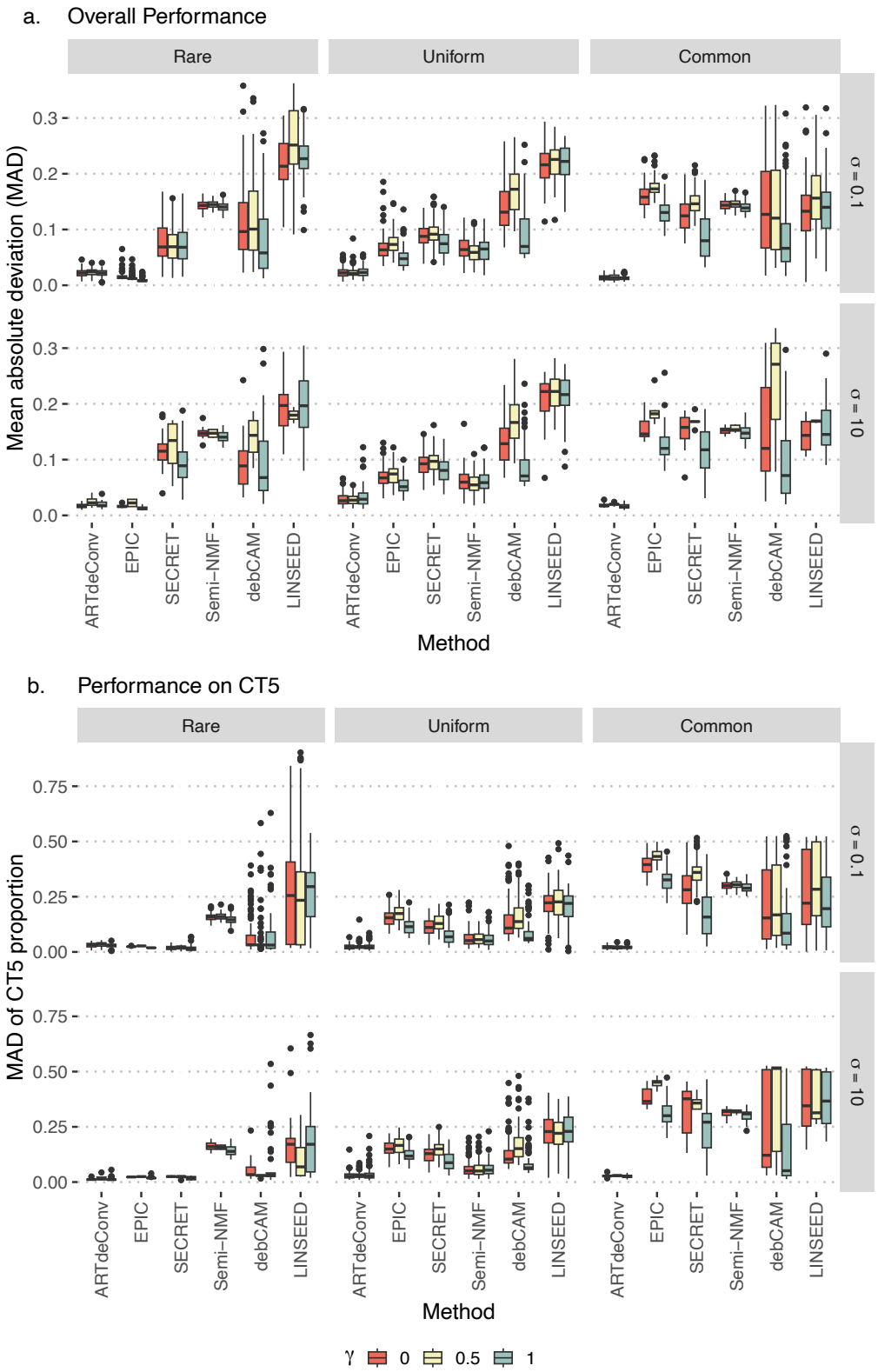

Figure 3: **a.** Mean absolute deviation (MAD) between true proportions and estimated cell-type proportions from simulated pseudo-bulks. **b.** MAD between estimated and true proportions of CT5 in the pseudo-bulks for the benchmark simulations. Each column represents a case of the relative abundance of CT5 against other cell types in the simulated pseudo-bulks. Each row represents the level of noises controlled by σ. Colors represent different levels of cell-type specificity of genes in $\mathbf{\Theta}^*$ regulated by different γ.

As expected, debCAM's performance improved with highly cell-type-specific signature genes ($\gamma = 1$). Conversely, Semi-NMF performed well when CT5 had similar abundance to other cell types in the pseudo-bulk samples. LINSEED, known for its insensitivity to cell-type mRNA amounts, consistently showed the worst performance among the methods, except in a few specific cases.

In conclusion, ARTdeConv showed superior performance to its semi-reference-based peers and reference-free methods in our simulations. We further remark that, in the simulations described above, $\mathbf{\Theta}^*$ included knowledge of marker genes for CT5. In practice, obtaining the marker genes of cell types without reference expression is, while not as straightforward, achievable through methods such as in Wang et al. (2016) and Qiu et al. (2021). Furthermore, even without well defined marker genes for those cell types, deconvolution by ARTdeConv is still completely feasible, for instance the analyses in Sections 3.2 and 3.3.

3.2. ARTdeConv Accurately Estimates Cell Type Proportions of PBMC from Bulk Gene Expression in a Human Influenza Vaccine Study

To benchmark the performance of ARTdeConv on real-world data and exemplify its application, we utilized a dataset sourced from Hoek et al. (2015). In this study, blood samples were obtained from healthy volunteer subjects who were administered a trivalent inactivated influenza vaccine (TIV). Both bulk PBMC samples and sorted PBMC cell lines from two enrolled subjects ("HD30" and "HD31") vaccinated with a single dose of 2011–2012 seasonal TIV were collected at four different time points: before vaccination (Day 0), and on Day 1, 3, and 7 post-vaccination.

Deconvolution was performed on eight bulk PBMC samples whose gene expression were measured in transcript per million (TPM). Gene expression in TPM from the sorted PBMC cell lines of the two subjects on Day 0 were used to construct the partial signature matrix. The study characterized four distinct PBMC cell types: T cells, B cells, natural killer (NK) cells, and monocytes. Additionally, there were other unspecified cell types collectively termed into one super cell type as "others", which could include, for instance, various types of dendritic cells and low density neutrophils (Kleiveland, 2015; Monaco et al., 2019). Detailed data processing steps could be referred to at Supplementary Material Section E.1.

The means and ranges of the cell-types in question were obtained through the reference values in Kleiveland (2015). For each cell type, the proportion means were calculated by taking the average of the upper and lower bounds of the reference values, while the ranges were calculated by taking the difference between the two bounds. For the "others" cell type, the mean was calculated by subtracting the means of all other cell types from one, and the range the difference between the upper and lower possible proportions of the other cell types. The exact values of the means and ranges are reported in Supplementary Material Table 1 (details of the calculation are given in Supplementary Material Section E.2). The vector of ranges was further normalized such that it has unit L_2 norm before deconvolution. Additional details regarding the tuning grid and algorithm parameter setup of this analysis can be found at Supplementary Material Section E.3.

The PBMC percentages on Day 0 were measured by flow cytometry (Hoek et al., 2015; Racle et al., 2017), which were used as the ground truths for evaluating the performance of ARTdeConv. Indeed, ARTdeConv achieved a notable degree of precision in estimating cell-

type proportions on Day 0 as compared to the flow cytometry measurements (Figure 4a), achieving a canonical correlation coefficient (CCC) of 0.974 (Pearson's correlation = 0.974, MAD = 0.036). This performance is on par with that of EPIC and surpassed those of CIBERSORT etc. as reported by Racle et al. (2017).

We also estimated proportions of PBMC cell types across all time points utilizing the same partial signature matrix (Figure 4c). On HD30, a decline of T cell abundances before Day 1 and an increase between Days 1 and 3 were observed, followed by a slight decline until Day 7. An increase of monocyte abundances before and a decrease after Day 1 were also seen. Both trends are consistent with the profile of a virus shedder of the H1N1 virus in Rahil et al. (2020) with a small time shift. The time shift could be attributed to differences in viral strains and strengths between Rahil et al. (2020) and Hoek et al. (2015), as the former is a study on real H1N1 patients. The lack of notable changes in the cell-type proportions of HD31 resembles more to the profile of a non-virus shedder. More information on the viral progression of these two subjects is needed to confirm these observations.

We wish to remark on the utility of adjusting for cell-type mRNA amounts in this case. It is known that normalization by TPM loses such information, which was corrected by ARTdeConv via $\text{diag}(\hat{\boldsymbol{s}})$. When re-running ARTdeConv on the same samples with $\text{diag}(\boldsymbol{s})$ forced to be the identity matrix, we observed less accurate estimates of proportions (Figure 4b), corroborating the need for the adjsuetment in this scenario.

3.3. ARTdeConv Reveals Changes in Key PBMC Cell Type Proportion in COVID-19 Patients

We performed an extensive deconvolution analysis on PBMC bulk samples gathered from 17 healthy controls and 14 patients with COVID-19 diagnosis recruited for a systemic immunity assessment against COVID-19 infections in humans by Arunachalam et al. (2020). The study also classified the COVID-19 severity of infected patients. Patients in the study were designated with three levels of severity: moderate, severe, and Intensity Care Unit-hospitalized (ICU). Details regarding the patient recruitment and the classification of COVID-19 severity are in Supplementary Material Section F.1.

Bulk RNA-seq data downloaded using NCBI GEO accession number GSE152418 were utilized to perform deconvolution for estimating the proportions of the four major PBMC cell types: T cell, B cell, natural killer (NK) cell, and monocyte. The authors also included single cell RNA-seq data on separate independent blood samples from five healthy controls and seven COVID-19 patients. Of these, all healthy and six COVID-19 infected subjects had matching bulk and single cell samples that passed quality control. The single cell data were downloaded using GEO accession GSE155673 and were used to construct the partial signature matrix. Complete descriptions of the bulk and single cell data, their quality control and pre-processing, as well as the creation of gene signature matrices can be found in Supplementary Material Section F.1.

Deconvolution was performed separately on samples from healthy controls and COVID-19 patients using bulk and signature matrices that matched the disease status. For each set of samples, we assumed knowledge on the reference expression of the four major PBMC cell types: T cell, B cell, NK cell, and monocyte. All other cell types' reference expression were assumed unkown and they were grouped into the super cell type "Others". Since obtaining the reference means and ranges of the cell types separately for healthy and diseased populations proved infeasible, we opted to use the same set of numbers in

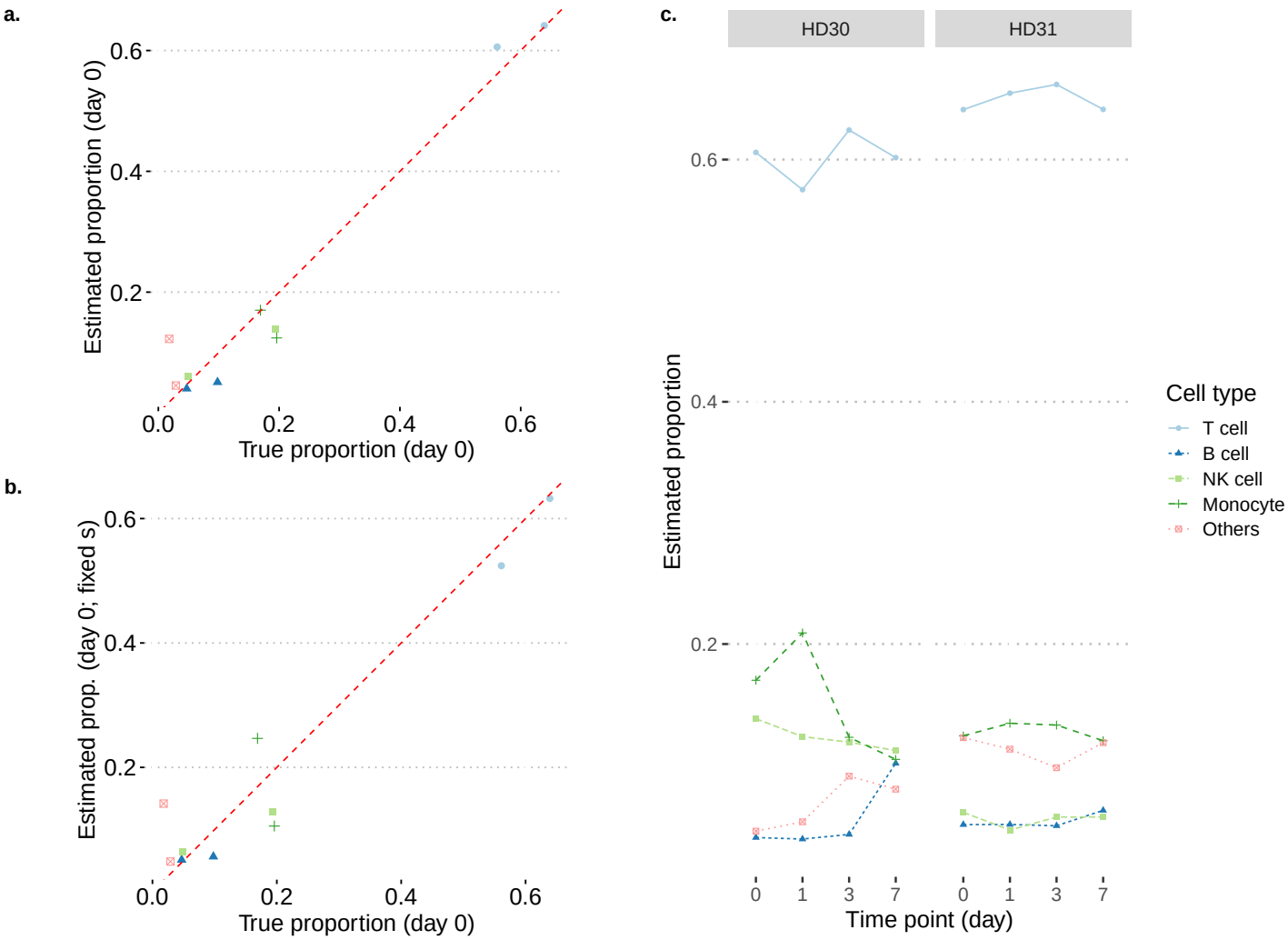

Figure 4: **a.** Estimated PBMC proportions by ARTdeConv versus true proportions measured by flow cytometry for two PBMC samples collected on Day 0 with flexible mRNA amount parameters. **b.** Estimated PBMC proportions by ARTdeConv versus true proportions measured by flow cytometry for two PBMC samples collected on Day 0 with mRNA amount parameters coerced to 1 for all cell types. **c.** Estimated PBMC proportions by ARTdeConv on all of the eight PBMC samples from two subjects across time points with flexible mRNA amount parameters.

Supplementary Table 1 for running ARTdeConv. Detailed descriptions of the tuning grid and other ARTdeConv parameters can be found in Supplementary Material Section F.2.

The complete results of deconvolved cell type proportions from ARTdeConv are shown in Figure 5a (boxplots) as well as Supplementary Figures 1 and 2 (bar charts). We observed that in COVID-19 samples, the proportions of T cells were notably lower than in the healthy control samples (Figure 5a). Among the diseased samples, T cell depletion were found in several severe and ICU samples (Figure 5b). This could be explained by T lymphopenia, commonly observed on blood samples of COVID-19 patients with severe symptoms but less frequently on those from patients with mild to moderate symptoms, as a result of the immunological responses of T cells to COVID-19 (Chen and John Wherry, 2020). It was also observed that Severe or ICU samples exhibited higher monocyte abundances in PBMC (Figure 5c). Similar trends have also been observed by Zhang et al. (2022) in blood samples of patients with severe COVID-19.

Arunachalam et al. (2020) used the abundances of cells from the single-cell samples as a surrogate measurement for true abundances (except for dendritic cells, which were manually enriched in the single-cell samples). The relative deconvolved abundances of T cells, B cells, NK cells, and monocytes were then compared against the relative abundances of single cells. ARTdeConv demonstrated satisfactory accuracy in deconvolving cell-type abundances (Figure 5d), achieving a CCC of 0.815 among healthy control samples (Pearson's correlation = 0.860, MAD = 0.098) and 0.694 among COVID-19 samples (Pearson's correlation = 0.717, MAD = 0.103).

While we observed a slightly lower accuracy among COVID-19 patients compared to healthy controls, the COVID-19 patients spanned several disease severity classes and various duration of infection. These factors might increase the variation of gene expression, both on the cell type and the bulk levels, making deconvolution more challenging. The analysis also identified one outlier in healthy controls, S066 (Supplementary Figure 2). Given the typical high abundances of T cells in human PBMC samples, it was surprising that ARTdeConv did not detect any T cell in this sample, while attributed monocytes as the most abundant cell type. After further explorations in the expression of *CD3* and *CD14* genes, which are experimentally confirmed marker genes of T cells and classical monocytes respectively (Kleiveland, 2015), we discovered exceptionally low bulk *CD3* expression (Supplementary Figure 3) and high *CD14* expression (Supplementary Figure 4) in the sample from S066, which were consistent with this result.

4. Discussion

In this paper, we introduce ARTdeConv, an innovative deconvolution approach. An important feature of ARTdeConv is its adoption of a tri-factor model, which integrates the cell-type mRNA amounts during the deconvolution process. As a semi-reference-based method, ARTdeConv offers enhanced flexibility compared to reference-based methods, as it accommodates cell types whose reference gene expression is not known by grouping them into one super cell type, while presenting advantages over reference-free methods by incorporating the partial signature matrix. Moreover, the method makes effective use of reference information on proportion means and ranges derived from external studies.

Additionally, we derive the MU algorithm for ARTdeConv and present a theorem that establishes the convergence of this algorithm to stationary points. This proof is derived by casting ARTdeConv's algorithm as a special case of the BSUM algorithm introduced

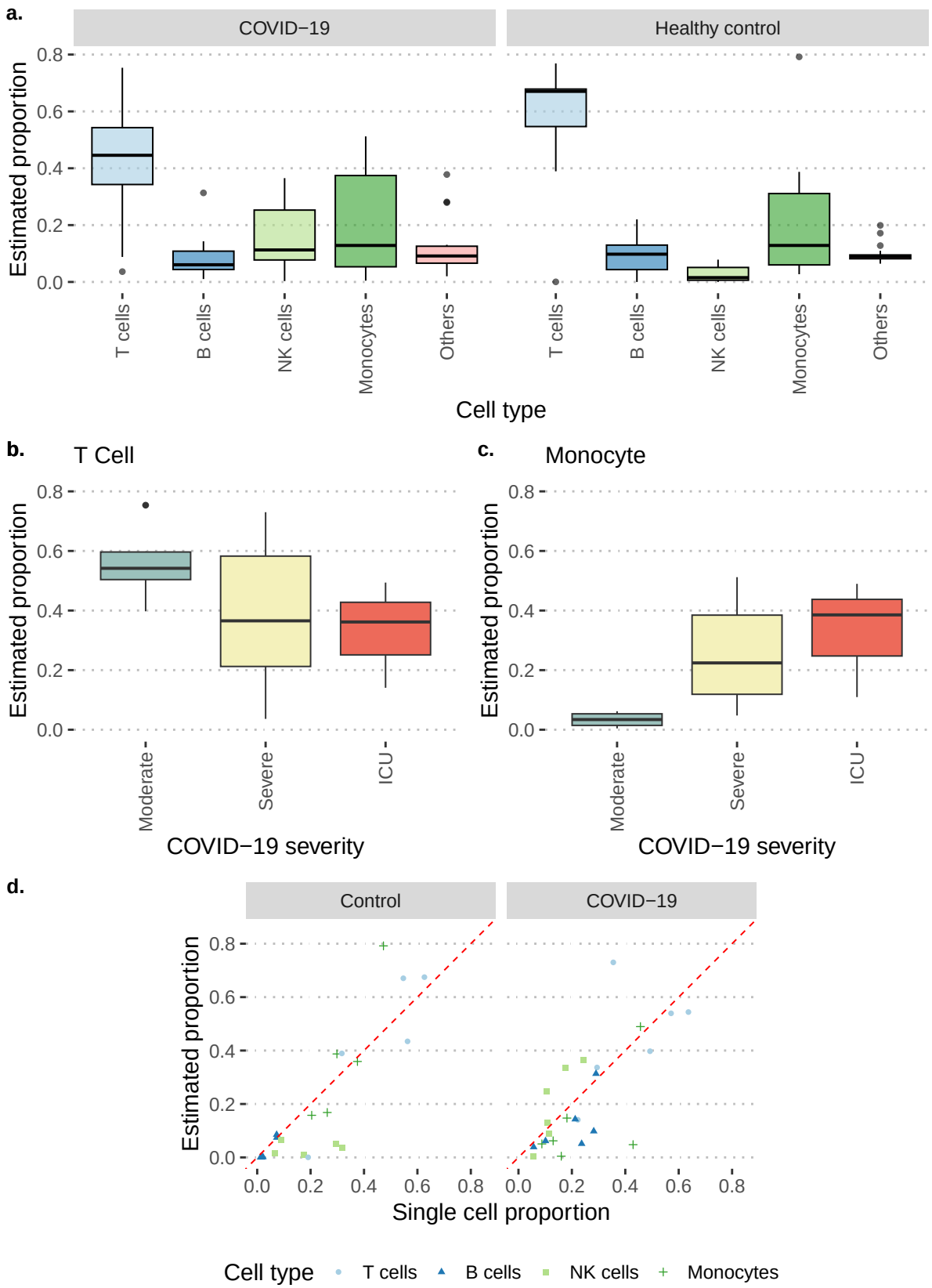

Figure 5: **a.** Box plots for estimated PBMC proportions by ARTdeConv in separate deconvolution analyses for healthy control and COVID-19-infected samples. **b.** and **c.** Box plots for estimated T cell and monocyte proportions on COVID-19-infected samples of different severity. **d.** Scatter plots for estimated PBMC proportions vs. matching tissue PBMC proportions from independent single-cell studies from five healthy controls and six COVID-19-diagnosed subjects.

by Razaviyayn et al. (2013).

On simulated pseudo-bulks, we demonstrated the advantages of ARTdeConv over other semi-reference-based methods. Notably, ARTdeConv performed better compared to EPIC when the cell types without reference expression became relatively abundant and had an overall edge against SECRET and other well-known reference-free methods. Moreover, both EPIC and SECRET require manual re-normalization of the estimated proportions by relatively precise cell-type mRNA amounts, specifications that are not required for ARTdeConv. In practice, such precise amounts can be difficult to obtain when the cell types with missing reference expression become numerous. Compared to quanTIseq and BayICE, ARTdeConv can flexibly utilize customized partial signature matrices, which are common in deconvolution applications. To the best of our knowledge, current semi-reference-based methods only take in one cell type or super cell type whose reference expression is unknown. A direction for future research in methodology is developing novel semi-reference-based methods that can distinguish multiple cell types with missing reference, i.e. truly allowing $K > K_0 + 1$.

An argument can be made that obtaining precise reference information on proportions demands additional efforts in practice. However, the advantages of integrating this information are substantial. Without it, the task of associating estimated proportions with their respective cell types can prove challenging. First, the estimated proportions of the cell types without reference expression could be biased towards zero, as in the case of semi-reference-based methods like EPIC. Secondly, for all reference-free methods, we had to perform manual matching of results (as demonstrated in Section 3.1), which becomes unfeasible if multiple cell types lack reference expression.

On the algorithmic side, ARTdeConv provides a guarantee of convergence, closing a gap in the theoretical characterization of deconvolution methods in previous studies. Of note, ARTdeConv does not assure that the estimated proportions are globally optimal. This challenge is not particular to ARTdeConv and is shared by other deconvolution methods with iterative optimization procedures such as SECRET, NMF, and debCAM. To counter this, we recommend employing marker genes for the deconvolution process and considering multiple restarts of ARTdeConv. While methods like debCAM mitigate this issue by using a different mathematical framework, their assumptions are usually breached in practice, and their performance diminishes as a result.

We demonstrated the application of ARTdeConv to two different sets of real data. While the data from Hoek et al. (2015) had limited sample size, we validated ARTdeConv's performance using the data from the more extensive study of Arunachalam et al. (2020). We remark that both sets of data offer an edge where both the bulk samples and the independent samples for making partial signature matrices were collected from independent samples concurrently, minimizing the influence of most technical artifacts. In situations where bulk and cell-type reference expression are derived from different studies, additional caution against those technical artifacts should be exercised during data pre-processing prior to deconvolution.

Software Availability

ARTdeConv is available as an R package for download from the GitHub repository: https://github.com/gr8lawrence/ARTDeConv.

Data Availability

Raw sequencing data in the fastq format for the analysis in Section 3.2 can be accessed through Sequence Read Archive (SRA) BioProject PRJNA271578. Measured proportions for PBMC cell types on Day 0 samples can be found in the supplementary materials of (Racle et al., 2017). Bulk and single cell data for the analysis in Section 3.3 could be accessed from NCBI GEO using the accession number GSE152418 and GSE155673 respectively.

Acknowledgements

The authors gratefully acknowledge the support of National Institute of Allergy and Infectious Diseases at the National Institutes of Health through the grant U19AI144181 and that of National Institute on Aging through the grant R01AG073259.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

S. Anders and W. Huber. Differential expression analysis for sequence count data. *Nature Precedings*, 11(10):R106, Oct. 2010.

P. S. Arunachalam, F. Wimmers, C. K. P. Mok, R. A. P. M. Perera, M. Scott, T. Hagan, N. Sigal, Y. Feng, L. Bristow, O. Tak-Yin Tsang, D. Wagh, J. Coller, K. L. Pellegrini, D. Kazmin, G. Alaaeddine, W. S. Leung, J. M. C. Chan, T. S. H. Chik, C. Y. C. Choi, C. Huerta, M. Paine McCullough, H. Lv, E. Anderson, S. Edupuganti, A. A. Upadhyay, S. E. Bosinger, H. T. Maecker, P. Khatri, N. Rouphael, M. Peiris, and B. Pulendran. Systems biological assessment of immunity to mild versus severe COVID-19 infection in humans. *Science*, 369(6508):1210–1220, Sept. 2020.

F. Avila Cobos, J. Alquicira-Hernandez, J. E. Powell, P. Mestdagh, and K. De Preter. Benchmarking of cell type deconvolution pipelines for transcriptomics data. *Nat. Commun.*, 11(1):1–14, Nov. 2020.

B. Chen, M. S. Khodadoust, C. L. Liu, A. M. Newman, and A. A. Alizadeh. Profiling tumor infiltrating immune cells with CIBERSORT. *Cancer Systems Biology: Methods and Protocols*, pages 243–259, 2018.

L. Chen, C.-T. Wu, N. Wang, D. M. Herrington, R. Clarke, and Y. Wang. debCAM: a bioconductor R package for fully unsupervised deconvolution of complex tissues. *Bioinformatics*, 36(12):3927–3929, 2020.

Z. Chen and E. John Wherry. T cell responses in patients with COVID-19. *Nat. Rev. Immunol.*, 20(9):529–536, Sept. 2020.

F. Elloumi, Z. Hu, Y. Li, J. S. Parker, M. L. Gulley, K. D. Amos, and M. A. Troester. Systematic bias in genomic classification due to contaminating non-neoplastic tissue in breast tumor samples. *BMC Med. Genomics*, 4(1):1–12, 2011.

F. Finotello, C. Mayer, C. Plattner, G. Laschober, D. Rieder, H. Hackl, A. Krogsdam, Z. Loncova, W. Posch, D. Wilflingseder, S. Sopper, M. Ijsselsteijn, T. P. Brouwer, D. Johnson, Y. Xu, Y. Wang, M. E. Sanders, M. V. Estrada, P. Ericsson-Gonzalez, P. Charoentong, J. Balko, N. F. d. C. C. de Miranda, and Z. Trajanoski. Molecular and pharmacological modulators of the tumor immune contexture revealed by deconvolution of RNA-seq data. *Genome Med.*, 11(1):34, May 2019.

R. Gaujoux and C. Seoighe. A flexible R package for nonnegative matrix factorization. *BMC Bioinformatics*, 11(1):1–9, 2010.

R. Gaujoux and C. Seoighe. CellMix: a comprehensive toolbox for gene expression deconvolution. *Bioinformatics*, 29(17):2211–2212, 2013.

C. Hafemeister and R. Satija. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol.*, 20(1): 1–15, 2019.

K. L. Hoek, P. Samir, L. M. Howard, X. Niu, N. Prasad, A. Galassie, Q. Liu, T. M. Allos, K. A. Floyd, Y. Guo, Y. Shyr, S. E. Levy, S. Joyce, K. M. Edwards, and A. J. Link. A cell-based systems biology assessment of human blood to monitor immune responses after influenza vaccination. *PLoS One*, 10(2):e0118528, Feb. 2015.

P. Huang, M. Cai, X. Lu, C. McKennan, and J. Wang. Accurate estimation of rare cell type fractions from tissue omics data via hierarchical deconvolution. *bioRxiv*, pages 2023–2003, Mar. 2023.

H. Jin and Z. Liu. A benchmark for RNA-seq deconvolution analysis under dynamic testing environments. *Genome Biol.*, 22:1–23, 2021.

J. Kim, Y. He, and H. Park. Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. *J. Global Optimiz.*, 58(2): 285–319, 2014.

C. R. Kleiveland. *Peripheral Blood Mononuclear Cells.* Springer, 2015.

K. Lange. *MM optimization algorithms.* SIAM, 2016.

D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Adv. Neural Inf. Process. Syst.*, 13, 2000.

Z. Li and H. Wu. TOAST: improving reference-free cell composition estimation by cross-cell type differential analysis. *Genome Biol.*, 20(1):1–17, Sept. 2019.

C.-J. Lin. On the convergence of multiplicative update algorithms for nonnegative matrix factorization. *IEEE Trans. Neural Netw.*, 18(6):1589–1596, 2007.

L. I. Lin. A concordance correlation coefficient to evaluate reproducibility. *Biometrics*, 45 (1):255–268, Mar. 1989.

Y. Lu, Q. M. Chen, and L. An. Semi-reference based cell type deconvolution with application to human metastatic cancers. *NAR Genom Bioinform*, 5(4):lqad109, Dec. 2023.

N. Lytal, D. Ran, and L. An. Normalization methods on single-cell RNA-seq data: an empirical survey. *Front. Genet.*, 11:41, 2020.

S. Mohammadi, N. Zuckerman, A. Goldsmith, and A. Grama. A critical survey of deconvolution methods for separating cell types in complex tissues. *Proc. IEEE*, 105(2): 340–366, Feb. 2017.

G. Monaco, B. Lee, W. Xu, S. Mustafah, Y. Y. Hwang, C. Carré, N. Burdin, L. Visan, M. Ceccarelli, M. Poidinger, A. Zippelius, J. Pedro de Magalhães, and A. Larbi. RNA-seq signatures normalized by mRNA abundance allow absolute deconvolution of human immune cell types. *Cell Rep.*, 26(6):1627–1640.e7, Feb. 2019.

J. S. Morris and S. Kopetz. Tumor microenvironment in gene signatures: critical biology or confounding noise? *Clin. Cancer Res.*, 22(16):3989–3991, 2016.

A. M. Newman, C. L. Liu, M. R. Green, A. J. Gentles, W. Feng, Y. Xu, C. D. Hoang, M. Diehn, and A. A. Alizadeh. Robust enumeration of cell subsets from tissue expression profiles. *Nat. Methods*, 12(5):453–457, May 2015.

Y. Qiu, J. Wang, J. Lei, and K. Roeder. Identification of cell-type-specific marker genes from co-expression patterns in tissue samples. *Bioinformatics*, 37(19):3228–3234, 2021.

J. Racle and D. Gfeller. EPIC: A tool to estimate the proportions of different cell types from bulk gene expression data. *Methods Mol. Biol.*, 2120:233–248, 2020.

J. Racle, K. de Jonge, P. Baumgaertner, D. E. Speiser, and D. Gfeller. Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. *Elife*, 6:e26476, Nov. 2017.

Z. Rahil, R. Leylek, C. M. Schürch, H. Chen, Z. Bjornson-Hooper, S. R. Christensen, P. F. Gherardini, S. S. Bhate, M. H. Spitzer, G. K. Fragiadakis, and Others. Landscape of coordinated immune responses to H1N1 challenge in humans. *J. Clin. Invest.*, 130(11): 5800–5816, 2020.

M. Razaviyayn, M. Hong, and Z.-Q. Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM J. Optim.*, 23(2):1126–1153, 2013.

M. H. Spitzer and G. P. Nolan. Mass cytometry: single cells, many features. *Cell*, 165(4): 780–791, 2016.

A.-S. Tai, G. C. Tseng, and W.-P. Hsieh. BayICE: A bayesian hierarchical model for semireference-based deconvolution of bulk transcriptomic data. *aoas*, 15(1):391–411, Mar. 2021.

A. Takami, S. Watanabe, Y. Yamamoto, H. Miyachi, Y. Bamba, M. Ohata, S. Mishima, H. Kubota, A. Nishiura, T. Inaba, M. Enomoto, T. Mitsuhashi, K. Nakanishi, R. Miura, E. Nonaka, K. Shimbo, Y. Yatomi, K. Tohyama, and Japanese Society for Laboratory

Hematology subcommittee on Standardization of Blood Cell Morphology (JSLH-SBCM). Reference intervals of white blood cell parameters for healthy adults in japan. *Int. J. Lab. Hematol.*, 43(5):948–958, Oct. 2021.

N. Wang, E. P. Hoffman, L. Chen, L. Chen, Z. Zhang, C. Liu, G. Yu, D. M. Herrington, R. Clarke, and Y. Wang. Mathematical modelling of transcriptional heterogeneity identifies novel markers and subpopulations in complex tissues. *Sci. Rep.*, 6(1):18909, 2016.

X. Wang, J. Park, K. Susztak, N. R. Zhang, and M. Li. Bulk tissue cell type deconvolution with multi-subject single-cell expression reference. *Nat. Commun.*, 10(1):1–9, Jan. 2019.

G. Wigerblad, Q. Cao, S. Brooks, F. Naz, M. Gadkari, K. Jiang, S. Gupta, L. O'Neil, S. Dell'Orso, M. J. Kaplan, and Others. Single-cell analysis reveals the range of transcriptional states of circulating human neutrophils. *bioRxiv*, 2022.

K. Zaitsev, M. Bambouskova, A. Swain, and M. N. Artyomov. Complete deconvolution of cellular mixtures based on linearity of transcriptional signatures. *Nat. Commun.*, 10(1): 1–16, May 2019.

Y. Zhang, S. Wang, H. Xia, J. Guo, K. He, C. Huang, R. Luo, Y. Chen, K. Xu, H. Gao, J. Sheng, and L. Li. Identification of monocytes associated with severe COVID-19 in the PBMCs of severely infected patients through Single-Cell transcriptome sequencing. *Proc. Est. Acad. Sci. Eng.*, 17:161–169, Oct. 2022.

Supplementary Material

A. Technical Assumptions

Before presenting the main results, we state the technical assumptions.

Assumption 1 (Initial conditions). *(i)* $\forall k$, $\exists j_k \in \{1, \ldots, m\}$ *such that* $\theta^0_{j_k k} > 0$*; (ii)* $\forall k$, $\exists i_k \in \{1, \ldots, n\}$ *such that* $p^0_{ki_k} > 0$*. (iii)* $\forall i' \in \{i_1, i_2, \ldots, i_K\}$, $\forall j$, $Y_{ji'} > 0$*;* $\forall j' \in \{j_1, j_2, \ldots, j_K\}$, $\forall i$, $Y_{j'i} > 0$.

Assumption 2 (Boundedness). *(i) For any* $1 \leq j \leq m$ *and* $1 \leq i \leq n$, $Y_{ji} \leq M_Y < \infty$*. (ii) For any* $1 \leq j \leq m$ *and* $1 \leq k \leq K_0$, $\Theta_{0_{jk}} \leq M_\Theta < \infty$.

These assumptions are reasonable in practice. For every cell type with established reference expression, it is safe to assume that there exist some genes that display non-zero cell-type-specific expression, thereby fulfilling Assumption 1(*i*). For cell types lacking reference expression, an initial value in $\boldsymbol{\Theta}^0$ can be straightforwardly set to a positive value, thereby also satisfying the same assumption. Assumption 1(*ii*) is automatically satisfied due to how $\boldsymbol{P}^0$ is initialized in Algorithm 1. Assumption 1(*iii*) plays a pivotal role in ensuring that positive initialized values remain positive throughout the iterative process (see Supplementary Material Section B for details). This assumption is easy to verify and holds true unless the bulk expression matrix is excessively sparse, which is uncommon in practice. Assumption 2 is a mild assumption on upper bounds of gene expression.

B. The MU Steps

B.1. A Preliminary Lemma

Similar to Lee and Seung (2000), $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ is minimized using block-wise auxiliary functions that are quadratic over-estimators, which then leads to the MU steps in (4), (5), and (6). The claims that these updates will keep the positive initialized values positive will be investigated at the end of this section. An auxiliary function is defined next.

Definition 1 (Auxiliary function). *Given a vector space* V *and* $\boldsymbol{v}, \boldsymbol{v}' \in V$*, a function* $g(\boldsymbol{v}|\boldsymbol{v}') : V \times V \mapsto \mathbb{R}$ *is an auxiliary function for the function* $f(\boldsymbol{v}) : V \mapsto \mathbb{R}$ *at* $\boldsymbol{v}'$ *if the following conditions are satisfied:*

$$g(\boldsymbol{v}|\boldsymbol{v}') \geq f(\boldsymbol{v}'), \forall \boldsymbol{v} \in V, \text{ and } g(\boldsymbol{v}'|\boldsymbol{v}') = f(\boldsymbol{v}').$$

As we shall see, the vector blocks in the auxiliary functions would be each row of $\boldsymbol{\Theta}$, each column of $\boldsymbol{P}$, and $\boldsymbol{s}$. To find the specific auxiliary functions for $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$, a slightly modified form of a technical lemma given by Lin (2007) is needed and stated next. Its proof is also given in the same paper and thus omitted.

Lemma 1. *Given a positive semi-definite matrix* $\boldsymbol{Q} \in \mathbb{R}^{b \times b}_+$ *and* $\boldsymbol{w} \in \mathbb{R}^b_+$*, let* $I \subseteq \{1, \ldots, b\}$ *be a set of indices such that* $w_k > 0$ *for any* $k \in I$*,* $\boldsymbol{Q}_{II} \in \mathbb{R}^{|I| \times |I|}_{++}$*, and* $\boldsymbol{w}_I \in \mathbb{R}^{|I|}_{++}$ *be the sub-matrix of* $\boldsymbol{Q}$ *and the sub-vector of* $\boldsymbol{w}$ *with indices in* I*, where* $|I|$ *denotes the cardinality*

of I. Then,

$$diag\left\{\frac{(\boldsymbol{Q}_{II}\boldsymbol{w}_I)}{\boldsymbol{w}_I}\right\} - \boldsymbol{Q}_{II} = \begin{bmatrix} \frac{(\boldsymbol{Q}_{II}\boldsymbol{w}_I)_1}{w_{I1}} & & \\ & \ddots & \\ & & \frac{(\boldsymbol{Q}_{II}\boldsymbol{w}_I)_{|I|}}{w_{I|I|}} \end{bmatrix} - \boldsymbol{Q}_{II}$$

is always positive semi-definite.

With Lemma 1, the auxiliary functions and multiplicative update (MU) steps are ready to be derived.

B.2. Derivations of The MU Steps

In addition to those in Section 2.1, we introduce two additional notation: for two square matrices of the same dimension, we write $\boldsymbol{A} \succ \boldsymbol{B}$ and $\boldsymbol{A} \succeq \boldsymbol{B}$ if $\boldsymbol{A} - \boldsymbol{B}$ is positive-definite and positive-semidefinite, respectively.

The auxiliary function and update step for each row $\boldsymbol{\theta}_j^\top$ is derived and then combined for $j = 1, \ldots, m$ into the form of (4). For simplicity in notations, the index j is omitted in subscripts. The iteration number t is also omitted for $\boldsymbol{s}$ and $\boldsymbol{P}$ since both are treated as fixed. That is, we denote $\boldsymbol{y}_j^\top$ as $\boldsymbol{y}^\top$, $\boldsymbol{\theta}_j^\top$ as $\boldsymbol{\theta}^\top$, and $\boldsymbol{\delta}_j^\top$ as $\boldsymbol{\delta}$. We also define $\boldsymbol{\delta}_j^c = \boldsymbol{J} - \boldsymbol{\delta}_j$ for all j. Thus, for the j-th row of $\boldsymbol{\Theta}$, we can define the sub-problem of $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ as

$$\underset{\boldsymbol{\theta}\in\mathbb{R}_+^K}{\operatorname{argmin}} \frac{1}{2mn}\|\boldsymbol{y}^\top - \boldsymbol{\theta}^\top \operatorname{diag}(\boldsymbol{s})\boldsymbol{P}\|_2^2 + \frac{\alpha_1}{2}\|\boldsymbol{\delta}^\top \odot (\boldsymbol{\theta}^\top - \boldsymbol{\theta}_0^\top)\|_2^2 + \frac{\alpha_2}{2}\|\boldsymbol{\delta}^{c\top} \odot \boldsymbol{\theta}^\top\|_2^2 + C_1(\boldsymbol{s}, \boldsymbol{P}) \tag{S.8}$$

$$= \underset{\boldsymbol{\theta}\in\mathbb{R}_+^K}{\operatorname{argmin}} \frac{1}{2mn}\|\boldsymbol{y}^\top - \boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s})\boldsymbol{\theta}\|_2^2 + \frac{\alpha_1}{2}\|\boldsymbol{\delta} \odot (\boldsymbol{\theta} - \boldsymbol{\theta}_0)\|_2^2 + \frac{\alpha_2}{2}\|\boldsymbol{\delta}^c \odot \boldsymbol{\theta}\|_2^2 + C_1(\boldsymbol{s}, \boldsymbol{P}), \tag{S.9}$$

where $\operatorname{diag}(\boldsymbol{s})$ and $\boldsymbol{P}$ are treated as fixed and $C_1(\boldsymbol{s}, \boldsymbol{P})$ is a constant with respect to $\boldsymbol{\theta}$. We call the function in (S.9) $f_\theta(\boldsymbol{\theta})$. (S.9) is derived from (S.8) by transposing. Now, to find the auxiliary function $h_\theta(\boldsymbol{\theta}|\boldsymbol{\theta}^t)$, by performing a second-order Taylor series expansion on $f_\theta(\boldsymbol{\theta})$ at $\boldsymbol{\theta} = \boldsymbol{\theta}^t$, we have

$$f_\theta(\boldsymbol{\theta}) = f_\theta(\boldsymbol{\theta}^t) + \nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t)^\top(\boldsymbol{\theta} - \boldsymbol{\theta}^t) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^t)^\top \nabla_{\boldsymbol{\theta}}^2 f_\theta(\boldsymbol{\theta}^t)(\boldsymbol{\theta} - \boldsymbol{\theta}^t).$$

Next, letting $\boldsymbol{V}(\boldsymbol{\delta}) = \operatorname{diag}(\boldsymbol{\delta})$ and $\boldsymbol{V}^c(\boldsymbol{\delta}^c) = \operatorname{diag}(\boldsymbol{\delta}^c)$, it can be observed that $\boldsymbol{\delta} \odot \boldsymbol{\theta} = \boldsymbol{V}\boldsymbol{\theta}$ and $\boldsymbol{\delta}^c \odot \boldsymbol{\theta} = \boldsymbol{V}^c\boldsymbol{\theta}$. Also, $\boldsymbol{V} + \boldsymbol{V}^c = \boldsymbol{I}$. The first and second derivatives of $f_\theta(\boldsymbol{\theta})$ are thus:

$$\nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}) = \frac{1}{mn}\operatorname{diag}(\boldsymbol{s})\boldsymbol{P}(\boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s})\boldsymbol{\theta} - \boldsymbol{y}^\top) + \alpha_1 \boldsymbol{V}(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \alpha_2 \boldsymbol{V}^c\boldsymbol{\theta},$$

and

$$\nabla_{\boldsymbol{\theta}}^2 f_\theta(\boldsymbol{\theta}) = \frac{1}{mn}\operatorname{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s}) + \alpha_1 \boldsymbol{V} + \alpha_2 \boldsymbol{V}^c.$$

It can be guaranteed $\nabla_{\boldsymbol{\theta}}^2 f_\theta(\boldsymbol{\theta}) \succ \boldsymbol{0}$ given $\alpha_1, \alpha_2 > 0$, as $\nabla_{\boldsymbol{\theta}}^2 f_\theta(\boldsymbol{\theta}) \succeq \alpha_1 \boldsymbol{V} + \alpha_2 \boldsymbol{V}^c \succeq \min(\alpha_1, \alpha_2)\boldsymbol{I} \succ \boldsymbol{0}$. Then, define $\boldsymbol{d}(\boldsymbol{\theta}^t) = \operatorname{diag}(\boldsymbol{\theta}^t)^{-1}\nabla_{\boldsymbol{\theta}}^2 f_\theta(\boldsymbol{\theta}^t)\boldsymbol{\theta}^t$, and $\boldsymbol{D}(\boldsymbol{\theta}^t) = \operatorname{diag}(\boldsymbol{d}(\boldsymbol{\theta}^t))$. Lemma 1 implies that, on coordinates that correspond to the positive elements of $\boldsymbol{\theta}$,

$\boldsymbol{D}(\boldsymbol{\theta}^t) - \nabla^2_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t) \succeq 0$. Define $\mathbb{R}^{K_1}_{++}$, where $K_1 \leq K$, as the open half space that contains this positive subset of $\boldsymbol{\theta}$. For simplification, we use the same notations for the full vectors and matrices on those positive coordinates. As mentioned in Section 2.2, we shall see that once a coordinate of $\boldsymbol{\theta}$ reaches zero, it stays zero thereafter, which justifies the focus on these positive coordinates. Next, define

$$h_\theta(\boldsymbol{\theta}|\boldsymbol{\theta}^t) = f_\theta(\boldsymbol{\theta}^t) + \nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t)^\top(\boldsymbol{\theta} - \boldsymbol{\theta}^t) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^t)^\top \boldsymbol{D}(\boldsymbol{\theta}^t)(\boldsymbol{\theta} - \boldsymbol{\theta}^t), \tag{S.10}$$

which is an auxiliary function of (S.9) (thus of $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ when we combine all subproblems together for $j = 1, \ldots m$). To demonstrate that it satisfies Definition 1, simply see that

$$h_\theta(\boldsymbol{\theta}|\boldsymbol{\theta}^t) - f_\theta(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^t)^\top(\boldsymbol{D}(\boldsymbol{\theta}^t) - \nabla^2_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t))(\boldsymbol{\theta} - \boldsymbol{\theta}^t) \geq 0, \ \forall \boldsymbol{\theta}, \boldsymbol{\theta}^t \in \mathbb{R}^{K_1}_{++}$$

and due to Lemma 1 and that $h_\theta(\boldsymbol{\theta}^t|\boldsymbol{\theta}^t) - f_\theta(\boldsymbol{\theta}^t) = 0, \ \forall \boldsymbol{\theta}^t \in \mathbb{R}^{K_1}_{++}$. To derive the MU step from this auxiliary function, we find $\boldsymbol{\theta}^{t+1}$ such that

$$\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta} \in \mathbb{R}^{K_1}_{+}}{\operatorname{argmin}} \ h_\theta(\boldsymbol{\theta}|\boldsymbol{\theta}^t). \tag{S.11}$$

To guarantee the existence of a solution for (S.11), the feasible set of $\boldsymbol{\theta}$ is a closed space that includes 0. Since $\boldsymbol{D}(\boldsymbol{\theta}^t) - \nabla^2_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t) \succeq 0$ and $\nabla^2_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t) \succ 0$ by the assumptions, $\boldsymbol{D}(\boldsymbol{\theta}^t) \succ 0$, which implies that (S.10) is strictly convex. Thus, (S.10) has a global minimizer in the closed half space $\mathbb{R}^{K_1}_{+}$. By the optimality condition, if a feasible $\tilde{\boldsymbol{\theta}} \in \mathbb{R}^{K_1}_{+}$ satisfies $\nabla_{\boldsymbol{\theta}} h_\theta(\tilde{\boldsymbol{\theta}}|\boldsymbol{\theta}^t) = 0$, then $\tilde{\boldsymbol{\theta}}$ minimizes (S.10) globally and $\boldsymbol{\theta}^{t+1} = \tilde{\boldsymbol{\theta}}$ becasue of (S.11). Thus, we solve

$$\nabla_{\boldsymbol{\theta}} h_\theta(\boldsymbol{\theta}^{t+1}|\boldsymbol{\theta}^t) = \nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t) + \boldsymbol{D}(\boldsymbol{\theta}^t)(\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^t) = 0. \tag{S.12}$$

Because $\boldsymbol{D}(\boldsymbol{\theta}^t)$ is diagonal, for each $k \in 1, \ldots, K_1$, (S.12) implies

$$\theta_k^{t+1} = \theta_k^t - \frac{\nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t)_k}{D(\boldsymbol{\theta}^t)_k}. \tag{S.13}$$

We further simplify the notations by letting

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}) &= \frac{1}{mn}\operatorname{diag}(\boldsymbol{s})\boldsymbol{P}(\boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s})\boldsymbol{\theta} - \boldsymbol{y}^\top) + \alpha_1 \boldsymbol{V}(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \alpha_2 \boldsymbol{V}^c \boldsymbol{\theta} \\
&= \left(\frac{1}{mn}\operatorname{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s}) + \alpha_1 \boldsymbol{V} + \alpha_2 \boldsymbol{V}^c\right)\boldsymbol{\theta} \\
&- \left(\frac{1}{mn}\operatorname{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top \operatorname{diag}(\boldsymbol{s})\boldsymbol{y}^\top + \alpha_1 \boldsymbol{V}\boldsymbol{\theta}_0\right) \\
&= \boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b}.
\end{aligned}$$

It follows immediately that $\nabla^2_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}) = \boldsymbol{A}$. Plugging in the values of $\nabla_{\boldsymbol{\theta}} f_\theta(\boldsymbol{\theta}^t)_k$ and $D(\boldsymbol{\theta}^t)_k$

to (S.13), we have

$$
\begin{aligned}
\theta_k^{t+1} &= \theta_k^t - \frac{(\boldsymbol{A}\boldsymbol{\theta}^t - \boldsymbol{b})_k}{\frac{1}{\theta_k^t}(\boldsymbol{A}\boldsymbol{\theta}^t)_k} \\
&= \theta_k^t - \frac{(\boldsymbol{A}\boldsymbol{\theta}^t)_k}{(\boldsymbol{A}\boldsymbol{\theta}^t)_k}\theta_k^t + \frac{b_k}{(\boldsymbol{A}\boldsymbol{\theta}^t)_k}\theta_k^t \\
&= \frac{b_k}{(\boldsymbol{A}\boldsymbol{\theta}^t)_k}\theta_k^t \\
&= \frac{\left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{y}^\top + \alpha_1\boldsymbol{V}\boldsymbol{\theta}_0\right)_k}{\left[\left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top\mathrm{diag}(\boldsymbol{s}) + \alpha_1\boldsymbol{V} + \alpha_2\boldsymbol{V}^c\right)\boldsymbol{\theta}^t\right]_k}\theta_k^t \\
&= \frac{\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{y}^\top + mn\alpha_1\boldsymbol{V}\boldsymbol{\theta}_0\right)_k}{\left[\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top\mathrm{diag}(\boldsymbol{s}) + mn(\alpha_1\boldsymbol{V} + \alpha_2\boldsymbol{V}^c)\right)\boldsymbol{\theta}^t\right]_k}\theta_k^t. \qquad \text{(S.14)}
\end{aligned}
$$

The multiplicative form of (S.14) guarantees that $\boldsymbol{\theta}^{t+1}$ is feasible. Therefore, $\boldsymbol{\theta}^{t+1}$ satisfies (S.11).

It is straightforward to check that (S.14) also applies to zero coordinates of $\boldsymbol{\theta}^t$ ($\{\theta_k^t : \theta_k^t \in \mathbb{R}_+^K \setminus \mathbb{R}_{++}^{K_1}\}$), as zero coordinates remain zero after any multiplication. Therefore, it is the unified update step for the entire $\boldsymbol{\theta}^t \in \mathbb{R}_+^K$. Since it corresponds to the update for the (j,k)-th element in $\boldsymbol{\Theta}$, combining all elements gives (4).

We remark that (S.13) reflects the block coordinate descent nature of the multiplicative update in $\boldsymbol{\Theta}$, as observed by Kim et al. (2014). In fact, the update steps for $\boldsymbol{P}$ and $\mathrm{diag}(\boldsymbol{s})$ can be written in similar fashions.

The update step for $\boldsymbol{P}$ in (5) can be similarly derived as that for $\boldsymbol{\Theta}$. We begin with specifying the sub-problem for $\boldsymbol{p}_i$, $i = 1, \ldots, n$, treating $\boldsymbol{\Theta}$ and $\mathrm{diag}(\boldsymbol{s})$ as fixed and omitting the subscripts and the iteration number (t for $\boldsymbol{s}$ and $t+1$ for $\boldsymbol{\Theta}$) for convenience in notations:

$$
\underset{\boldsymbol{p}\in\mathbb{R}_+^K}{\mathrm{argmin}}\ f_p(\boldsymbol{p}) = \frac{1}{2mn}\|\boldsymbol{y} - \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{p}\|_2^2 + \frac{\beta}{2}\boldsymbol{\rho}^{-1}\|\boldsymbol{p} - \boldsymbol{m}\|_2^2 + K_2(\boldsymbol{\Theta}, \boldsymbol{s}), \qquad \text{(S.15)}
$$

where $K_2(\boldsymbol{\Theta}, \boldsymbol{s})$ only depends on $\boldsymbol{\Theta}$ and $\boldsymbol{s}$ and is a constant with respect to $\boldsymbol{P}$. Then, its first and second derivatives with respect to $\boldsymbol{p}$ are

$$
\begin{aligned}
\nabla_{\boldsymbol{p}} f_p(\boldsymbol{p}) &= \left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s}) + \beta\boldsymbol{\rho}^{-1}\right)\boldsymbol{p} - \left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{y} + \beta\boldsymbol{\rho}^{-1}\boldsymbol{m}\right) \\
&= \boldsymbol{C}\boldsymbol{p} - \boldsymbol{q}, \qquad \text{(S.16)}
\end{aligned}
$$

and

$$
\nabla_{\boldsymbol{p}}^2 f_p(\boldsymbol{p}) = \frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s}) + \beta\boldsymbol{\rho}^{-1} = \boldsymbol{C}. \qquad \text{(S.17)}
$$

To find the auxiliary function of (S.15), define $\boldsymbol{d}(\boldsymbol{p}^t) = \mathrm{diag}(\boldsymbol{p}^t)^{-1}\nabla_{\boldsymbol{p}}^2 f_p(\boldsymbol{p}^t)\boldsymbol{p}^t$, and $\boldsymbol{D}(\boldsymbol{p}^t) = \mathrm{diag}(\boldsymbol{d}(\boldsymbol{p}^t))$, and

$$
h_p(\boldsymbol{p}|\boldsymbol{p}^t) = f_p(\boldsymbol{p}^t) + \nabla_{\boldsymbol{p}} f_p(\boldsymbol{p}^t)^\top(\boldsymbol{p} - \boldsymbol{p}^t) + \frac{1}{2}(\boldsymbol{p} - \boldsymbol{p}^t)^\top\boldsymbol{D}(\boldsymbol{p}^t)(\boldsymbol{p} - \boldsymbol{p}^t). \qquad \text{(S.18)}
$$

On the $K_2 \le K$ positive coordinates in $\boldsymbol{p}^t$, (S.18) is the auxiliary function of (S.15) due to Lemma 1. The verification is the same to that for $h_\theta(\boldsymbol{\theta}|\boldsymbol{\theta}^t)$ and omitted here. Next, we find $\boldsymbol{p}^{t+1}$ such that

$$
\boldsymbol{p}^{t+1} = \underset{\boldsymbol{p}\in\mathbb{R}_+^{K_2}}{\mathrm{argmin}}\ h_p(\boldsymbol{p}|\boldsymbol{p}^t).
$$

This in turn corresponds to finding $\boldsymbol{p}^{t+1}$ such that

$$\nabla_{\boldsymbol{p}} h_p(\boldsymbol{p}^{t+1}|\boldsymbol{p}^t) = \nabla_{\boldsymbol{p}} f_p(\boldsymbol{p}^t) + \boldsymbol{D}(\boldsymbol{p}^t)(\boldsymbol{p}^{t+1} - \boldsymbol{p}^t) = 0,$$

given that such a $\boldsymbol{p}^{t+1}$ is feasible. After some algebra, for any $k = 1, \ldots, K_2$,

$$\begin{aligned} p_k^{t+1} &= \frac{q_k}{(\boldsymbol{C}\boldsymbol{p}^t)_k} p_k^t \\ &= \frac{\left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top \boldsymbol{y} + \beta \boldsymbol{\rho}^{-1}\boldsymbol{m}\right)_k}{\left\{\left(\frac{1}{mn}\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s}) + \beta\boldsymbol{\rho}^{-1}\right)\boldsymbol{p}\right\}_k} p_k^t \\ &= \frac{\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top \boldsymbol{y} + mn\beta \boldsymbol{\rho}^{-1}\boldsymbol{m}\right)_k}{\left\{\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s}) + mn\beta\boldsymbol{\rho}^{-1}\right)\boldsymbol{p}\right\}_k} p_k^t, \end{aligned} \tag{S.19}$$

which produces a feasible $\boldsymbol{p}^{t+1}$. Moreover, (S.19) also applies to the zero coordinates in $\boldsymbol{p}^t$. Since (S.19) corresponds to the update step for the (k, i)-th element for the $\boldsymbol{P}$ matrix, combining all of these elements yields (5).

Deriving the updates step for $\mathrm{diag}(\boldsymbol{s})$ in (6) largely follows the same footsteps as for $\boldsymbol{\Theta}$ and $\boldsymbol{P}$. To find a suitable sub-problem, we start from the following identity

$$\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P} = \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P} = \sum_{k=1}^{K} s_k \boldsymbol{\theta}_k \boldsymbol{p}_k^\top. \tag{S.20}$$

Denoting $\boldsymbol{G}_k = \boldsymbol{\theta}_k \boldsymbol{p}_k^\top$ and only assuming $\boldsymbol{s}$ a variable, we transform the objective function in $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ with respect to $\boldsymbol{s}$ as follows

$$\begin{aligned} f_s(\boldsymbol{s}) &= \frac{1}{2mn}\|\boldsymbol{Y} - \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\|_F^2 + C_3(\boldsymbol{\Theta}, \boldsymbol{P}) && \text{(by rewriting } f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})) \\ &= \frac{1}{2mn}\|\boldsymbol{Y} - \sum_{k=1}^{K} s_k \boldsymbol{G}_k\|_F^2 + C_3(\boldsymbol{\Theta}, \boldsymbol{P}) && \text{(by (S.20))} \\ &\propto \frac{1}{2}\left\{\sum_{k=1}^{K} s_k^2 \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{G}_k\right) - 2\sum_{k=1}^{K} s_k \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{Y}\right) + \sum_{k \neq l} s_k s_l \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{G}_l\right)\right\} + C_3'(\boldsymbol{\Theta}, \boldsymbol{P}) \\ &\qquad \text{(by expanding the squared Frobenius norm and dropping } m, n) \\ &= \frac{1}{2}\left\{\sum_{k=1}^{K} s_k^2 \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{G}_k\right) + \sum_{k \neq l} s_k s_l \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{G}_l\right)\right\} - \sum_{k=1}^{K} s_k \mathrm{tr}\left(\boldsymbol{G}_k^\top \boldsymbol{Y}\right) + C_3'(\boldsymbol{\Theta}, \boldsymbol{P}) \\ &\qquad \text{(by rearranging the last step)} \\ &= \frac{1}{2}\boldsymbol{s}^\top \boldsymbol{Z}\boldsymbol{s} - \boldsymbol{u}^\top \boldsymbol{s} + C_3'(\boldsymbol{\Theta}, \boldsymbol{P}), \end{aligned} \tag{S.21}$$

where $\boldsymbol{Z} \in \mathbb{R}_+^{K \times K}$, $Z_{ij} = \mathrm{tr}\left(\boldsymbol{G}_i^\top \boldsymbol{G}_j\right) = \mathrm{tr}\left(\boldsymbol{p}_i \boldsymbol{\theta}_i^\top \boldsymbol{\theta}_j \boldsymbol{p}_j^\top\right)$, $\boldsymbol{u} \in \mathbb{R}_+^K$, and $u_k = \mathrm{tr}\left(\boldsymbol{Y}^\top \boldsymbol{G}_k\right) = \mathrm{tr}\left(\boldsymbol{Y}^\top \boldsymbol{\theta}_k \boldsymbol{p}_k^\top\right)$. Both C_3 and C_3' only depend on $\boldsymbol{\Theta}$ and $\boldsymbol{P}$, hence are constant with respect to $\boldsymbol{s}$. (S.21) is then our sub-problem for $\boldsymbol{s}$. From (S.21), the first and second derivatives for $\boldsymbol{s}$ are $\nabla_{\boldsymbol{s}} f_s(\boldsymbol{s}) = \boldsymbol{Z}\boldsymbol{s} - \boldsymbol{u}$, and $\nabla_{\boldsymbol{s}}^2 f_s(\boldsymbol{s}) = \boldsymbol{Z}$. Again, define $\boldsymbol{d}(\boldsymbol{s}^t) = \mathrm{diag}(\boldsymbol{s}^t)^{-1}\nabla_{\boldsymbol{s}}^2 f_s(\boldsymbol{s}^t)\boldsymbol{s}^t$, $\boldsymbol{D}(\boldsymbol{s}^t) = \mathrm{diag}(\boldsymbol{d}(\boldsymbol{s}^t))$, and

$$h_s(\boldsymbol{s}|\boldsymbol{s}^t) = f_s(\boldsymbol{s}^t) + \nabla_{\boldsymbol{s}} f_s(\boldsymbol{s}^t)^\top(\boldsymbol{s} - \boldsymbol{s}^t) + \frac{1}{2}(\boldsymbol{s} - \boldsymbol{s}^t)^\top \boldsymbol{D}(\boldsymbol{s}^t)(\boldsymbol{s} - \boldsymbol{s}^t). \tag{S.22}$$

(S.22) is an auxiliary function of $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ for $\boldsymbol{s}$ according to Lemma 1. Since $\boldsymbol{s} \in \mathbb{R}^K_{++}$, there is no need for sub-setting to positive coordinates. Thus, we obtain $\boldsymbol{s}^{t+1}$ by finding a feasible

$$\boldsymbol{s}^{t+1} = \underset{\boldsymbol{s} \in \mathbb{R}^K_+}{\operatorname{argmin}}\ h_s(\boldsymbol{s}|\boldsymbol{s}^t). \tag{S.23}$$

We remark that this is a slight relaxation of the problem to update $\boldsymbol{s}$, since it is originally assumed that $\boldsymbol{s} \in \mathbb{R}^K_{++}$. However, this allows us to obtain a guaranteed closed form global minimizer for (S.23) in the closed half space $\mathbb{R}^K_+$, and the resulting update satisfies is feasible in $\mathbb{R}^K_{++}$ as long as not all values of $\boldsymbol{\theta}_k$ or $\boldsymbol{p}_k^\top$ are 0 for any $k = 1, \ldots, K$, which is stated as Assumption 1(*i*) and 1(*ii*). (S.23) leads to solving

$$\nabla_{\boldsymbol{s}} h_s(\boldsymbol{s}^{t+1}|\boldsymbol{s}^t) = \nabla_{\boldsymbol{s}} f_s(\boldsymbol{s}^t) + \boldsymbol{D}(\boldsymbol{s}^t)(\boldsymbol{s}^{t+1} - \boldsymbol{s}^t) = 0,$$

from which we have

$$s_k^{t+1} = \frac{u_k}{(\boldsymbol{Z}\boldsymbol{s}^t)_k} s_k^t. \tag{S.24}$$

(S.24) corresponds to the update for the k-th coordinate in (6), which yields feasible s_k^{t+1} under the assumptions, since $Z_{ij} = \operatorname{tr}\big(\boldsymbol{G}_i^\top \boldsymbol{G}_j\big) = (\boldsymbol{\theta}_i^\top \boldsymbol{\theta}_j)\operatorname{tr}\big(\boldsymbol{p}_i \boldsymbol{p}_j^\top\big)$, and $u_k = \operatorname{tr}\big(\boldsymbol{Y}^\top \boldsymbol{G}_k\big) = \operatorname{tr}\big(\boldsymbol{Y}^\top \boldsymbol{\theta}_k \boldsymbol{p}_k^\top\big)$. Moreover,

$$Z_{ij} = \operatorname{tr}\big(\boldsymbol{p}_i \boldsymbol{\theta}_i^\top \boldsymbol{\theta}_j \boldsymbol{p}_j^\top\big) = (\boldsymbol{\theta}_i^\top \boldsymbol{\theta}_j)(\boldsymbol{p}_j^\top \boldsymbol{p}_i), \tag{S.25}$$

and that

$$u_k = \operatorname{tr}\big(\boldsymbol{Y}^\top \boldsymbol{\theta}_k \boldsymbol{p}_k^\top\big) = \sum_{j=1}^m \sum_{i=1}^n y_{ji} \theta_{jk} p_{ki}. \tag{S.26}$$

For the denominator in (S.24), we see that

$$(\boldsymbol{Z}\boldsymbol{s}^t)_k = \sum_{l=1}^K Z_{kl} s_l^t \geq Z_{kk} s_k^t = \|\boldsymbol{\theta}_k\|_2^2 \|\boldsymbol{p}_k\|_2^2 s_k^t > 0, \tag{S.27}$$

where the last step is a result of (S.25) and the stability of updates based Assumption 1(*i*) and 1(*ii*), which will be discussed in details below. As for the numerator of (S.24), taking from the stability of updates due to Assumption 1(*i*) and 1(*ii*) and assuming $\theta_{j_k k} > 0$ and $p_{k i_k} > 0$, by (S.26), we have

$$u_k = \sum_{j=1}^m \sum_{i=1}^n y_{ji} \theta_{jk} p_{ki} \geq y_{j_k i_k} \theta_{j_k k} p_{k i_k} > 0,$$

as long as $y_{j_k i_k} > 0$. Given Assumption 1(*iii*), we conclude that $\boldsymbol{s}^{t+1} \in \mathbb{R}^K_{++}$.

B.3. Stability of The Update Steps

To justify the claim that for any j and k, $\theta_{jk}^{t+1} > 0$ whenever $\theta_{jk}^t > 0$ for $t = 0, 1, 2, \ldots$, notice that the denominator of (S.14) is automatically positive whenever $\alpha_1, \alpha_2 > 0$. For guaranteeing that its numerator is also positive, we want to show

$$\big(\operatorname{diag}(\boldsymbol{s}^t)\boldsymbol{P}^t \boldsymbol{y}_j^\top\big)_k = s_k^t \sum_{i=1}^n p_{ki}^t y_{ji} \geq s_k^t p_{k i_k}^t y_{j i_k} > 0, \tag{S.28}$$

where the index i_k is defined as in Assumption 1(*ii*). By Assumption 1(*iii*), $y_{ji_k} > 0$, and $s_k^t > 0$ as discussed above. Thus, (S.28) holds whenever $p_{ki_k}^t > 0$. Similarly, for proving the claim that for any k and i, $p_{ki}^{t+1} > 0$ whenever $p_{ki}^t > 0$, notice that the denominator for (S.19) is also positive whenever $\beta > 0$. To prove that its numerator is positive, we need to show

$$\left(\operatorname{diag}(\boldsymbol{s}^t)(\boldsymbol{\Theta}^{t+1})^\top \boldsymbol{y}_i\right)_k = s_k^t \sum_{j=1}^m \theta_{jk}^{t+1} y_{ji} \geq s_k^t \theta_{j_k k}^{t+1} y_{j_k i} > 0, \tag{S.29}$$

where j_k is defined as in Assumption 1(*i*). (S.29) holds whenever $\theta_{j_k k}^{t+1} > 0$.

Since by Assumption 1(*ii*) and Assumption 1(*i*), $p_{ki_k}^0 > 0$ and $\theta_{j_k k}^0 > 0$. Therefore, we can show that $p_{ki_k}^t > 0$ and $\theta_{j_k k}^{t+1} > 0$ for any $t = 0, 1, 2, \ldots$ by induction through (S.28) and (S.29). Thus, the claims on the stability of updates hold.

C. Proof of Theorem 1

To prove Theorem 1, we leverage on the convergence properties of the block successive upper-bound minimization (BSUM) algorithms proved by Razaviyayn et al. (2013). Namely, we can view Algorithm 1 as a BSUM algorithm by counting a total of $m + n + 1$ separate blocks which the algorithm minimizes successively: m for updating $\boldsymbol{\Theta}^t$, n for $\boldsymbol{P}^t$, and one for $\boldsymbol{s}^t$. To utilize such properties, we need to prove the following proposition, which states the conditions on the update steps' auxiliary functions for Algorithm 1 to formally qualify as BSUM, which is translated from Assumption 2 of Razaviyayn et al. (2013):

Proposition 1 (Algorithm 1 is BSUM)**.** *$h_\theta(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t)$, $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t)$, and $h_s(\boldsymbol{s}|\boldsymbol{s}^t)$ satisfy the following conditions for any $j = 1, 2, \ldots, m$ and $i = 1, 2, \ldots, n$: (1) $h_{\theta_j}(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t) \geq f_\theta(\boldsymbol{\theta}_j^t)$; $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t) \geq f_p(\boldsymbol{p}_i^t)$; $h_s(\boldsymbol{s}|\boldsymbol{s}^t) \geq f_s(\boldsymbol{s}^t)$; (2) $h_{\theta_j}(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t)$, $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t)$, and $h_s(\boldsymbol{s}|\boldsymbol{s}^t)$ are continous on their domains; (3) Any directional derivative of $h_{\theta_j}(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t)$, $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t)$, and $h_s(\boldsymbol{s}|\boldsymbol{s}^t)$ within their domains are equal to the direactional derivative of $f_\theta(\boldsymbol{\theta}_j^t)$, $f_p(\boldsymbol{p}_i^t)$, and $f_s(\boldsymbol{s}^t)$ respectively.*

Proof. The proof is quite straightforward. (1) and (2) are direct from the construction of the surrogate functions and Lemma 1, which also manifests the majorization-minimization (MM) property (Lange, 2016) of each block's update. (3) is seen by differentiating (S.10), (S.18), and (S.22), which leads to $\nabla_{\boldsymbol{\theta}_j} h_\theta(\boldsymbol{\theta}_j^t|\boldsymbol{\theta}_j^t) = \nabla_{\boldsymbol{\theta}_j} f_\theta(\boldsymbol{\theta}_j^t), \forall j$, $\nabla_{\boldsymbol{p}_i} h_p(\boldsymbol{p}_i^t|\boldsymbol{p}_i^t) = \nabla_{\boldsymbol{p}_i} f_p(\boldsymbol{p}_i^t), \forall i$, and $\nabla_{\boldsymbol{s}} h_s(\boldsymbol{s}^t|\boldsymbol{s}^t) = \nabla_{\boldsymbol{s}} f_s(\boldsymbol{s}^t)$. Equal gradients automatically guarantee equal directional derivatives, proving the proposition. □

Now that Algorithm 1 is found to be BSUM, Theorem 1 can be proved directly by the results of Theorem 2(b) in (Razaviyayn et al., 2013). But a few additional conditions presented in the following propositions need to be verified, both of which are proved thereafter.

Proposition 2 (Compact Sub-level Sets)**.** *The sub-level set $\mathcal{X}^0 : \{(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) : f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) \leq f(\boldsymbol{\Theta}^0, \boldsymbol{s}^0, \boldsymbol{P}^0)\}$ of $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ given the update steps in Algorithm 1 is compact.*

Proposition 3 (Unique Global Minimizer of Auxiliary Functions)**.** *Given Assumption 1, $\forall t \in \mathbb{R}_+$ and for all i and j, the auxiliary functions $h_\theta(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t)$, $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t)$, and $h_s(\boldsymbol{s}|\boldsymbol{s}^t)$ each have a unique global minimizer in $\mathbb{R}_+^K$.*

C.1. Proof of Proposition 2

We first introduce the definition of coercive functions through:

Definition 2 (Coercivity)**.** *A function $f(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) : V_1 \times \cdots \times V_N \to \mathbb{R}$, where V_i is a Euclidean vector space endorsed with the L_2 norm if its elements are vectors or with the Frobenius norm if the elements are matrices, is called coercive if $\forall i$, $\|\boldsymbol{X}_i\| \to \infty$ implies $f(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) \to \infty$.*

In fact, continuous coercive functions always have compact sub-level sets, as shown by the following lemma:

Lemma 2. *If a continuous function $f(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) : V_1 \times \cdots \times V_N \to \mathbb{R}$ is jointly coercive, then its sublevel set $\mathcal{X}^r := \{(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) : f(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) \leq r\}$ is compact for any $r \in \mathbb{R}$.*

Proof. Because f is continuous and the space $f(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n) \leq r$ is closed, $\mathcal{X}^r$ is closed. We prove the boundedness of $\mathcal{X}^r$ by contrapositive. We let $\mathcal{X}^r = \mathcal{X}_1^r \times \cdots \times \mathcal{X}_N^r$, where $\mathcal{X}_i^r \subseteq V_i$. If $\mathcal{X}^r$ is unbounded, then one of $\mathcal{X}_i^r$ is unbounded. If $\mathcal{X}_i^r$ is unbounded, we can find a sequence of $\{\boldsymbol{X}_k\}_{k=1}^{\infty}$ such that $\|\boldsymbol{X}_k\| \to \infty$ as $k \to \infty$, which implies f is not jointly coercive. Therefore, whenever f is jointly coercive, $\mathcal{X}^r = \mathcal{X}_1^r \times \cdots \times \mathcal{X}_N^r$ is bounded. Since Euclidean spaces admit the Heine-Borel property, $\mathcal{X}^r$ is compact. □

Coming back to the proof of Proposition 2, it is easy to check that $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ is continuously differentiable. To prove that it is also coercive, we need to show that $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ tends to ∞ whenever $\|\boldsymbol{\Theta}\|_F$, $\|\boldsymbol{s}\|_2$, or $\|\boldsymbol{P}\|_F$ does so.

Beginning with the case for $\boldsymbol{\Theta}$, we have, by $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$,

$$\begin{aligned}
f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) &\geq \frac{\alpha_1}{2}\|\boldsymbol{\Delta} \odot \boldsymbol{\Theta} - \boldsymbol{\Theta}_0\|_F^2 + \frac{\alpha_2}{2}\|\boldsymbol{\Delta}^c \odot \boldsymbol{\Theta}\|_F^2 \\
&\geq \min\left(\frac{\alpha_1}{2}, \frac{\alpha_2}{2}\right)\left(\|\boldsymbol{\Delta} \odot \boldsymbol{\Theta} - \boldsymbol{\Theta}_0\|_F^2 + \|\boldsymbol{\Delta}^c \odot \boldsymbol{\Theta}\|_F^2\right) \\
&\geq \min\left(\frac{\alpha_1}{2}, \frac{\alpha_2}{2}\right)\left(\|\boldsymbol{\Delta} \odot \boldsymbol{\Theta}\|_F^2 - \|\boldsymbol{\Theta}_0\|_F^2 + \|\boldsymbol{\Delta}^c \odot \boldsymbol{\Theta}\|_F^2\right) \\
&= \min\left(\frac{\alpha_1}{2}, \frac{\alpha_2}{2}\right)\left(\|\boldsymbol{\Theta}\|_F^2 - \|\boldsymbol{\Theta}_0\|_F^2\right) \\
&= \min\left(\frac{\alpha_1}{2}, \frac{\alpha_2}{2}\right)\|\boldsymbol{\Theta}\|_F^2\left(1 - \frac{\|\boldsymbol{\Theta}_0\|_F^2}{\|\boldsymbol{\Theta}\|_F^2}\right). \qquad \text{(S.30)}
\end{aligned}$$

Since $\min\left(\frac{\alpha_1}{2}, \frac{\alpha_2}{2}\right) > 0$ and $\frac{\|\boldsymbol{\Theta}_0\|_F^2}{\|\boldsymbol{\Theta}\|_F^2} \leq \frac{mK_0M_\Theta}{\|\boldsymbol{\Theta}\|_F^2} \to 0$ as $\|\boldsymbol{\Theta}\|_F \to \infty$ by Assumption 2, (S.30) then guarantees that $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) \to \infty$ when $\|\boldsymbol{\Theta}\|_F \to \infty$.

In the case for $\boldsymbol{P}$, whenever $\|\boldsymbol{P}\|_F \to \infty$, there is a k such that $\boldsymbol{p}_k^\top$ satisfies $\|\boldsymbol{p}_k^\top\|_2 \to \infty$. Also, since $\boldsymbol{p}_k \in \mathbb{R}_+^n$, there exists a column indexed i_k as defined in Assumption 1(*ii*) such that $p_{ki_k} \to \infty$. Then,

$$\begin{aligned}
f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) &\geq \frac{\beta}{2}\sum_{l=1}^{K} r_l^{-1}\|\boldsymbol{p}_l^\top - \boldsymbol{m}_l\|_2^2 \geq \frac{\beta}{2} r_k^{-1}\|\boldsymbol{p}_k^\top - \boldsymbol{m}_k\|_2^2 \\
&= \frac{\beta}{2} r_k^{-1}\sum_{i=1}^{n}(p_{ki} - \nu_k)^2 \geq \frac{\beta}{2} r_k^{-1}(p_{ki_k} - \nu_k)^2 \to \infty,
\end{aligned}$$

as ν_k is assumed fixed and $\beta > 0$.

Lastly for $\boldsymbol{s}$, we can similarly argue that, because $\boldsymbol{s} \in \mathbb{R}^K_{++}$, $\|\boldsymbol{s}\|_2 \to \infty$ implies there is a k such that $s_k \to \infty$. From the steps leading to (S.21), we have

$$\begin{aligned} f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) &\geq \frac{1}{2mn}\|\boldsymbol{Y} - \boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\|_F^2 = \frac{1}{2mn}\|\boldsymbol{Y} - \sum_{k=1}^K s_k \boldsymbol{G}_k\|_F^2 \\ &\geq \frac{1}{2mn}\left(\|\sum_{l=1}^K s_l \boldsymbol{G}_l\|_F^2 - \|\boldsymbol{Y}\|_F^2\right) \\ &= \frac{1}{2mn}\left(s_k\|\boldsymbol{G}_k\|_F^2 - \|\boldsymbol{Y}\|_F^2\right) \to \infty. \end{aligned}$$

The last line is because $\|\boldsymbol{G}_k\| > 0$ due to Assumption 1(*i*) and Assumption 1(*ii*) and that $\|\boldsymbol{Y}\|_F \leq mnM_Y < \infty$ by Assumption 2. Combining the cases for $\boldsymbol{\Theta}$, $\boldsymbol{P}$, and $\boldsymbol{s}$ completes the proof that $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ is coercive.

Since $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$ is continuous and coercive, the set $\mathcal{X}^0 : \{(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) : f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P}) \leq f(\boldsymbol{\Theta}^0, \boldsymbol{s}^0, \boldsymbol{P}^0)\}$ is compact according to Lemma 2. We have concluded the proof of Proposition 2.

C.2. Proof of Proposition 3

Since $h_\theta(\boldsymbol{\theta}_j|\boldsymbol{\theta}_j^t)$, $h_p(\boldsymbol{p}_i|\boldsymbol{p}_i^t)$, and $h_s(\boldsymbol{s}|\boldsymbol{s}^t)$ are quadratic functions, proving Proposition 3 is equivalent to showing their Hessians are positive definite. Also, we only focus on the positive coordinates of $\boldsymbol{\theta}_j^t$ and $\boldsymbol{p}_i^t$ since other coordinates will remain zero through updates, and the proposition is vacuously true. By (S.10), (S.18), and (S.22), these Hessians are $\boldsymbol{D}(\boldsymbol{\theta}_j^t)$, $\boldsymbol{D}(\boldsymbol{p}_i^t)$, and $\boldsymbol{D}(\boldsymbol{s}^t)$ respectively. Since they are all diagonal matrices, we want to show that their diagonal values are all positive.

For $\boldsymbol{D}(\boldsymbol{\theta}_j^t)$, we have $D(\boldsymbol{\theta}_j^t)_{kk} = \boldsymbol{d}(\boldsymbol{\theta}_j^t)_k = \frac{(\boldsymbol{A}\boldsymbol{\theta}_j^t)_k}{\theta_{jk}^t}$. The denominator of $D(\boldsymbol{\theta}_j^t)_{kk}$ is positive. Dropping the index j, its numerator is

$$\begin{aligned} \left\{\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{P}\boldsymbol{P}^\top\mathrm{diag}(\boldsymbol{s}) + mn(\alpha_1\boldsymbol{V} + \alpha_2\boldsymbol{V}^c)\right)\boldsymbol{\theta}_j^t\right\}_k &\geq \left\{mn(\alpha_1\boldsymbol{V} + \alpha_2\boldsymbol{V}^c)\boldsymbol{\theta}_j^t\right\}_k \\ &\geq mn\min(\alpha_1, \alpha_2)\theta_{jk}^t > 0, \end{aligned}$$

since $\boldsymbol{V} + \boldsymbol{V}^c = \boldsymbol{I}$. Hence, $D(\boldsymbol{\theta}_j^t)_{kk}$ is positive.

For $\boldsymbol{D}(\boldsymbol{p}_i^t)$, from (S.16) and (S.17), we also have $D(\boldsymbol{p}_i^t)_{kk} = \boldsymbol{d}(\boldsymbol{p}_i^t)_k = \frac{(\boldsymbol{C}\boldsymbol{p}_i^t)_k}{p_{ki}^t}$. Likewise, the denominator is positive, and the numerator after dropping the index i is

$$\begin{aligned} \left\{\left(\mathrm{diag}(\boldsymbol{s})\boldsymbol{\Theta}^\top\boldsymbol{\Theta}\mathrm{diag}(\boldsymbol{s}) + mn\beta\boldsymbol{\rho}^{-1}\right)\boldsymbol{p}_i^t\right\}_k &\geq \left(mn\beta\boldsymbol{\rho}^{-1}\boldsymbol{p}_i^t\right)_k \\ &= mn\beta r_k^{-1}p_{ki}^t > 0. \end{aligned}$$

Lastly, $D(\boldsymbol{s}^t)_{kk}$ is already shown to be positive from the conclusion of (S.27). This completes the proof of Proposition 3.

To finally prove Theorem 1, we combine the results of Propositions 1-3, and apply these conclusions to Theorem 2(b) in Razaviyayn et al. (2013). This way, we have shown that Algorithm 1 converges to a set of stationary points of $f(\boldsymbol{\Theta}, \boldsymbol{s}, \boldsymbol{P})$.

D. Supplementary Information for The Simulated Pseudo-Bulk Data

D.1. Data Generation

For $k = 1, 2, \ldots, 5$, s_k^* was generated independently from a χ^2 distribution with 5 degrees of freedom. For the rare class, $\boldsymbol{p}_i^* \overset{i.i.d.}{\sim} \text{Dir}(2^5, 2^4, 2^3, 2^2, 2)$, where Dir represents the Dirichlet distribution. For the uniform class, we first drew $\boldsymbol{\alpha}_i$ from a Dirichlet distribution with $\alpha_{ik} \overset{i.i.d.}{\sim} U(0.3, 0.7), k = 1, \ldots, 5$, and then generated $\boldsymbol{p}_i^*$ from $\text{Dir}(\boldsymbol{\alpha}_i)$. For the extra class, $\boldsymbol{p}_i^* \overset{i.i.d.}{\sim} \text{dir}(2, 2^2, 2^3, 2^4, 2^5)$, in reverse of the rare class. To generate values in $\boldsymbol{\Theta}^*$, for each gene $j = 1, 2, \ldots, M$, a 5×1 vector of expression was generated by

$$\boldsymbol{\theta}_j^{*\top} \sim \Pi\left[\chi^2_{200} \cdot \{(1-\gamma)\text{dir}(1, 1, \ldots, 1) + \gamma \text{dir}(1, \ldots, 1, 5)\}\right],$$

where $\Pi(\cdot)$ is the sampling without replacement operator. $\Pi(\cdot)$ allowed all of CT1-CT5 have a number of gene signatures.

The random errors $\boldsymbol{\epsilon}$ were generated based on the principle of mean-variance dependency in gene expression data: the higher the mean of a gene's expression, the higher their variation (Anders and Huber, 2010). For each gene j, its error-free mean of bulk expression $\bar{y}_j$ was calculated. Then its associated error for each sample i was generated by $e_{ji} \overset{i.i.d.}{\sim} N(0, \sigma^2 \bar{y}_j^2),\ i = 1, 2, \ldots, n,$.

D.2. Method For Selecting Highly Cell-Type-Specific Genes

First, for each gene, the cell type for which it had the highest expression was picked as the target cell type. This gene then became a candidate for a marker of this target cell type. Second, for each candidate marker, the ratio of expression in its target cell type to those in the cell type with the second highest expression was calculated. Third, for each cell type, its candidate marker genes were ranked by the ratio calculated in the second step from highest to lowest. Finally, the top 100 candidate marker genes for each cell type were selected.

D.3. Tuning Grid And Algorithm Parameters

Cross-validation (CV) was performed in each simulation to find the best values for α_1 and β according to Section 2.4. A tuning grid $\mathfrak{A}_1 = \mathfrak{B} = (10^{-2}, 10^{-1}, \cdots, 10^3, 10^4)$ was set up with the tolerance parameter chosen as $\delta = 10^{-5}$.

D.4. Matching The Estimated Proportions to Cell Types For Reference-Free Methods

We utilized $\hat{\boldsymbol{\Theta}}$ and the partial signature matrix $\boldsymbol{\Theta}^*_{K_0}$. Starting from the first column of $\boldsymbol{\Theta}^*_{K_0}$, $\boldsymbol{\theta}^*_{K_{01}}$, we found the column of $\hat{\boldsymbol{\Theta}}$, say column k, that had the highest Pearson's correlation with $\boldsymbol{\theta}^*_{K_{01}}$. Then, the k-th row of $\hat{\boldsymbol{P}}$ was determined to be estimated proportions for CT1. We did this for the rest of the columns of $\boldsymbol{\Theta}^*_{K_0}$ and those of $\hat{\boldsymbol{\Theta}}$, until there was one last unmatched column in $\hat{\boldsymbol{\Theta}}$. That unmatched column, and its corresponding row in $\hat{\boldsymbol{P}}$, were then matched to CT5.

E. Supplementary Method And Results For The Deconvolution of Bulk PBMC Samples from The Human Influenza Vaccine Study

E.1. Data Processing

FASTQ files containing 50bp (base pairs) pair-end raw nucleotide sequence reads of transcripts from Illumina Hiseq 2000 sequencers were downloaded from the Sequence Read Archive (SRA) BioProject PRJNA271578. The raw reads were pre-processed and filtered where reads with length $<$ 50bp, with $>$ 30% bases with quality scores $<$ 30, having an average quality score $<$ 25 in any 10bp interval, or corresponding to special adapter sequences were removed using the software `fastp` (, Chen et al., 2018). Then, reads passing the filters were mapped to reference human genome hg38 GRCh38 Release 43 (downloaded from the GENCODE website) and quantified using `salmon` (, Patro et al., 2017). The output files were then processed using the R package `tximport` (, Soneson et al., 2015): only protein-coding genes with an official gene symbol were selected and their transcripts per million (TPM) values were gathered into a gene-by-sample matrix of gene expression. Additionally, genes with a total TPM of less than 10 across the four cell types were excluded from consideration.

Distinct marker genes for the four cell types were individually selected for each subject using sorted bulk profiles of individual cell types. The selection process identified the top 10 genes exhibiting the highest cell-type specificity for each cell type from each subject according the ratio of expression method outlined in Section 3.1. The union of the two sets of marker genes, totalling 73 genes, were used in the deconvolution.

E.2. Calculating Reference PBMC Proportion Medians and Ranges from External Data

To calculate the reference PBMC cell-type proportion medians and ranges in Table 1, we adopted the data from Kleiveland (2015), a highly cited textbook on PBMC. The median and range for the cell types with reference expression were calculated through the descriptions in the textbook using a simple method. For example, the book states that lymphocytes account for 70% - 90% of all PBMCs, and among the lymphocytes 70% - 85% are T cells. Thus, the range of T cell percentage among PBMCs is 49% - 76.5%, or 0.49 - 0.765 in proportion. The median is thus $(0.49 + 0.765)/2 = 0.6275$ and the range is $0.765 - 0.49 = 0.275$, corresponding to the reference values for T cell in Table 1. The reference values for B cell, NK cell, and monocyte were similarly calculated. For the "other cells", their median proportion is $1 - (0.6275 + 0.0625 + 0.1075 + 0.15) = 0.0525$. As for the ranges, the upper limit of the proportion for "other cells" occurs under the lower limit of lymphocytes and monocyte, which according to Kleiveland (2015) is $1 - (0.7 + 0.1) = 0.2$ (70% lymphocyte, which includes T cell, B cell, and NK cell combined, and 10% monocyte), while the lower limit is 0 (when there are 90% lymphoyte and 20% monocyte, an impossible scenario). This leads to a range of 0.2.

We notice a range of 0.2 could be a little wide for "other cells". As it turned out, these coarse parameters for medians and ranges sufficed for ARTdeConv to work properly. Thus, it is not always necessary to obtain super accurate reference parameters as long as they are in the right orders of magnitude.

E.3. Tuning Grid And Algorithm Parameters

A random seed of 1000 was set in the R programming environment. The tuning grid for α_1 was set as the interval from 2^{-5} to 2^0 with a step size of 0.2 in the power, and that for β was set as the interval from 2^0 to 2^5 also with a step size of 0.2 in the power. α_2 was fixed at 10^{-12}. The difference in magnitudes between α_1 and β in the tuning grids ensured proper regularization, for there was a scale difference between gene expression and the proportions. A 4-fold cross-validation was invoked to choose the optimal tuning parameters. The eventual selected tuning parameter values were within the boundaries of the grids. The tolerance parameter was set as $\delta = 10^{-4}$.

E.4. Fixing mRNA Amounts as One For All Cell Types using ARTdeConv Results in Biased Deconvolution Results

In Section 3.2, it was posited that bulk and gene signature matrices whose expression values were measured in TPM would lose the information on cell-type mRNA amounts, necessitating the inclusion of $\boldsymbol{s}$ described in the underlying model (1). To verify this claim, we re-ran the data analysis on the same data in TPM as in Section 3.2 using the same empirical estimates $\boldsymbol{M}$ and $\boldsymbol{\rho}$, the same tolerance parameter, and the same tuning grid for the hyperparaters, and the same random seed, but coerced diag($\boldsymbol{s}$) to be the identity matrix (i.e. coerced the cell-type mRNA amounts to 1) throughout updates in Algorithm 1. We observed that the estimated proportions for PBMC cell types for both samples on Day 0 deviates more from those measured by flow cytometry when compared to when diag($\boldsymbol{s}$) is not coerced, as shown in Figure 4b.

Table 1: Means and ranges of PBMC cell type proportions calculated from the reference values.

	T cells	**B cells**	**NK cells**	**Monocytes**	**Other cells**
Means	0.6275	0.0625	0.1075	0.15	0.0525
Ranges	0.275	0.055	0.145	0.1	0.2

F. Supplementary Method And Results For The Deconvolution of Bulk PBMC Samples from The COVID-19 Study

F.1. Study Design And Data Processing

The dataset generated from the study of Arunachalam et al. (2020) and downloaded using NCBI GEO accession number GSE152418 contained 34 subjects recruited from Atlanta, GA, USA. It included gene expression from human blood PBMC bulks samples collected from 17 healthy control subjects, one convalescent subject, and 16 subjects diagnosed with COVID-19. Healthy controls were asymptomatic adults whose samples were collected before the widespread circulation of SARS-COV-2 virus in the community. Subjects with COVID-19 diagnosis were further classified into three levels of disease severity based on the based on the adaptation of the Sixth Revised Trial Version of the

Novel Coronavirus Pneumonia Diagnosis and Treatment Guidance. Moderate cases were defined as respiratory symptoms with radiological findings of pneumonia. Severe cases were defined as requiring supplemental oxygen, and ICU-hospitalized cases were those in critical conditions who needed ICU care due to organ failures.

The gene symbols were annotated to 24,259 genes using the gconvert method of the `gProfileR` package in R software. The raw counts were then converted to counts per million (CPM) using those annotated genes. Among the 16 samples from COVID-19 subjects, two (labeled S155 and S179) contained abnormally high levels of *HBB* expression (namely, *HBB* was the most highly expressed gene in those two samples; results are not shown), which could only been found in red blood cells and suggested sample contamination. Therefore, these two samples were removed from the deconvolution procedures. This also led to S155, which also provided the study with single cell samples, being excluded from the comparison in Figure 5d. The one convalescent sample was also excluded due to its unique designation.

The scRNA-seq data were generated by Arunachalam et al. (2020) using CITE-seq of >63,000 cell samples from five healthy controls and seven COVID-19 diagnosed patients. Of the seven patients, three were labeled as moderate cases, three severe cases, and one requiring ICU care. Notably, dendritic cells were enriched by the experimenters and mixed back into the samples for CITE-seq. All five healthy subjects and six of the seven (except S155, which were previously excluded) had measured bulk gene expression from independent PBMC samples as well for the analysis. To process the raw scRNA-seq data downloaded from GEO accession GSE155673, we first merged the single cell gene expression from all 12 subjects together in R package `Seurat` V4 (Hao et al., 2021). We then performed a quality control procedure for the cells, retaining those with a detected number of genes per cell between 200 and 5000 and filtering out cells with >15% mitochondrial counts. Genes with total expression less or equal to 2 across all cell samples were removed as well. We applied the global-scaling normalization method LogNormalize to normalize the feature expression measurements for each cell by the total expression, multiplied this by a scale factor of 10^6, and log-transformed the result. We employed the UMAP dimensional reduction technique to determine the cell type notation. The R package `SingleR` (Aran et al., 2019) was used for the final determination of cell types, using gene signature data from Monaco et al. (2019) as the cell type reference. The single-cell data input now has genes as rows and estimated cell type notations with subjects as columns. In the end, 41,146 cells and 26,531 genes were retained in the gene-by-cell expression matrix of all 12 samples.

We then separated the gene-by-cell matrix into two matrices, one containing only cells from healthy controls (23,531 cells) and the other only those from COVID-19 subjects (17,615 cells). Due to the limits of memory allocations to CIBERSORTx users, 11,765 cells from healthy controls were randomly sampled for each cell type proportionally to its original relative abundance. The cell types were re-grouped by merging all cell subsets of the four major cell types of interest together. The cellular gene expression from cells not belonging to any of the four cell types were discarded for gene signature generation. The processed sub-matrices were then separately sent to CIBERSORTx software's Create Signature Matrix module (, Newman et al., 2019) to obtain gene signature matrices for both control and COVID-19 cell samples. The signature matrix for healthy controls contains 1,368 genes for the four cell types, while that for COVID-19 samples containts 1,388 genes. After matching the common genes from the bulk expression matrices and the signature matrices returned from CIBERSORTx, there were 1,280 genes for the deconvolution

of healthy control samples and 1,297 genes for the deconvolution of COVID-19 patient samples.

F.2. Tuning Grid And Algorithm Parameters

We adopted a tuning grid and a set of parameters for ARTdeConv to the settings in Supplementary Material Section E.3 for the analysis in Section 3.2. A random seed of 1000 was set in the R programming environment. For the deconvolution of healthy control samples, the tuning grid for α_1 was set as the interval from 2^{-3} to 2^1 with a step size of 0.2 in the power, and that for β was set as the interval from 2^2 to 2^6 also with a step size of 0.2 in the power. α_2 was fixed at 10^{-12}. For that of COVID-19 infected samples, the tuning grid for α_1 was set as the interval from 2^{-4} to 2^0 with a step size of 0.2 in the power, and that for β was also set as the interval from 2^2 to 2^6 also with a step size of 0.2 in the power. α_2 was again fixed at 10^{-12}. A 4-fold cross-validation was invoked in both deconvolution analyses to choose the optimal tuning parameters. The eventual selected tuning parameter values were within the boundaries of the grids in each case. The tolerance parameter was set as $\delta = 10^{-4}$.

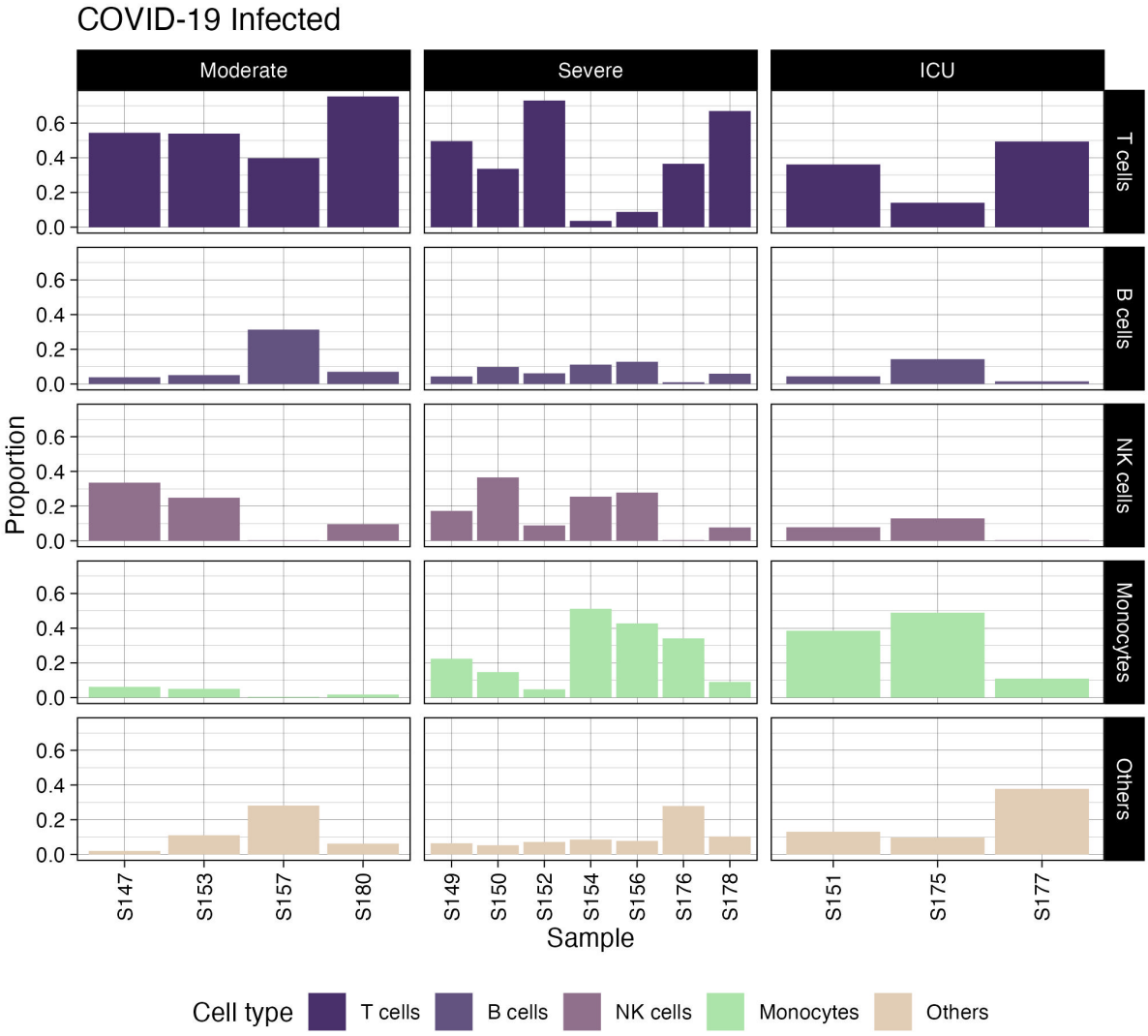

Figure 1: Bar graphs of the deconvolved cell type proportions of all COVID-19 infected samples, grouped by their classified disease severity

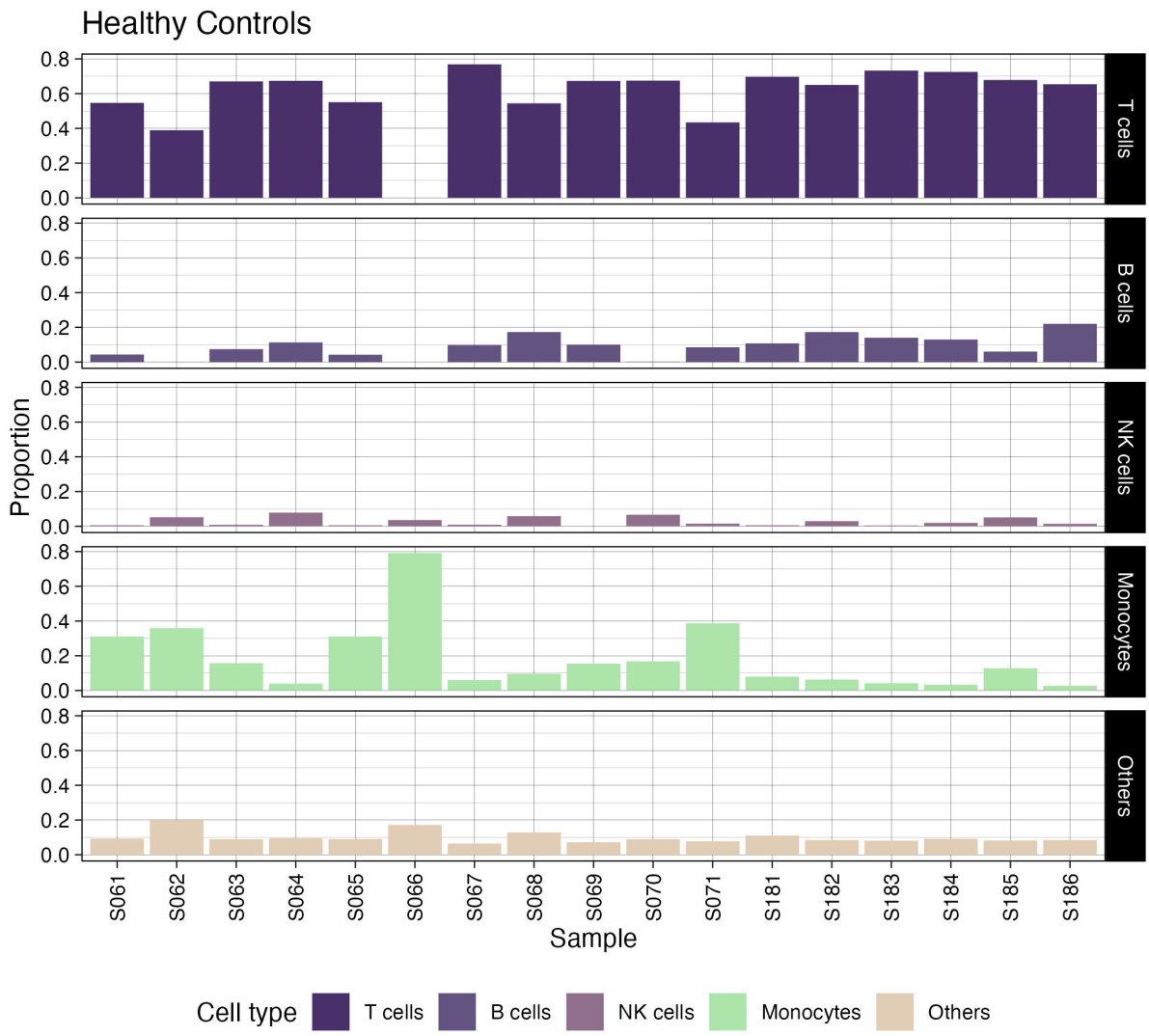

Figure 2: Bar graphs of the deconvolved cell type proportions of all healthy control samples.

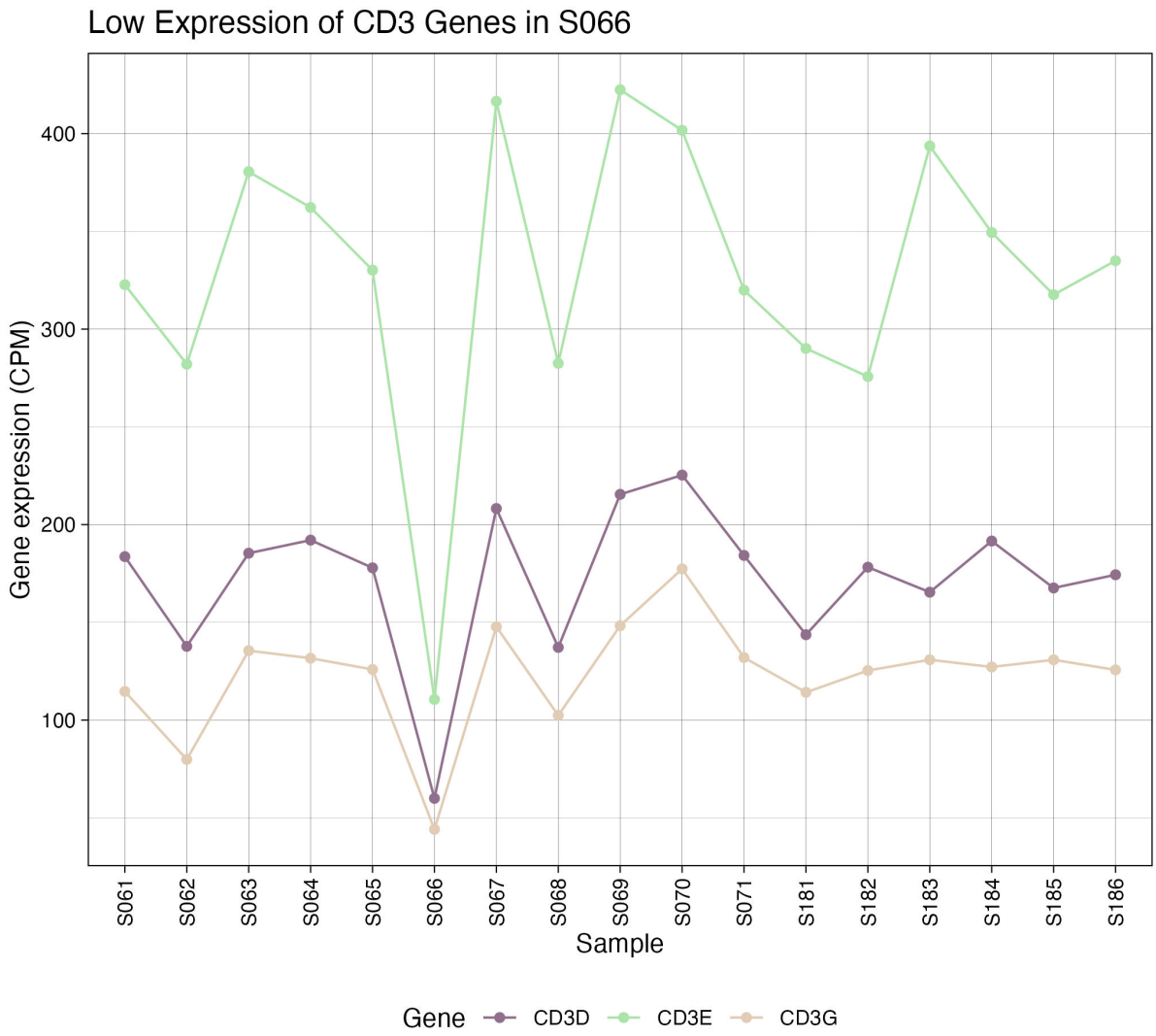

Figure 3: Expression of *CD3* genes, the translation of which produces T cells' CD3 markers, among healthy samples with exceptionally low expression observed in sample S066.

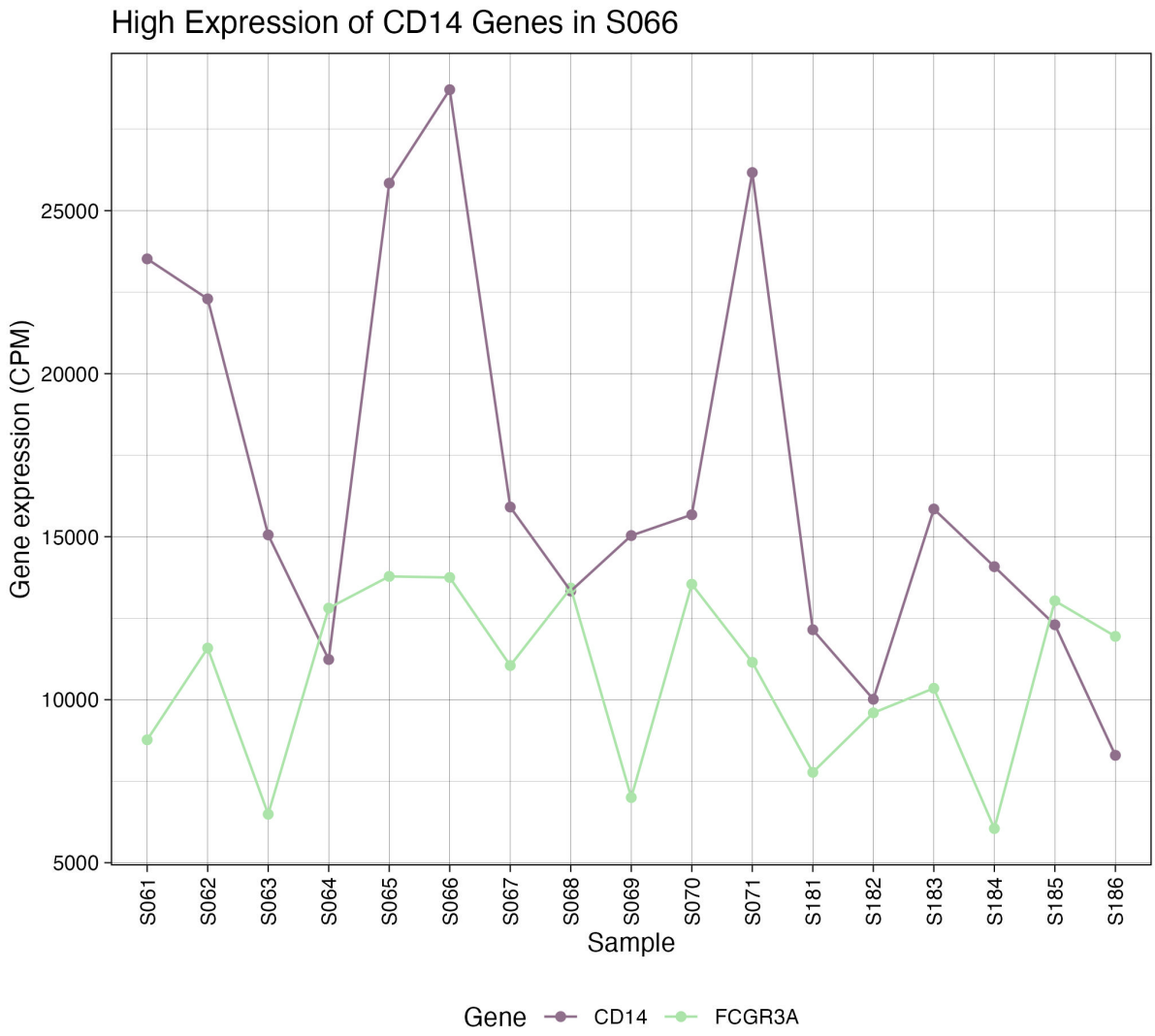

Figure 4: Expression of *CD14* and *FCGR3A* genes among healthy samples with high *CD14* expression observed in sample S066. *CD14* is responsible for the production of CD14 surface markers on classical/intermediate monocytes and *FCGR3A* responsible for that of CD16 surface markers on non-classical/intermediate monocytes.

References

Razaviyayn, M., Hong, M., & Luo, Z.-Q. (2013). *A unified convergence analysis of block successive minimization methods for nonsmooth optimization. SIAM Journal on Optimization*, 23(2), 1126–1153.

Lin, C.-J. (2007). *On the convergence of multiplicative update algorithms for nonnegative matrix factorization. IEEE Transactions on Neural Networks*, 18(6), 1589–1596.

Lee, D., & Seung, H. S. (2000). *Algorithms for non-negative matrix factorization.* In *Advances in neural information processing systems*, 13.

Kim, J., He, Y., & Park, H. (2014). *Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. Journal of Global Optimization*, 58(2), 285–319.

Lange, K. (2016). *MM optimization algorithms.*

Anders S. & Huber W (2010). *Differential expression analysis for sequence count data. Nature Precedings*, pages 1–1.

Chen S., Zhou Y., Chen Y., & Gu J. (2018). *fastp: an ultra-fast all-in-one fastq preprocessor. Bioinformatics*, 34(17):i884–i890.

Patro R., Duggal G., Love M. I., Irizarry R. A., & Kingsford C. (2017). *Salmon provides fast and bias-aware quantification of transcript expression. Nature methods*, 14(4):417–419.

Soneson C., Love M. I., & Robinson M. D. (2015). *Differential analyses for rna-seq: transcript-level estimates improve gene-level inferences. F1000Research*, 4.

Arunachalam, P. S., Wimmers F., Mok C. K. P., Perera R. A. P. M., Scott M., Hagan T., Sigal N., et al. (2020). *Systems Biological Assessment of Immunity to Mild versus Severe COVID-19 Infection in Humans. Science* 369 (6508): 1210–20.

Monaco, G., Lee, B., Xu, W., Mustafah, S., Hwang, Y. Y., Carré, C., Burdin, N., Visan, L., Ceccarelli, M., Poidinger, M., Zippelius, A., Pedro de Magalhaes, J., & Larbi, A. (2019). *RNA-Seq Signatures Normalized by mRNA Abundance Allow Absolute Deconvolution of Human Immune Cell Types. Cell Reports*, 26(6), 1627–1640.e7.

Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W. M., 3rd, Zheng, S., Butler, A., Lee, M. J., Wilk, A. J., Darby, C., Zager, M., Hoffman, P., Stoeckius, M., Papalexi, E., Mimitou, E. P., Jain, J., Srivastava, A., Stuart, T., Fleming, L. M., Yeung, B., ... Satija, R. (2021). *Integrated analysis of multimodal single-cell data. Cell*, 184(13), 3573–3587.e29.

Aran, D., Looney, A. P., Liu, L., Wu, E., Fong, V., Hsu, A., Chak, S., Naikawadi, R. P., Wolters, P. J., Abate, A. R., Butte, A. J., & Bhattacharya, M. (2019). *Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. Nature Immunology*, 20(2), 163–172.

Newman, A. M., Steen, C. B., Liu, C. L., Gentles, A. J., Chaudhuri, A. A., Scherer, F., Khodadoust, M. S., Esfahani, M. S., Luca, B. A., Steiner, D., Diehn, M., & Alizadeh, A. A. (2019). *Determining cell type abundance and expression from bulk tissues with digital cytometry. Nature Biotechnology*, 37(7), 773–782.