Journal Title Here, 2024, pp. 1–22

doi: DOI HERE
Advance Access Publication Date: Day Month Year
Paper

PAPER

VarNMF: Non-negative Probabilistic Factorization with Source Variation

Ela Fallik[1,2] and Nir Friedman[1,2,*]

[1]School of Computer Science and Engineering, The Hebrew University of Jerusalem and [2]Lautenberg Center for Immunology and Cancer Research, Faculty of Medicine, The Hebrew University of Jerusalem
[*]Corresponding author. nir.friedman@mail.huji.ac.il
FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

Motivation: Non-negative Matrix Factorization (NMF) is a powerful tool often applied to genomics data, to identify non-negative latent components that comprise linearly mixed samples. It is useful when the observed signal combines contributions from multiple sources, such as cell types in bulk measurement of heterogeneous tissue. NMF accounts for two types of variations between samples - disparities in proportions of sources and observation noise. However, in many settings, there is also non-trivial variation in the values each source contributes to the samples.
Results: We present VarNMF, a probabilistic extension of NMF that explicitly models source variation. We show that by modeling sources as non-negative distributions, we can recover source variation directly from mixed samples without observing any of the sources directly. We apply VarNMF to a cell-free ChIP-seq dataset of two cancer cohorts and a healthy cohort, demonstrating that VarNMF provides a better estimation of the data distribution. Moreover, VarNMF extracts cancer-associated source distributions that decouple the tumor characteristics from the amount of tumor contribution, and identify patient-specific disease behaviors. This decomposition highlights the inter-tumor variability that is obscured in the mixed samples.
Availability: Code is available at https://github.com/Nir-Friedman-Lab/VarNMF. cfChIP data is based on Sadeh et al. [2021] (publicly available data) and Fialkoff et al. [2022] (provided through authors).
Contact: nir.friedman@mail.huji.ac.il

Introduction

The last few decades have brought great advances in DNA sequencing technologies, facilitating the collection of rich and diverse genomic datasets [Marguerat and Bähler, 2010, Kimura, 2013]. In many applications the sequenced sample represents an aggregate of multiple sources. For example, a liver tissue sample contains hepatocytes but also endothelial cells and multiple types of immune cells. Similarly, a cancer biopsy contains tumor cells, but also a variety of other cell-types from the surrounding tissue [Haider et al., 2020]. Therefore, the signals obtained from such samples represent mixed information from the multitude of cell-types that are present in the physical sample. In many settings, we aim to separate the sources contributing to the mixed signal to gain insights on the underlying processes. For example, biopsies from two cancer patients may differ in the proportion of tumor cells, but also in the tumor-contributing signal itself: certain genes can exhibit variation in signal within the tumor cells of the two patients [Rudin et al., 2019]. Ideally, we want to distinguish between the two types of deviations.

Our motivation for examining separation of mixed signals stems from analysis of genomic data from an assay we recently introduced — *cell-free chromatin immunoprecipitation followed by sequencing* (cfChIP-seq, Fig. 1) [Sadeh et al., 2021]. This assay is performed on plasma (the cell-free portion of blood) and captures DNA fragments marked with a specific epigenetic modification called H3K4me3, which is associated with active and poised promoters [Heintzman et al., 2007]. These cell-free DNA fragments in the plasma originate from dying cells of various sources — cell-types or tissues across the body — according to the cell-type death proportions in the individual [Aucamp et al., 2018]. The chosen modification marks DNA fragments that were located in active genes in the original cells [Soares et al., 2017]. Therefore, the signal from each of these cell-types is tightly coordinated with gene activity. Decomposing the mixed signal into individual components would potentially identify gene related activity in each sub-population of cells contributing to it (e.g., tumor cells, immune cells). This is crucial for interpreting complex samples.

The challenge of decomposing mixed signals was approached from many directions (see review in Shen-Orr and Gaujoux [2013]). Most previous works rely on reference data — a molecular characterization of potential contributing sources — used for estimating and quantifying the proportion of each source in the mixed sample. However, by relying only on previously characterized

sources, these methods are unable to identify new or inaccessible sources of signal. Additionally, these characterizations are usually determined by direct observations from the isolated sources. In many cases obtaining such direct observations is infeasible, and in others the isolation process (e.g., physical cell seperation and sorting) incurs technical biases and thus estimated characterizations are non-transferable.

A different approach to decomposition is a data-driven approach of matrix factorization, employing algorithms such as Principal Component Analysis (PCA) [Jolliffe and Cadima, 2016] and Independent Component Analysis (ICA) [Comon, 1994]. However, since sequencing signal is based on counts of molecules, both the mixed signal and the sources contributing to it are non-negative. Therefore, a more natural model is the *Non-negative Matrix Factorization* (NMF) model [Lee and Seung, 2000], which decomposes the non-negative mixed signal (an $N \times M$ matrix) into two low-rank non-negative matrices:

$$V \approx W \cdot H \tag{1}$$

where $H_k \in \mathbb{R}^M_{\geq 0}$ represents the k-th source as a constant component, and each sample $i = 1, ..., N$ is the result of linearly mixing these components with weights $W[i] \in \mathbb{R}^K_{\geq 0}$ plus a "noise" term. NMF serves as a powerful tool for capturing underlying structure in mixed samples. Beyond sequencing data, it is also relevant to multiple other decomposition problems (e.g., Lee and Seung [1999], Smaragdis and Brown [2003], Arora et al. [2012]).

In the NMF model, variations between samples are due to (i) the mixing proportions W and (ii) observation noise (Fig. 2A,B). However, in many cases there are differences that are not explained entirely by these two factors. cfChip-seq data is one such application: In Sadeh et al. [2021], cfChIP-seq was applied to large number of subjects. The results display differences between samples in the signal originating from specific cell-types, that are not explained by (i) the proportions of cell-death, or (ii) observation noise. For example, in a cohort of patients with liver diseases, there are clear differences between the patient samples and the healthy baseline (Fig. 5 of Sadeh et al. [2021]). Accounting for the increased cell-death proportions of liver tissue in the patients accounts for some of these differences, but not all of them, even when focusing on genes that are specific to the liver. This suggests variation (iii) between samples in the liver signal. In a more recent paper from our group [Fialkoff et al., 2023], a specific variation in liver signal was also identified in an auto-immune liver disease. We can view these results as liver signal not having a constant characterization, but rather as having variation in some genes signal between subjects and across time. Such differences in the state of cells cannot be captured or reasoned about in NMF and its existing extensions.

To account for (iii) variation in sources signal between samples, we introduce a probabilistic graphical model we call VarNMF, where each source k is modeled as a distribution p_k instead of a constant component vector (Fig. 2C), and the parameters of these distributions and the mixing weights are estimated from data using an EM procedure [Dempster et al., 1977]. This modeling also allows us to estimate a source contribution to each particular sample, and to investigate unique behaviors that differ from the "normal" contribution. Next, we present the NMF and VarNMF models formally, discuss algorithms to estimate parameters for both models and compare their performances on synthetic data. Finally, we apply both to real-world cfChIP-seq data, to test how VarNMF confronts the problem presented above, and what is the extent of its interpretability.

Background and Related Works

Notations We use i as sample index, and $X[i]$ as the variable X in the i'th sample. We use j as feature (gene) index, and k as component index. We denote V data matrix of N samples with M features, where each row is a sample $V[i] \in \mathbb{R}^M_{\geq 0}$ that mix signals from K sources according to the weights $W[i] \in \mathbb{R}^K_{\geq 0}$. The sources are represented as K non-negative M-dimensional vectors H_k or distributions p_k (depending on the model, see below). In the later case, we regard each sample-specific instance of the component distribution p_k as a non-negative M-dimensional random vector $\mathcal{H}[i]_k$. Fig. 2D details the dimensions of each object. We use $V[i]_j$ to represent the j'th feature in $V[i]$, and $H_{k,j}, \mathcal{H}[i]_{k,j}$ to represent the j'th feature in the k'th source.

NMF Given a non-negative dataset V, the objective of NMF is to decompose the mixed signal into K sources, each represented by a constant component vector $H_k \in \mathbb{R}^M_{\geq 0}$. The classic formulation from Lee and Seung [2000] is as an optimization problem (Eq. 2), where we look for two low-rank non-negative matrices s.t. their product is the closest to the observations, under some error function $\mathcal{D}$. An equivalent formulation of the same problem (Eq. 3) is as a generative model for which we try to find a maximum likelihood parameters under the assumptions of non-negativity and some noise model [Lee and Seung, 1999]:

$$\hat{W}, \hat{H} = \underset{W,H \geq 0}{\arg\min}\ \mathcal{D}\left(V \mid\mid R\left(W, H\right)\right) \tag{2}$$

$$\text{where } R\left(W, H\right) = W \cdot H$$

$$H \in \mathbb{R}^{K \times M}_{\geq 0}, W \in \mathbb{R}^{N \times K}_{\geq 0}$$

$$\Updownarrow$$

$$\hat{W}, \hat{H} = \underset{W,H \geq 0}{\arg\max}\ P\left(V \mid W, H\right) \tag{3}$$

$$\text{where } V[i]_j \sim P_{\text{obs}}\left(R[i]_j\right)$$

$$R[i]_j = \sum_k W[i]_k \cdot H_{k,j}$$

where $R[i]_j$ is the reconstruction of the mixed signal without noise. A graphical representation of this model is given in Supplementary Fig. S8. Specifically, we will focus on the KL-NMF model, where $\mathcal{D}$ is the KL-divergence error, and its equivalent

generative formulation where the observation probability P_{obs} is a Poisson distribution — a common assumption in the case of biological sequencing data [Anders and Huber, 2010].

This optimization problem is convex in W and H separately, but not together, thus we can only guarantee finding local minima [Lee and Seung, 2000]. Many methods exist for approximating a solution. A simple and widely used method is an iterative algorithm called the *Multiplicative Update Rule* [Lee and Seung, 2000] which is equivalent to block-wise gradient descent over W and H separately, while choosing a learning rate which keeps all the elements of both matrices non-negative. In particular, the KL divergence is non-increasing under this multiplicative update rules and it is invariant under these updates if and only if W and H are at a stationary point of the divergence.

Related Works A summary of notable previous works is offered in Supplementary Fig. S7, including a comparison to VarNMF. We note the main points:

While it is beyond our scope here to discuss the many extensions to NMF (see review in [Wang and Zhang, 2012]), a common theme is adding some form of regularization, structure, or prior assumptions on the matrices H and W (i.e., assuming their values follow some distribution). Specifically, a Bayesian formulation of NMF was introduced in Schmidt et al. [2009], followed by many other formulations, including different choices of priors (e.g. Brouwer and Lio [2017]) and hierarchical Bayesian models (e.g. Lu and Ye [2022], Gopalan et al. [2015]). However, while these Bayesian extensions can integrate prior knowledge on the potential sources into the separation process, they still assume that constant components are shared between samples (even if these are integrated over [Brouwer and Lio, 2017]), and therefore they do not account for variation in the source signals between samples; see Fig. 2.

Recently, Andrade Barbosa et al. [2021] suggested a probabilistic model in which, similarly to VarNMF, components are distributions (in their case, log-normal distributions), and therefore accounts for the variation in source signal between samples. However, these prior distributions are pre-learned, that is, estimated from direct observations of the sources. This approach can insert technical biases and fail in cases where there is no direct access to the sources, or if the sources that comprise the data are a-priori unknown. For example, for cfChIP-seq data, one can in theory assay many liver biopsies of different pathological states to estimate the liver component variation. However, in practice, this approach is fraught with logistical and financial difficulties. For other tissue types (e.g., heart, brain) this is essentially impossible. Moreover, tissue in biopsy can differ from the *in vivo* tissue, due to operation procedure, storage from operation until assay, and more. Here, we overcome this issue by estimating the sources' distributions directly from the data.

Lastly, Rahmani et al. [2019], Wang et al. [2021] suggested tensor decomposition models. These models learn the source distributions from data, and estimate a per-sample component matrix $\mathcal{H}[i]$ for each sample i using the posterior estimation, similar to what we present below. However, their models assume Normal source distributions. While they can restrict the mean of the distribution to achieve non-negative values, their posteriors $\mathcal{H}[i]$ have no such restriction and are likely to receive negative values to compensate for errors. It is therefore hard to interpret these estimations as per-sample source signal. This, among other reasons detailed below, is why we focus instead on the non-negative Gamma distribution, which will result in non-negative mean and posterior estimations of the source signal, contributing to the interpretability of the results.

Method: VarNMF

VarNMF Model We start with the probabilistic formulation of NMF from Eq. 3, and consider the possible variation in source values between samples. To model this variation, we take each component H_k to be a random vector. That is, for each sample i, we have the latent components matrix:

$$\mathcal{H}[i] \in \mathbb{R}_{\geq 0}^{K\times M} \text{ s.t. } \mathcal{H}[i]_k \overset{\text{i.i.d by } i}{\sim} p_k \tag{4}$$

and the data distribution becomes:

$$V[i]_j \sim \text{Poisson}\left(R[i]_j\right) \tag{5}$$
$$\text{where } R[i]_j = \sum_k W[i]_k \cdot \mathcal{H}[i]_{k,j}$$

This model is described in its graphical form in Supplementary Fig. S 8. Importantly, the component signal is now a random vector that has its own instantiation for each sample, and is sampled from the source distribution p_k with some parameters $\theta_{\mathcal{H}_k}$.

Now, given N independent observation vectors $V = (V[1], ..., V[N]) \in \mathbb{R}_{\geq 0}^{N\times M}$, we look for the maximum likelihood estimator (MLE) of $\theta = (W, \theta_{\mathcal{H}})$, that is, the proportion vectors $W \in \mathbb{R}_{\geq 0}^{N\times K}$ and the source distributions' parameters $\theta_{\mathcal{H}}$ s.t.

$$\hat{\theta} = \arg\max_{W,\theta_H} \mathcal{L}_{\text{VarNMF}}\left(W, \theta_{\mathcal{H}}; V\right) \tag{6}$$

For simplicity, we assume that the different features in all components are independent. We also follow a common modeling choice for gene expression [Anders and Huber, 2010] and assume that for each source k, the signal of feature j is distributed according to a Gamma distribution with its own parameters $\theta_{\mathcal{H}_{k,j}} = (A_{k,j}, B_{k,j})$:

$$p_k\left(\mathcal{H}[i]_k\right) = \prod_j p_{k,j}\left(\mathcal{H}[i]_{k,j}\right) \tag{7}$$
$$= \prod_j p_{\text{Gamma}}\left(\mathcal{H}[i]_{k,j}; A_{k,j}, B_{k,j}\right)$$

Together with the Poisson observation noise, we get the commonly used Negative Binomial distribution (Lemma S5).

Likelihood function The task defined by Eq. 6 is hard, as it involves integration over the latent tensor $\mathcal{H}$ for each sample. Specifically, computing the likelihood of a single observation requires a K-dimensional integral:

$$P\left(V[i]_j \mid \theta\right) = \int_{\vec{h}\in\mathbb{R}^K_{\geq 0}} P\left(V[i]_j \mid \mathcal{H}[i]_{:,j} = \vec{h}, W[i]\right) \times \\ \times\, p\left(\mathcal{H}[i]_{:,j} = \vec{h} \mid A_{:,j}, B_{:,j}\right) d\vec{h} \tag{8}$$

One simplification that partially alleviate this complexity is as follows: the Poisson distribution is linear w.r.t its rate. Therefore, we can define another set of latent variables $\mathcal{Y}$ that represent the contribution of signal from each source, with its own Poisson noise:

$$\mathcal{Y}[i]_{k,j} \sim \text{Poisson}\left(W[i]_k \cdot \mathcal{H}[i]_{k,j}\right) \tag{9}$$

and get the deterministic dependency

$$V[i]_j = \sum_k \mathcal{Y}[i]_{k,j} \tag{10}$$

Essentially, we separate the Poisson noise that differentiates the observation $V[i]_j$ from the reconstruction $R[i]_j$, to the noises originating from each source. Now, given the values of $\mathcal{Y}$, the components of H are independent of each other. Thus, we can replace the K-dimensional integration with a K-dimensional summation that can be calculated using dynamic programming (Supplementary Section S2). Moreover, $P\left(\mathcal{Y}[i]_{k,j}; A_{k,j}, B_{k,j}\right)$ which involves integration over $\mathcal{H}[i]_{k,j}$ has a closed form solution (Lemma S5).

Complete-data log-likelihood While we can now theoretically optimize the likelihood by searching over the parameter space, this is infeasible in practice. Instead, we use the EM procedure (Supplementary Section S2 for full details). We start by examining the log-likelihood as though we observe the latent variables $\mathcal{Y}$ and $\mathcal{H}$:

$$\ell^*\left(\theta; V, \mathcal{Y}, \mathcal{H}\right) \overset{\text{def}}{=} \log p\left(V, \mathcal{Y}, \mathcal{H} \mid \theta\right) \tag{11}$$

which can be decomposed into three factors: $\log P\left(V \mid \mathcal{Y}\right)$, $\log P\left(\mathcal{Y} \mid W, \mathcal{H}\right)$ and $\log p\left(\mathcal{H} \mid A, B\right)$. The first factor is a log delta function. The second can be further decomposed for each sample i and source k into a separate log-likelihood function for the parameter $w = W[i]_k$, that accounts for the Poisson noise in $\mathcal{Y}[i]_k$:

$$\ell^{\mathcal{Y}*}_{i,k}\left(w\right) \overset{\text{def}}{=} \log P\left(\mathcal{Y}[i]_k \mid w, \mathcal{H}[i]_k\right) \tag{12}$$

These likelihood functions have sufficient statistics:

$$G[i]_k \overset{\text{def}}{=} \sum_j \mathcal{Y}[i]_{k,j}\,, \qquad\qquad T[i]_k \overset{\text{def}}{=} \sum_j \mathcal{H}[i]_{k,j} \tag{13}$$

The last factor $\log p\left(\mathcal{H} \mid A, B\right)$ represents the source distributions. Since we assumed independence between sources and between features, we can maximize the Gamma log-likelihood of each source k and feature j separately w.r.t. $a = A_{k,j}$ and $b = B_{k,j}$:

$$\ell^{\mathcal{H}*}_{k,j}\left(a, b\right) \overset{\text{def}}{=} \log p\left(\mathcal{H}[1]_{k,j}, ..., \mathcal{H}[N]_{k,j} \mid a, b\right) \tag{14}$$

using the sufficient statistics of the Gamma distribution:

$$S^0 \overset{\text{def}}{=} N, \qquad S^1_{k,j} \overset{\text{def}}{=} \sum_i \mathcal{H}[i]_{k,j}\,, \qquad S^{\log}_{k,j} \overset{\text{def}}{=} \sum_i \log \mathcal{H}[i]_{k,j} \tag{15}$$

Expectation Maximization (EM) procedure Given a starting point $\theta^{(0)} = \left(W^{(0)}, \theta^{(0)}_{\mathcal{H}}\right)$, we apply the following Expectation (E-) and Maximization (M-) steps iteratively until convergence of the marginal log-likelihood $\ell_{\text{VarNMF}}\left(W, \theta_{\mathcal{H}}; V\right)$:

In the E-step we calculate the expectation of the sufficient statistics (the ESS) from Eq. 13 and Eq. 15 w.r.t. the posterior $p\left(\mathcal{Y}, \mathcal{H} \mid V, \theta^{(t)}\right)$. The full process is described in Supplementary Section S2, but essentially it is sufficient to calculate the following probabilities for each sample i and feature j:

$$\forall_{k,d},\ p\left(V[i]_j \mid \mathcal{Y}[i]_{k,j} = d; \theta^{(t)}\right) \\ = p\left(\sum_{l\neq k} \mathcal{Y}[i]_{l,j} = V[i]_j - d; \theta^{(t)}\right) \tag{16}$$

which can be achieved using the same dynamic programming procedure of the log-likelihood calculation.

In the M-step, we maximize the expectation of the complete-data log-likelihood:

$$\theta^{(t+1)} = \arg\max_\theta\, \mathbb{E}_{p\left(\mathcal{Y},\mathcal{H}\mid V,\theta^{(t)}\right)}\left[\ell^*\left(\theta; V, \mathcal{Y}, \mathcal{H}\right)\right] \tag{17}$$

From linearity of expectation, we can find $\theta^{(t+1)} = (W, A, B)$ by separately maximizing

$$\mathbb{E}\left[\ell^{\mathcal{Y}*}_{i,k}\left(w\right)\right] \qquad\qquad \mathbb{E}\left[\ell^{\mathcal{H}*}_{k,j}\left(a, b\right)\right] \tag{18}$$

These are the same functions as the log-likelihood functions $\ell^{\mathcal{Y}*}_{i,k}$ and $\ell^{\mathcal{H}*}_{k,j}$, only with the ESS calculated in the E-step replacing the actual sufficient statistics. Therefore, they can be maximized in the same way.

Convergence and implementation Following the EM scheme, we assure convergence to a local maximum. In our case, the E-step is computationally demanding while the M-step is straightforward. As a starting point we use the NMF solution of W and H (with a random start and the multiplicative update algorithm). We use the estimated H to initialize the mean of the Gamma distributions over $\mathcal{H}$ and initialize the variance s.t. the coefficient of variation is constant. We use a simple stopping criteria of $T = 100$ iterations for synthetic data and $T = 250$ for real data in the training stage, and run until convergence in the test stage (see below). An additional issue with both NMF and VarNMF solutions is that they are not identifiable, and many solutions will have the same log-likelihood. Therefore, to compare between solutions, we need to normalize. We expand on the normalization issue in Supplementary Section S2.

Posterior expected signal Using the training data, we estimate a prior distribution for each source, $\hat{p}_k = \text{Gamma}\left(\hat{A}_k, \hat{B}_k\right)$. The mean of this distribution can be interpreted similarly to the constant components provided by NMF. However, under the VarNMF model assumptions, each source k contributes some signal $\mathcal{H}[i]_k$ to sample i, weighted by $W[i]_k$. This sample-specific signal represents the isolated contribution of source k to sample i, and estimating it can help identify cases where the true contribution of the source to the particular sample is far from expected. We estimate this signal using the expectation of the *posterior source distribution* of the sample,

$$\hat{p}\,[i]_k\,(h) = p(\mathcal{H}[i]_k = h \mid V[i]\,, \hat{A}_k, \hat{B}_k) \tag{19}$$

which is calculated in the E-step (Supplementary Section S2).

Experiments

We start by applying VarNMF to synthetic data based on a cfChIP-seq dataset properties, with increasing variation in the sources signal. We compare the results with those of the NMF model. Next, we apply both algorithms to a cfChIP-seq dataset, to test real life performance.

Synthetic Data

To illustrate the capability of VarNMF for non-negative decomposition with source variation, we consider mixed datasets with $M = 100$ features and $K = 1, ..., 10$ sources. The sources are modeled as Gamma distributions with mean $\mu_{k,j}$ and variance $\sigma^2_{k,j}$ for the j'th feature in the k'th component. The means are generated by random permutations of an NMF solution trained on a real cfChIP-seq dataset. We control the variation level of the sources by setting the coefficient of variation to a constant value, thus $\sigma_{k,j} = \text{CV} \cdot \mu_{k,j}$. In each realization, we generate $N = 100$ samples from each source (Lemma S2):

$$\begin{aligned} &\mathcal{H}[i]_{k,j} \overset{\text{i.i.d.}}{\sim} \text{Gamma}\,(A_{k,j}, B_{k,j}) \\ &\text{with } A_{k,j} = \frac{1}{\text{CV}^2}, B_{k,j} = \frac{1}{\text{CV}^2 \mu_{k,j}} \end{aligned} \tag{20}$$

We mix them using a weight matrix W whose rows are drawn i.i.d. from a symmetric Dirichlet (1) distribution, multiplied by a per-sample scaling factor $\lambda\,[i] \sim \mathcal{U}\,([1, 2])$. We then sample an observation $V[i]_j$ from a Poisson with the mixture $R[i]_j = W[i]^T \cdot \mathcal{H}[i]_j$ as its rate.

To evaluate VarNMF vs. NMF, we examine their performance with the correct number of sources K (see Supplementary Section S3 for results with the wrong K). Since the number of parameters in VarNMF is higher than in NMF, we also apply NMF with $\tilde{K}$ sources, where $\tilde{K}$ is the minimal number to compensate for the difference in degrees of freedom.

Our goal is to extract accurate source distributions to better analyse new samples. Thus, we measure the generalization abilities of each model on a new dataset sampled from the same distribution. We created test datasets with $N_{\text{test}} = 100$ and used a version of NMF and VarNMF to fit new weight matrices W_{test} while keeping the learned components\ source distributions parameters constant (Supplementary Section S2).

Applied to the synthetic datasets, the proposed VarNMF model achieves high log-likelihood for the train data, despite increasing CV (Fig. 3A and Supplementary Fig. S10). Although the NMF has similar performance for low levels of variation, its score drops sharply as these levels increase. This is not due to differences in number of parameters, as $\tilde{K}$-NMF follows the same trend. Importantly, the test log-likelihood performance of all methods are similar to the train results, suggesting there is little overfitting. The results are also similar when increasing the number of samples N and the observational noise λ (Supplementary Section S3). This suggests that VarNMF better captures the datasets distribution in the presence of high source variation.

Next, we examine the learned components against the ground truth parameters and observe that the means of the VarNMF distributions are closer to the ground truth means than NMF constant estimates (Fig. 3B and Supplementary Fig. S10). A similar improvement can be seen when comparing the ground truth per-sample contribution of source k, $\mathcal{H}[i]_k$, to the VarNMF posterior expected signal versus the NMF constant components (Fig. 3C and Supplementary Fig. S10).

Overall, we conclude that for data with source variation, VarNMF learns more accurate mean signal for the different sources, and allows for a sample-specific source value estimation via the posterior expect signal.

Real Data

We collected a dataset of cfChIP-seq samples from Sadeh et al. [2021] and Fialkoff et al. [2022]. This data includes 80 plasma samples of healthy subjects, 139 samples of small-cell lung cancer (SCLC) patients, and 86 samples of colorectal cancer (CRC)

patients (some patients were sampled at multiple time points). The two cancer cohorts represent different diseases, yet they are both solid tumors and are expected to have some common features. The cell-free literature reports on large variation in the fraction of tumor DNA in circulation [Zill et al., 2018]. Thus, there is non-trivial heterogeneity in terms of the mixing proportions. There are also reports on molecular differences among cancers of the same type [Sadeh et al., 2021, Fialkoff et al., 2023], and therefore we expect to observe some variability in the signal of the component(s) representing each cancer type.

Out of this dataset, we selected $M = 7000$ genes (Supplementary Section S4). Additionally, instead of training with the EM algorithm directly, we use a scheme that alternates optimization of W and A, B, to allow for parallel optimization of the Gamma distributions of the different features (Supplementary Section S2). Another adjustment for this data is for a non-specific background noise originating from the sequencing assay [Sadeh et al., 2021]. The levels of this noise are estimated as part of the assay and we regard to it as another source of signal (i.e., another component), with pre-estimated contribution and signal. In particular, both NMF and VarNMF can incorporate this noise with minor changes to their training algorithms. We randomly divided the cohort into 185 training samples and 120 test samples (repeated 5 times). We trained the different procedures on the training data with increasing number of components K without any labels. We test the trained model using the same scheme as for synthetic data (Supplementary Section S2).

We start by evaluating the ability of the different models to generalize to new instances from the same cohort. Plotting the log-likelihood as a function of the number of components K (Fig. 4A), we observe that all models perform better on training data as K increases. However, there is a clear advantage for VarNMF on test data ($\sim$1.2 nats/observation) that does not diminish with larger Ks. Consequently, we conclude that VarNMF is more effective at learning representations of the inherent structure in the data. Moreover, these results suggest the relevance of the source variation model for describing the underlying biology.

Next, we examine the biological relevance of the learned components in a specific solution. While there is no clear optimal K, we choose the $K = 4$ solution as a representative example (Supplementary Section S4 for analysis of other values of K). Examining the range of values of W for the three sub-cohorts (Fig. 4B) we see two cancer-specific components (that have non-zero weights mostly in one cancer type) - components #3 and #4, and two shared "healthy" components (that are high in healthy samples but also appear in a range of weights in the cancer samples) - components #1 and #2. This is to be expected given that the cancer contributes only a part of the cell-free material in the plasma. Moreover, the values of W for the two cancer-specific components are correlated to supervised estimations of disease scores ($r = 0.98$ and 0.95, for W3 and W4 respectively, Supplementary Fig. S13).

An alternative way of interpreting the biological association of the components is to examine the mean contribution of a component to the value of each gene. Specifically, we can choose genes that have high mean value in the component and low mean values in all other components (differential genes; Supplementary Section S4). We then test whether these genes are significantly over-represented in curated genes-lists associated with a specific tissue, cell-type or cell-line [Chen et al., 2013]. Results for the first three components are similar between NMF and VarNMF. The first two components' genes are strongly enriched for Platelets and Neutrophils, which are the two main sources of cell-free DNA in healthy samples [Sadeh et al., 2021, Moss et al., 2018]. Component #2 is also enriched for Macrophages that differentiate from Monocytes which are also found in high concentrations in cell-free DNA from healthy samples. Component #3 (associated mainly with SCLC patients) is enriched for SCLC-derived cell-lines in both the NMF and VarNMF solutions. This indicates that this component indeed represents the tumor derived cell-free DNA in the SCLC patients plasma. Component #4 (associated mainly with CRC patients) displays different associations between the two models: The NMF solution is enriched exclusively for colon-derived cell-lines, aligning with the CRC patients diagnosis. On the other hand, the VarNMF solution is enriched for both colon and liver derived cell-lines. The later enrichment may reflect liver metastasises or liver damage in many of the CRC patients in this cohort. We conclude that the mean of the components estimated by VarNMF captures the main expected contributors of cell-free DNA to the cohort samples. Importantly, the NMF solution yields a similar interpretation, and these findings are not exclusive to the VarNMF estimation.

To illustrate the unique features of VarNMF, we examine a specific sample of a CRC-patient (Fig. 5A). Mixing the NMF constant components according to the weights learned for this sample results in a reconstruction that significantly diverges from the observed signal in hundreds of genes. Similarly, mixing the means of the component distributions learned by VarNMF according to the VarNMF weights for the sample also results in many unexplained genes, even more than for the NMF reconstruction. However, when using the sample-specific component posteriors estimated by VarNMF, the observed signal is fully explained. Thus, while much of the variation between samples can be accounted for by the mixing proportions, there is a non-trivial additional layer of variability due to variation in signals within components. Examining these posterior signals (Fig. 5B), we notice that while for the first three components the posterior signal is close to the prior value, the fourth CRC-associated component exhibits sample-specific signal. This suggests that most of the discrepancies originate from the CRC source, i.e., that the disease-specific features explain most of the previously unexplained signals in the sample. These behaviors are repeated in most samples in both the train and test datasets.

Looking at this phenomena more generally (Fig. 5C), the main directions of variation in the mixed samples are the estimated percentage of disease (Pearson correlation of PC1 and $W_3 = 0.89$; of PC2 and $W_4 = 0.84$). In contrast, the posterior expected signal allows us to separate sample-specific disease behavior and observe inter-cancer variability: the main directions of variation are no longer associated with degree of disease, but with unique features of sub-populations of patients. For example, in the CRC-associated component (component #4), the first two PCs of the posterior expected signal separate two sub-populations from the main population of patients. These sub-populations have two different known genomic amplifications [Sadeh et al., 2021] (ERBB2 with PVE=32% and HNF4A with PVE=12%). Similarly, in the SCLC-associated component (component #3), PC1 is associated with MYC amplification (PVE=25%) and PC2-5 with other specific amplifications (PVE=9% to 3%). Genomic amplification results in increased copy-number of a chromosomal region, thereby amplifying the expression of one or more oncogenes that contribute to the increased fitness of cells with the amplification, and contributing to cancer development. Therefore, identifying these amplifications can be relevant to prognosis, treatment planning, etc.

Beyond amplifications, we can also examine more complex patterns of variation. Focusing on the CRC-related posterior of the differential genes of the CRC-related component, we obtain a complex cluster map (Fig. 6). We identify four main pairs of sample - gene clusters, where the relevant genes are elevated in the associated samples. They match biological aspects of the patients and tumor: ERBB2 genomic amplification, but also functional phenotype of the tumor, and increased liver damage.

Conclusion

Here, we presented VarNMF, a model for decomposition of non-negative data into source distributions and mixing proportions, in a way that tackles variation in source values between samples. We evaluated VarNMF on synthetic data and found that in the presence of source variation, it better generalizes to unseen samples and offers a more accurate representation of the data distribution. We further applied VarNMF to cfChIP-seq data, illustrating its capacity to decompose real-world data into biologically relevant source distributions that accurately represent the entire population. Additionally, VarNMF estimates the sample-specific posterior expected signal of a source, which reflects the source contribution to a specific sample and can indicate patient-specific disease behavior. This potentially allows for direct access to disease behavior in patients across time and following treatment in a non-invasive manner.

More generally, we believe that source variation is prevalent in many scenarios where NMF is applied to biological data. In bulk samples, for example, contributing cells of some type can deviate from the stereotypical profile of the cell type. While in theory, one can compensate for source variation by adding multiple "related" constant components that span the variation (e.g., inflamed hepatocytes vs healthy hepatocytes), this solution is less robust and harder to interpret. Moreover, the distinction between mixing proportions and source values allows estimation of the unique profile in each sample. As we showed, such estimation uncovers additional biologically meaningful patterns in the data.

We presented VarNMF in the simplest form, and many improvements can be introduced. For example, while the alternating EM procedure allows for scaling to large datasets, it remains computationally expensive, and further efforts may be taken to speed up training. Learning prior distribution over W can provide another boost in generalization. The model of $p(\mathcal{H}_k)$ is simple and does not capture dependencies between features, although non-trivial dependencies are expected in genomics data. In principle, the modeling framework presented here can be extended to include these modeling changes.

In a broader perspective, the approach we presented here provides a framework for learning about distributions of complex latent sources without observing them directly, a recurring theme when attempting to study *in vivo* molecular states from compound observations. This raises interesting questions regarding the limits on generalizing such models from mixed observations.

Competing interests

Nir Friedman is a founder and share holder in Senseera.

Acknowledgments

We would like to thank Gavriel Fialkoff and members of the Friedman lab for comments and suggestions. This work was supported in part by the European Research Council (ERC Adg #101019560 "cfChIP") and Israel Science Foundation (Grant no 3751/21).

References

S. Anders and W. Huber. Differential expression analysis for sequence count data. *Nat. Precedings*, 2010.

B. Andrade Barbosa, S. van Asten, J. Oh, A. Farina-Sarasqueta, J. Verheij, F. Dijk, H. van Laarhoven, B. Ylstra, J. Garcia Vallejo, M. van de Wiel, et al. Bayesian log-normal deconvolution for enhanced in silico microdissection of bulk gene expression data. *Nat. Comm.*, 12:6106, 2021.

S. Arora, R. Ge, and A. Moitra. Learning topic models–going beyond SVD. In *IEEE 53rd Ann. Symp. Found. Comp. Sci.*, pages 1–10, 2012.

J. Aucamp, A. Bronkhorst, C. Badenhorst, and P. Pretorius. The diverse origins of circulating cell-free DNA in the human body: A critical re-evaluation of the literature. *Biological Reviews*, 93:1649–1683, 2018.

T. Brouwer and P. Lio. Prior and likelihood choices for bayesian matrix factorisation on small datasets. *arXiv preprint arXiv:1712.00288*, 2017.

E. Chen, C. Tan, Y. Kou, Q. Duan, Z. Wang, G. Meirelles, N. Clark, and A. Ma'ayan. Enrichr: interactive and collaborative html5 gene list enrichment analysis tool. *BMC Bioinf.*, 14:1–14, 2013.

T. Chu, Z. Wang, D. Pe'er, and C. Danko. Cell type and gene expression deconvolution with bayesprism enables bayesian integrative analysis across bulk and single-cell rna sequencing in oncology. *Nat. Cancer*, 3:505–517, 2022.

P. Comon. Independent component analysis, a new concept? *Signal processing*, 36:287–314, 1994.

A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. the royal statistical society: series B (methodological)*, 39:1–22, 1977.

D. Donoho and V. Stodden. When does non-negative matrix factorization give a correct decomposition into parts? *Adv. Neu. Inf. Proc. Sys.*, 16, 2003.

G. Fialkoff, N. Takahashi, I. Sharkia, J. Gutin, L. Pongor, A. Rajan, S. Nichols, L. Sciuto, R. Vilimas, C. Graham, et al. Subtyping of small cell lung cancer using plasma cell-free nucleosomes. *bioRxiv*, pages 2022–06, 2022.

G. Fialkoff, A. Ben Ya'akov, I. Sharkia, R. Sadeh, J. Gutin, C. Goldstein, A. Khalaileh, A. Imam, R. Safadi, Y. Milgrom, et al. Identification of hepatocyte immune response in autoimmune hepatitis from human plasma cfChIP-seq. *medRxiv*, pages 2023–06, 2023.

P. Gopalan, J. Hofman, and D. Blei. Scalable recommendation with hierarchical poisson factorization. In *UAI*, pages 326–335, 2015.

S. Haider, S. Tyekucheva, D. Prandi, N. Fox, J. Ahn, A. Xu, A. Pantazi, P. Park, P. Laird, C. Sander, et al. Systematic assessment of tumor purity and its clinical implications. *JCO Precision Oncology*, 4:995–1005, 2020.

N. Heintzman, R. Stuart, G. Hon, Y. Fu, C. Ching, R. Hawkins, L. Barrera, S. Van Calcar, C. Qu, K. Ching, et al. Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. *Nat. Genetics*, 39:311–318, 2007.

K. Huang, N. Sidiropoulos, and A. Swami. Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition. *IEEE Trans. Sig. Proc.*, 62:211–224, 2013.

I. Jolliffe and J. Cadima. Principal component analysis: A review and recent developments. *Phil Trans Roy Soc A*, 374:20150202, 2016.

H. Kimura. Histone modifications for human epigenome analysis. *J. human genetics*, 58:439–445, 2013.

D. Koller and N. Friedman. *Probabilistic graphical models: Principles and techniques*. MIT press, 2009.

D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

D. Lee and H. Seung. Algorithms for non-negative matrix factorization. *Adv. Neu. Inf. Proc. Sys.*, 13, 2000.

J. Lu and X. Ye. Flexible and hierarchical prior for bayesian nonnegative matrix factorization. *arXiv preprint arXiv:2205.11025*, 2022.

S. Marguerat and J. Bähler. RNA-seq: From technology to biology. *Cellular and Molecular Life Sciences*, 67:569–579, 2010.

J. Moss, J. Magenheim, D. Neiman, H. Zemmour, N. Loyfer, A. Korach, Y. Samet, M. Maoz, H. Druid, P. Arner, et al. Comprehensive human cell-type methylation atlas reveals origins of circulating cell-free DNA in health and disease. *Nat. Comm*, 9:5068, 2018.

R. Neal and G. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, pages 355–368. 1998.

A. Newman, C. Steen, C. Liu, A. J. Gentles, A. Chaudhuri, F. Scherer, M. Khodadoust, M. Esfahani, B. Luca, D. Steiner, et al. Determining cell type abundance and expression from bulk tissues with digital cytometry. *Nat. Biotech*, 37:773–782, 2019.

R. Plemmons and R. Cline. The generalized inverse of a nonnegative matrix. *Proc. Am. Math. Soc.*, 31:46–50, 1972.

E. Rahmani, R. Schweiger, B. Rhead, L. Criswell, L. Barcellos, E. Eskin, S. Rosset, S. Sankararaman, and E. Halperin. Cell-type-specific resolution epigenetics without the need for cell sorting or single-cell biology. *Nat. Comm*, 10:3417, 2019.

C. Rudin, J. Poirier, L. Byers, C. Dive, A. Dowlati, J. George, J. Heymach, J. Johnson, J. Lehman, D. MacPherson, et al. Molecular subtypes of small cell lung cancer: A synthesis of human and mouse model data. *Nat. Rev. Cancer*, 19:289–297, 2019.

R. Sadeh, I. Sharkia, G. Fialkoff, A. Rahat, J. Gutin, A. Chappleboim, M. Nitzan, I. Fox-Fisher, D. Neiman, G. Meler, et al. Chip-seq of plasma cell-free nucleosomes identifies gene expression programs of the cells of origin. *Nat. Biotech*, 39:586–598, 2021.

M. Schmidt, O. Winther, and L. Hansen. Bayesian non-negative matrix factorization. In *Independent Component Analysis and Signal Separation: 8th International Conference*, pages 540–547, 2009.

S. Shen-Orr and R. Gaujoux. Computational deconvolution: Extracting cell type-specific information from heterogeneous samples. *Cur. Op. Immunology*, 25:571–578, 2013.

P. Smaragdis and J. Brown. Non-negative matrix factorization for polyphonic music transcription. In *IEEE Workshop on App. Sig. Proc. Audio and Acoustics*, pages 177–180, 2003.

L. Soares, P. He, Y. Chun, H. Suh, T. Kim, and S. Buratowski. Determinants of histone H3K4 methylation patterns. *Mol. Cell*, 68:773–785, 2017.

D. Wackerly, W. Mendenhall, and R. Scheaffer. *Mathematical statistics with applications*. Cengage Learning, 2014.

J. Wang, K. Roeder, and B. Devlin. Bayesian estimation of cell type–specific gene expression with prior derived from single-cell data. *Genome research*, 31:1807–1818, 2021.

Y. Wang and Y. Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Trans. Knowledge and Data Engineering*, 25:1336–1353, 2012.

O. Zill, K. Banks, S. Fairclough, S. Mortimer, J. Vowles, R. Mokhtari, D. Gandara, P. Mack, J. Odegaard, R. Nagy, et al. The landscape of actionable genomic alterations in cell-free circulating tumor DNA from 21,807 advanced cancer patients. *Clin. Cancer Res.*, 24:3528–3538, 2018.

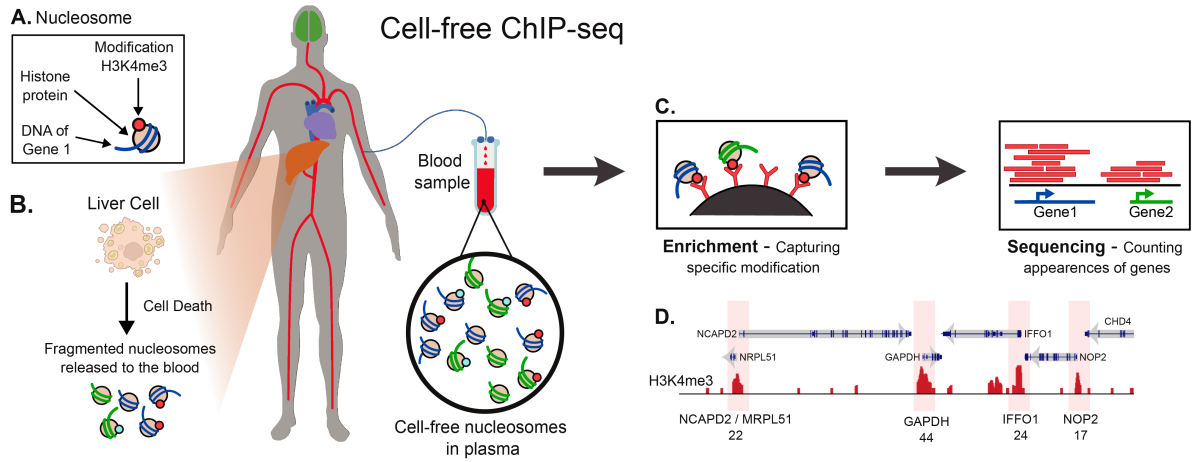
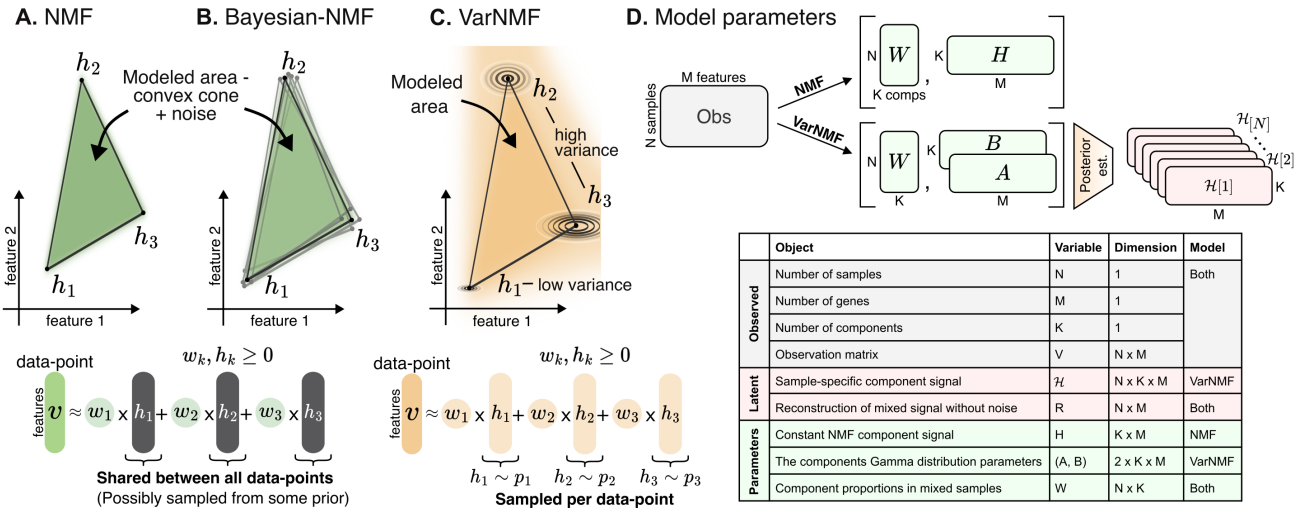


Fig. 1. cfChIP concept - A) Genomic DNA in the nucleus is packaged into nucleosome complexes made of DNA wrapped around histone proteins. The histone protein at each nucleosome can be modified in a way that is tightly coordinated with gene activity at that position [Soares et al., 2017]. B) Upon cell-death, the genome is fragmented and nucleosomes are released into the circulation as cell-free nucleosomes that retain their modifications [Aucamp et al., 2018]. C) cfChIP-seq [Sadeh et al., 2021] uses immunoprecipitation to capture modified nucleosomes from plasma, and then sequence the DNA fragments bound to these nucleosomes. By mapping these sequences to the genome, we can associate them with genes. Thus, similar to RNA-seq data, this assay provides a quantitative signal of activity associated with each gene. This signal reflects the aggregate contribution of all cells that released modified nucleosomes into the circulation, and thus if we could break it into individual components, we would be able to report on each sub-population of cells (e.g., tumor cells, immune cells). D) Illustration of how cfChIP-seq results are quantified. For each gene we use a pre-defined promoter region (pink box), and we count the number of fragments whose center lies within the region. The resulting vector of counts per gene is processed to estimate background and possible normalization [Sadeh et al., 2021].



	Object	Variable	Dimension	Model
Observed	Number of samples	N	1	Both
	Number of genes	M	1	
	Number of components	K	1	
	Observation matrix	V	N x M	
Latent	Sample-specific component signal	$\mathcal{H}$	N x K x M	VarNMF
	Reconstruction of mixed signal without noise	R	N x M	Both
Parameters	Constant NMF component signal	H	K x M	NMF
	The components Gamma distribution parameters	(A, B)	2 x K x M	VarNMF
	Component proportions in mixed samples	W	N x K	Both

Fig. 2. A-C) Illustrative examples of mixed samples with $K = 3$ non-negative components (in black) and two representative features. A) The NMF model assumptions allow us to model a convex cone between the constant components h_1, h_2, h_3, plus some noise. B) In Bayesian-NMF there is a prior over the location of h_1, h_2, h_3, but the modeled area is still a convex cone. C) In contrast, adding variation to the sources, each data-point has its own instantiation of sources contribution, sampled from the corresponding component distributions. This results in a wider modeled area. Separating the sources from mixed samples using NMF in this scenario will result in a solution that is wider than the original sources that created the data. D) Details of observed variables (grey), model parameters (green), and latent variables (red) – variables that are integrated over in the model – for NMF and VarNMF, and the dimensions of each object.

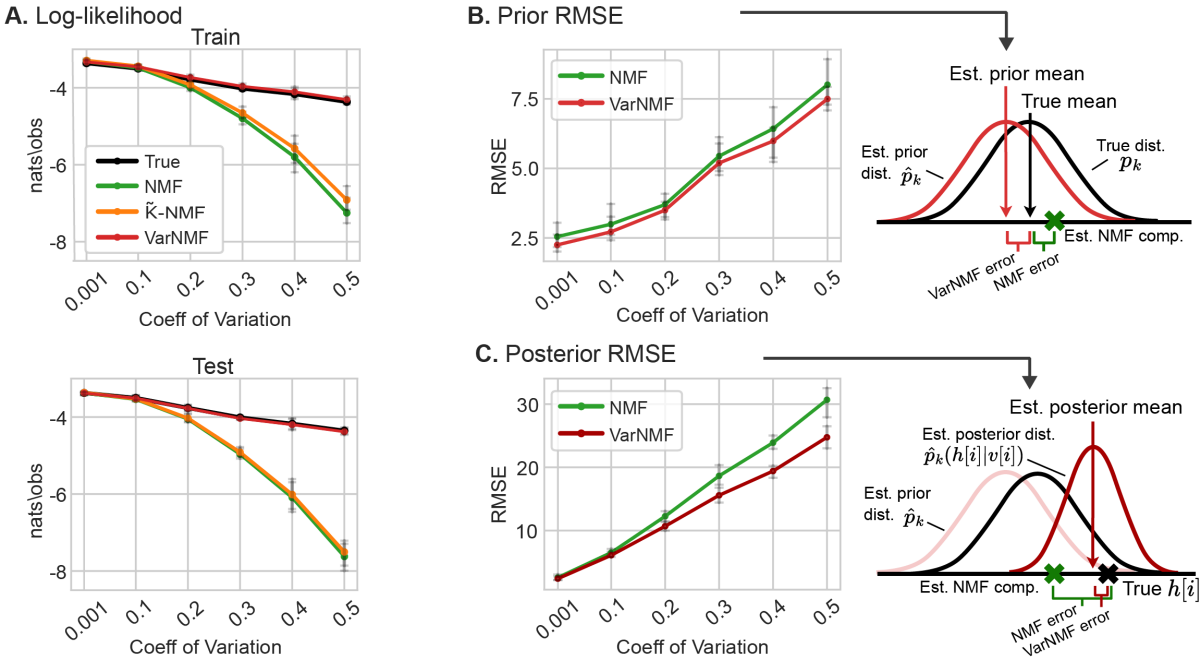


Fig. 3. Decomposing synthetic data with $K = 4$ components: A) Train and test log-likelihood of the ground truth parameters and three models - NMF, VarNMF and $\tilde{K}$-NMF (NMF with higher degrees of freedom than VarNMF), versus the coefficient of variation of the dataset. The log-likelihood values are normalized to nats/observation. B) Root mean square error (RMSE) of the mean of the distributions estimated by VarNMF (red on right panel) and the constant components estimated by NMF (green) versus the mean of the ground truth distributions (black). C) RMSE of the per-sample posterior expected signal estimated by VarNMF (red) and the constant components estimated by NMF (green) versus the ground truth $\mathcal{H}[i]$ (black). A,B,C show results for $T = 10$ runs.

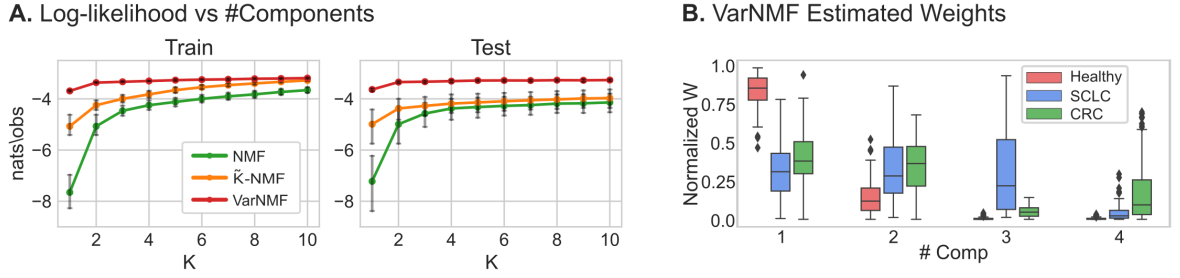


Fig. 4. Decomposing cell-free ChIP-seq data: A) Train and test log-likelihood curves for real cfChIP-seq dataset versus the number of components K used, for the three models and for $T = 5$ splits to train and test. The log-likelihood values are normalized to nats/observation. B) The VarNMF estimated weights for each component in the $K = 4$ solution (train and test shown together), aggregated by sample sub-cohort (healthy, SCLC and CRC). Weights are normalized so that each sample has a total weight of 1.

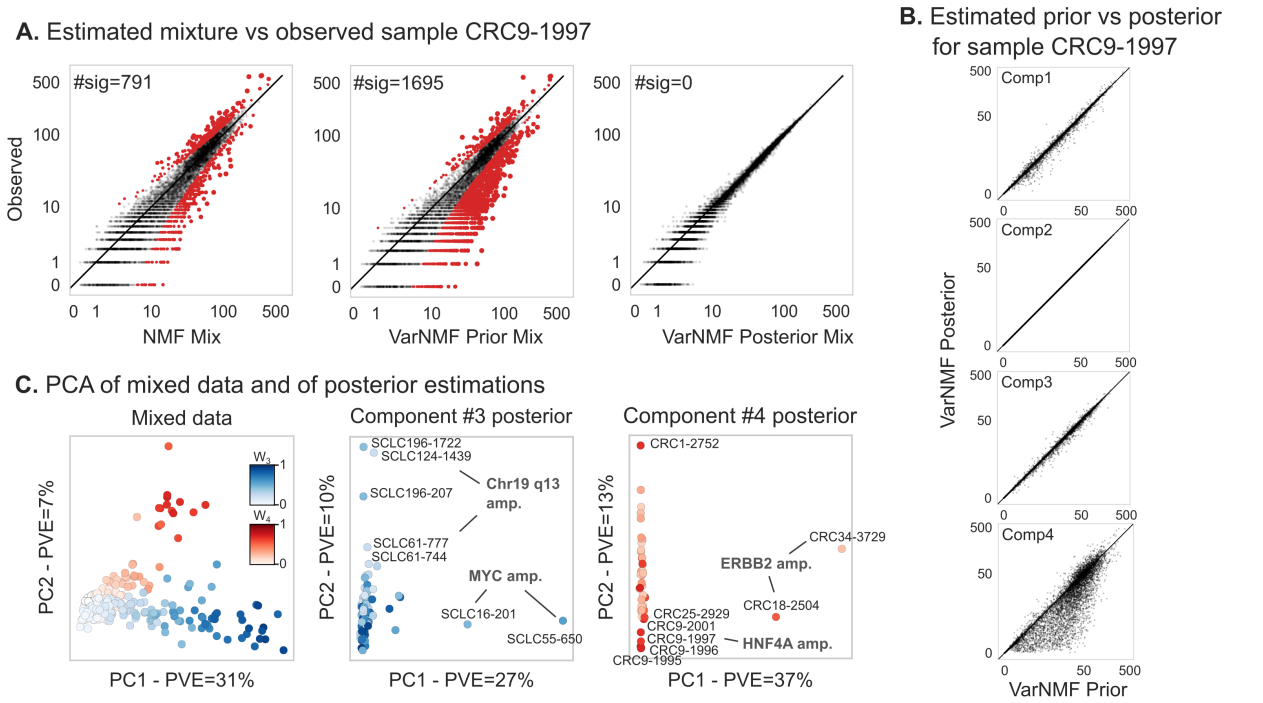


Fig. 5. Reconstructing $\mathcal{H}$ from data with the $K = 4$ solution: A) Reconstruction quality of a specific sample by NMF (left), VarNMF mean components (middle), and VarNMF posteriors (right). Each point is a gene, the x-axis shows the reconstructed value R and y-axis the observed value V. Red points are ones that are significantly different, taking into account Poisson sampling noise (q-values corrected for false discovery rate - < 0.05). B) For the same example, the prior (component mean) vs. posterior estimate per component. C) Principal Component Analysis (PCA) of the original mixed samples after normalization (Supplementary Section S4) (left) and of the component-wise posterior (middle, right). Train and test samples are shown together. Only samples with weight $> 15\%$ in the relevant component are shown.

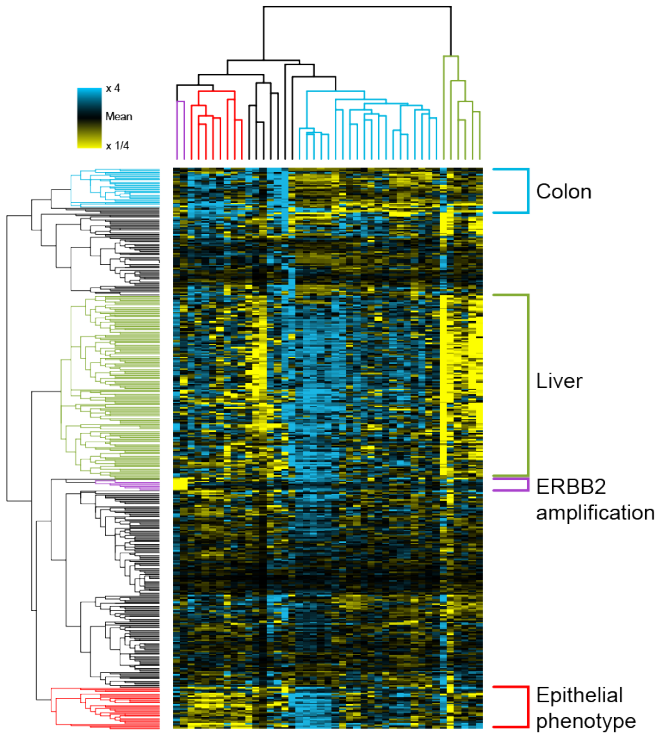


Fig. 6. Clustering of the CRC-related posterior (relative to the mean signal of each gene, with color saturation at 4-fold increase/decrease) in 383 genes that are significantly higher in the CRC-related component mean compared to the other components, and additional 6 genes that are in the ERBB2 amplification. Four sample - gene clusters were identified: ERBB2 amplification (purple), samples with an epithelial phenotype (red), samples with specific colon behavior (light blue) and samples with high liver signal (green). Only samples with weight $> 15\%$ in the CRC-related component were considered.

Supplementary Material

1. Comparison of alternative models

There are multiple differences between the various related approaches in the literature. In Figure 7, we compare multiple methods along with what we believe are the salient aspects. In addition, in Figure 8 we compare the probabilistic models of NMF, Bayesian NMF, and VarNMF. While these are similar, there crucial differences in the scope of the plates involving H.

Method	Assumes biological variation between samples in component signals	Uses prior knowledge (e.g. single-cell data)	Estimates the component proportions W	Calculates per-sample component signal	Non-negativity assumption
NMF	No. Variability between samples is only due to component composition and technical noise	Possible, but not necessary	Yes. Estimates W without prior knowledge	No	Yes
Bayesian NMF					
CIBERSORTx	Yes. Allows both group-mode estimation and "high-resolution" per-sample estimation	Yes. To construct signature matrix	Yes. Estimates W given prior knowledge of cell-types in the data	Yes. Uses a heuristic calculation based on differential genes	Yes
BayesPrism	Partially. Only accounts for differences in cell-state composition in each cell type	Yes. To construct cell-state profiles		Yes. Uses posterior estimation	Yes
BLADE	Yes. Each cell-type is modeled as a distribution	Yes. To estimate cell-type distributions			Yes. Assumes log-normal cell-type distributions
bMIND		Yes. To estimate priors on cell-type distributions	No. Assumed known or pre-estimated W		No. Assumes Normal cell-type distributions -> No non-negativity assumption on the per-sample cell-type values
TCA		No			
VarNMF	Yes. Each cell-type is modeled as a distribution	Possible, but not necessary	Yes. Estimates W without prior knowledge	Yes. Uses posterior estimation	Yes. Assumes Gamma component distributions

Fig. 7. Related works: Table highlighting the main features of VarNMF and previous works in the field. Red boxes indicate main differences. NMF: Lee and Seung [2000], Bayesian NMF: Schmidt et al. [2009], CIBERSORTx: Newman et al. [2019], BayesPrism: Chu et al. [2022], BLADE: Andrade Barbosa et al. [2021], bMIND: Wang et al. [2021], TCA: Rahmani et al. [2019]

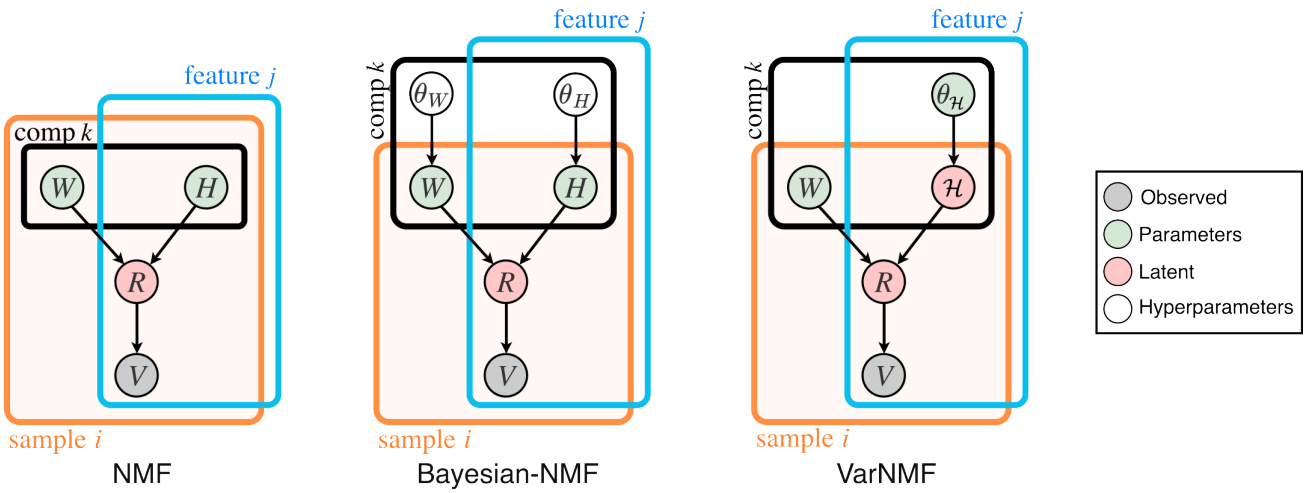


Fig. 8. Graphical plate representation [Koller and Friedman, 2009] for the NMF, Bayesian NMF and VarNMF models. Latent variables are ones that are integrated over in the model. Parameters are point estimate either by Maximum Likelihood or Maximum a Posteriori. Hyperparameters are pre-learned.

2. EM-details

Distributions Definitions and Properties

We use the following parametrizations of the Poisson, Gamma and NB distributions:

Definition 1 (Poisson Parametrization) We say that $Y \sim \text{Poisson}(\lambda)$ if

$$P(Y = k) = \frac{1}{k!} \cdot \lambda^k \cdot e^{-\lambda}$$

Lemma 1 (Poisson Properties) For $Y \sim \text{Poisson}(\lambda)$,

$$\mathbb{E}[Y] = \lambda, \quad \text{Var}[Y] = \lambda, \quad \text{CV}[Y] = 1/\sqrt{\lambda}$$

Definition 2 (Gamma Parametrization) We say that $X \sim \text{Gamma}(\alpha, \beta)$ if

$$p(X = x) = \frac{\beta^\alpha}{\Gamma(\alpha)} \cdot x^{\alpha-1} \cdot e^{-\beta x}$$

Lemma 2 (Gamma Properties)

1. For $X \sim \text{Gamma}(\alpha, \beta)$,
$$\mathbb{E}[X] = \frac{\alpha}{\beta}, \quad \text{Var}[X] = \frac{\alpha}{\beta^2}, \quad \text{CV}[X] = 1/\sqrt{\alpha}$$
2. If
$$\mathbb{E}[X] = \mu, \quad \text{Var}(X) = \sigma^2$$
then
$$X \sim \text{Gamma}\left(\alpha = \frac{\mu^2}{\sigma^2}, \beta = \frac{\mu}{\sigma^2}\right)$$
3. If
$$\mathbb{E}[X] = \mu, \quad \text{CV}[X] = \text{cv}$$
then
$$X \sim \text{Gamma}\left(\alpha = \frac{1}{\text{cv}^2}, \beta = \frac{1}{\mu \cdot \text{cv}^2}\right)$$

Definition 3 (Negative Binomial Parametrization) We say that $X \sim \text{NB}(r, p)$ if

$$P(X = k) = \binom{k + r - 1}{k} \cdot p^k \cdot (1 - p)^r$$

Lemma 3 (Negative Binomial Properties) For $X \sim \text{NB}(r, p)$,

$$\mathbb{E}[X] = \frac{pr}{1 - p}, \quad \text{Var}[X] = \frac{pr}{(1 - p)^2}, \quad \text{CV}[X] = 1/\sqrt{pr}$$

We state lemmas concerning properties of the Gamma and Poisson distributions [Wackerly et al., 2014]:

Lemma 4 If $X \sim \text{Gamma}(\alpha, \beta), \gamma > 0$ then $\gamma \cdot X \sim \text{Gamma}\left(\alpha, \frac{\beta}{\gamma}\right)$.

Lemma 5 If $Y \mid X \sim \text{Poisson}(X)$ and $X \sim \text{Gamma}(\alpha, \beta)$ then $Y \sim \text{NB}\left(\alpha, \frac{1}{1+\beta}\right)$.

Lemma 6 If $Y \mid X \sim \text{Poisson}(\gamma X), \gamma > 0$ and $X \sim \text{Gamma}(\alpha, \beta)$ then

$$(X|Y = t) \sim \text{Gamma}(\alpha + t, \beta + \gamma)$$

Lemma 7 If $X \sim \text{Gamma}(\alpha, \beta)$ then $\mathbb{E}[\log(X)] = \psi(\alpha) - \log\beta$, where ψ is the digamma function.

Log-Likelihood

First, we present a scheme that allow us to calculate the log-likelihood of a set of parameters $\theta = W, A, B$. We can calculate the log-likelihood for each observation separately:

$$\ell(\theta; V) \stackrel{\text{def}}{=} \log \sum_{i,j} p\left(V[i]_j \mid W[i], A_{:,j}, B_{:,j}\right) \tag{21}$$

To simplify notation, we set i, j and define

$$v = V[i]_j, y = \mathcal{Y}[i]_{:,j}, h = \mathcal{H}[i]_{:,j}, a = A_{:,j}, b = B_{:,j}, w = W[i], \theta = (w, a, b) \tag{22}$$

and look to calculate $\log p(v \mid \theta)$. As mentioned above, with the addition of the random variables Y to the model, this require a K-dimensional summation, and will be done using dynamic programming.

It is sufficient to calculate the joint distribution of (y_k, v) for some k, since:

$$p(v \mid \theta) = \sum_{d=0}^{v} p(y_k = d, v \mid \theta) \tag{23}$$

Moreover, we can write this joint as

$$p(y_k = d, v \mid \theta) = p(y_k = d \mid \theta) \cdot p(v \mid y_k = d, \theta) \tag{24}$$

The first factor can be directly calculated using the following lemma:

Lemma 8

$$\begin{cases} (y_k \mid w_k, h_k) \sim \text{Poisson}(w_k \cdot h_k) \\ (h_k \mid a_k, b_k) \sim \text{Gamma}(a_k, b_k) \end{cases} \quad \Rightarrow \quad (y_k \mid w_k, a_k, b_k) \sim \text{NB}\left(a_k, \frac{w_k}{w_k + b_k}\right)$$

Proof From Lemma 4,

$$(h_k \mid a_k, b_k) \sim \text{Gamma}(a_k, b_k) \Rightarrow \quad (w_k \cdot h_k \mid w_k, a_k, b_k) \sim \text{Gamma}\left(a_k, \frac{b_k}{w_k}\right)$$

and from Lemma 5 we get

$$\begin{cases} (y_k \mid w_k, h_k) \sim \text{Poisson}(w_k \cdot h_k) \\ (w_k \cdot h_k \mid w_k, a_k, b_k) \sim \text{Gamma}\left(a_k, \frac{b_k}{w_k}\right) \end{cases}$$

and therefore

$$(y_k \mid w_k, a_k, b_k) \sim \text{NB}\left(a_k, \frac{1}{1 + \frac{b_k}{w_k}}\right) = \text{NB}\left(a_k, \frac{w_k}{w_k + b_k}\right)$$

□

As for the second factor, $v = \sum_k y_k$ and v is discrete, therefore

$$\forall_{k,d},\ p(v \mid y_k = d, \theta) = p\left(\sum_{l \neq k} y_l = v - d \mid \theta\right) \tag{25}$$

and that can be calculated using dynamic programming:

For simplicity, we denote $p = p_\theta$ and $p_k = \text{NB}\left(a_k, \frac{w_k}{w_k + b_k}\right)$ (the distribution of y_k). We define two random variables:

$$X_s \stackrel{\text{def}}{=} \sum_{l=1}^{s} y_l \qquad\qquad Z_s \stackrel{\text{def}}{=} \sum_{l=s+1}^{K} y_l \tag{26}$$

and two tables

$$F[s, n] \stackrel{\text{def}}{=} p(X_s = n), \quad s = 1, ..., K-1,\ n = 0, ..., v \tag{27}$$

$$B[s, n] \stackrel{\text{def}}{=} p(Z_s = n), \quad s = 1, ..., K-1,\ n = 0, ..., v \tag{28}$$

Using law of total probability, we get the two following recursive formulas:

$$F[s, n] = \sum_{d=0}^{n} F[s-1, d] \cdot p_s(n - d) \tag{29}$$

$$B[s, n] = \sum_{d=0}^{n} B[s+1, d] \cdot p_{s+1}(n - d) \tag{30}$$

Now we have two dynamic programming tasks:

1. The Forward task of filling F by columns, from the initialization:

$$\forall_{0\leq n\leq v},\ F\left[1,n\right]=p_1\left(n\right)$$

and forward according to Eq. 29.

2. The Backward task of filling B by columns, with initial values for the last column:

$$\forall_{0\leq n\leq v},\ B\left[K-1,n\right]=p_K\left(n\right)$$

and going backward with Eq. 30.

Finally, in order to find the required probability we use these two tables to fill a new table

$$P\left[k,d\right]\overset{\text{def}}{=}p\left(v\mid y_k=d\right),\quad k=1,...,K,\ d=0,...,v \tag{31}$$

using the recursive formula

$$\begin{aligned}\forall_{1<k<K},\forall_d,\quad P\left[k,d\right]&=p\left(\sum_{l\neq k}y_l=v-d\right)\\&=p\left(\sum_{l=0}^{k-1}y_l+\sum_{l=k+1}^{K}y_l=v-d\right)\\&=p\left(X_{k-1}+Z_k=v-d\right)\\&=\sum_{n=0}^{v-d}p\left(Z_k=v-d-n\right)\cdot p\left(X_{k-1}=n\right)\\&=\sum_{n=0}^{v-d}B\left[k,v-d-n\right]\cdot F\left[k-1,n\right]\end{aligned} \tag{32}$$

and the initial condition

$$\forall_d,\ P\left[K,d\right]=F\left[K-1,v-d\right] \tag{33}$$

Complete-Data Log-Likelihood

To apply the EM procedure to our model, we need to calculate the expectations of the following sufficient statistics to get the ESS in the E-step:

$$G[i]_k=\sum_j\mathcal{Y}[i]_{k,j}\,,\qquad T[i]_k=\sum_j\mathcal{H}[i]_{k,j} \tag{34}$$

$$S^0=N,\qquad S^1_{k,j}=\sum_i\mathcal{H}[i]_{k,j}\,,\qquad S^{\log}_{k,j}=\sum_i\log\mathcal{H}[i]_{k,j} \tag{35}$$

and to maximize the expectation of the complete-data log-likelihood in the M-step:

$$\theta^{(t+1)}=\arg\max_\theta\ \mathbb{E}_{p\left(\mathcal{Y},\mathcal{H}\mid V,\theta^{(t)}\right)}\left[\ell^*\left(\theta;V,\mathcal{Y},\mathcal{H}\right)\right] \tag{36}$$

Here, we provide the full mathematical details of these steps:

M - Step

As mentioned above, the maximization of Eq. 36 can be achieved by separately maximizing the log-likelihood functions $\ell^{\mathcal{Y}*}_{i,k}$ and $\ell^{\mathcal{H}*}_{k,j}$, only with the ESS calculated in the E-step replacing the actual sufficient statistics.

Specifically, $\mathcal{Y}[i]_{k,j}\sim\text{Poisson}\left(W[i]_k\cdot\mathcal{H}[i]_{k,j}\right)$, therefore for each $w=W[i]_k$,

$$\begin{aligned}\ell^{\mathcal{Y}*}_{i,k}\left(w\right)&\overset{\text{def}}{=}\log P\left(\mathcal{Y}[i]_k\mid w,\mathcal{H}[i]_k\right)\\&=\text{const}+G[i]_k\cdot\log w-T[i]_k\cdot w\end{aligned} \tag{37}$$

Additionally, $\mathcal{H}[i]_{k,j}\sim\text{Gamma}\left(A_{k,j},B_{k,j}\right)$ and thus for each $a=A_{k,j},b=B_{k,j}$,

$$\begin{aligned}\ell^{\mathcal{H}*}_{k,j}\left(a,b\right)&\overset{\text{def}}{=}\log p\left(\mathcal{H}[1]_{k,j},...,\mathcal{H}[N]_{k,j}\mid a,b\right)\\&=\left[a\log b-\log\Gamma\left(a\right)\right]\cdot S^0-b\cdot S^1_{k,j}+\left(a-1\right)\cdot S^{\log}_{k,j}\end{aligned} \tag{38}$$

Now, given the ESS, we maximize both functions separately. Set i,j,k, and

$$G=G[i]_{k,j}\,,\qquad T=T[i]_{k,j}\,,\qquad S^1=S^1_{k,j},\qquad S^{\log}=S^{\log}_{k,j} \tag{39}$$

Then, differentiating each function and setting the gradient to 0,

1. For $\ell^{\mathcal{Y}*}$ we get that $\hat{w} = \frac{G}{T}$.
2. For $\ell^{\mathcal{H}*}$, we get the following system,

$$\begin{cases} \psi(a) - \frac{S^{\log}}{S^0} - \log a + \log \frac{S^1}{S^0} = 0 \\ b = a \cdot \frac{S^0}{S^1} \end{cases}$$

where $\psi(x) = \frac{(\Gamma(x))'}{\Gamma(x)}$ is the digamma function. The first equation can be solved by finding a root using the Newton Raphson algorithm.

E - Step

As described above, given $\theta^{(t)} = W, A, B$, we are required to calculate the expectation of the sufficient statistics from Eq. 34 and Eq. 35. To simplify notation, and as this process is done separately for each feature j in each sample i, we set i, j and define

$$v = V[i]_j, y = \mathcal{Y}[i]_{:,j}, h = \mathcal{H}[i]_{:,j}, a = A_{:,j}, b = B_{:,j}, w = W[i], \theta^{(t)} = (w, a, b) \tag{40}$$

From linearity of expectation, it is enough to calculate for each k,

$$\text{(i)}\ \mathbb{E}\left[y_k \mid v, \theta^{(t)}\right] \qquad \text{(ii)}\ \mathbb{E}\left[h_k \mid v, \theta^{(t)}\right] \qquad \text{(iii)}\ \mathbb{E}\left[\log h_k \mid v, \theta^{(t)}\right] \tag{41}$$

We start by calculating the posterior distribution of y_k, $p(y_k = d \mid v, \theta)$ for each $0 \leq d \leq v$. This can be done using the joint probability $p(y_k = d, v \mid \theta)$ calculated with the dynamic programming scheme from 9.2, and the fact that

$$\begin{aligned} p(y_k = d \mid v, \theta) &= \frac{p(y_k = d, v \mid \theta)}{p(v \mid \theta)} \\ &= \frac{p(y_k = d, v \mid \theta)}{\sum_{l=0}^{v} p(y_k = l, v \mid \theta)} \end{aligned} \tag{42}$$

Given the posteriors, $\{p(y_k = d \mid v, \theta)\}_{d=1}^{v}$, we can now calculate:

1. $\mathbb{E}[y_k \mid v, \theta] = \sum_{d=0}^{v} d \cdot p(y_k = d \mid v, \theta)$.
2. $\mathbb{E}[h_k \mid v, \theta] = \frac{a_k + \mathbb{E}[y_k \mid v, \theta]}{b_k + w_k}$ (Lemma 6).
3. $\mathbb{E}[\log h_k \mid v, \theta] = -\log(b_k + w_k) + \sum_{d=0}^{v} p(y_k = d \mid v, \theta) \cdot \psi(a_k + d)$ (Lemmas 6, 7).

Calculating the posterior expected signal

Given a prior source distribution (for some feature j) $p_{k,j} = \text{Gamma}(A_{k,j}, B_{k,j})$ estimated from training data, we want to estimate the source contribution to a particular sample i, $\mathcal{H}[i]_{k,j}$. We use the expectation of the posterior distribution of H according to the estimated prior distribution:

$$\hat{p}[i]_{k,j}(h) = p\big(\mathcal{H}[i]_{k,j} = h \mid V[i], A_{k,j}, B_{k,j}\big) \tag{43}$$

Specifically, we use $\mathbb{E}\left[\mathcal{H}[i]_{k,j} \mid V[i], A_{k,j}, B_{k,j}\right]$ which we calculated in the E-step (Eq. 41(ii)).

Alternating EM

As mentioned above, the E-step of the presented EM procedure is computationally heavy, and cannot be applied on a real dataset with a typical number of a few thousand features (genes). There are many possible solutions to this problem in literature. Here, we decided to use an alternating version of EM [Neal and Hinton, 1998], which also alternates between optimizing W and A,B (similar to the NMF multiplicative update rule). The algorithm steps are detailed in Fig. 9.

We start in step 1 with an NMF solution for W^{NMF} (which can be calculated fairly quickly, using the NMF algorithm mentioned above). For step 2, we use a variant of the EM procedure we call EMcW, to estimate (A, B) while keeping W^{NMF} constant. This process requires the calculation of the ESS of $S^1, S^{\log}$ for the E-step, and the maximization of (A, B) for the M-step. We note that given W^{NMF}, the parameters (A, B) and the ESS of $S^1, S^{\log}$ are independent between features. Thus we can divide the features into batches and parallelize the EMcW procedure entirely.

Next, we use the resulting (A, B) parameters to determine which features are component-specific: We take $\left(\tilde{A}, \tilde{B}\right)$ to be the resulting (A, B) for the $\tilde{M} = 100 \cdot K$ features with the highest difference between the mean signal of different components (normalized by mean signal of the features). We then use these features in step 3 to adjust the starting point W^{NMF}, by running the EM algorithm to estimate the mixing weights W while keeping $\left(\tilde{A}, \tilde{B}\right)$ constant (EMcAB variant). This requires the calculation of the ESS of G, T in the E-step, and the maximization of W in the M-step, and results in $\hat{W}$ that is adjusted for component variation. Lastly, in step 4 we readjust the component parameters $\left(\hat{A}, \hat{B}\right)$ using the EMcW variant with the constant $\hat{W}$.

In our case we stop here and return the resulting $\hat{W}$ and $\left(\hat{A}, \hat{B}\right)$ from the last step. However, this alternation can clearly be repeated several times (steps 3 and 4). Since the EMcW variant is parallelized by batches and the EMcAB only uses $\tilde{M} = 100 \cdot K$ features, each iteration is much more efficient in both time and memory than running the regular EM procedure over a large number of features.

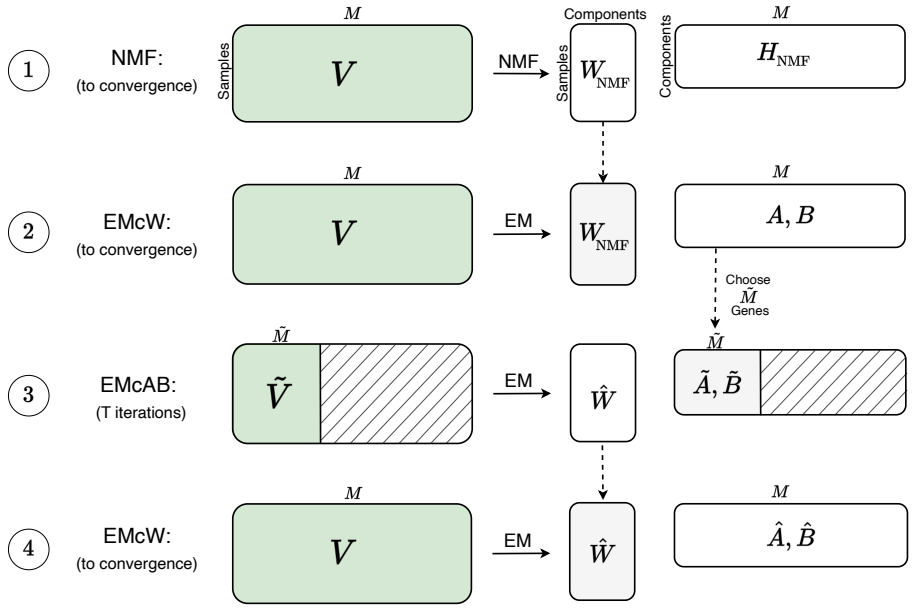


Fig. 9. The alternating EM procedure, which attempts to find an optimal solution for the VarNMF model while avoiding high computational cost. It uses the NMF solution as a starting point (step 1), and apply EM over one of the parameter matrices W or (A, B) at a time (steps 2 and 3). This process is then repeated once in step 4 for W, and can be applied iteratively. Green indicates a data matrix (or a subset of one) and grey indicates a fixed parameter matrix (that results from the previous step). $T = 250$ for all real-data runs.

Normalization

A known limitation of the NMF model is that it is not identifiable: Say W, H are an NMF solution for dataset V. Then, for every invertible matrix J s.t.

$$W^{\text{new}} \stackrel{\text{def}}{=} W \cdot J \geq 0 \qquad\qquad H^{\text{new}} \stackrel{\text{def}}{=} J^{-1} \cdot H \geq 0 \tag{44}$$

we have

$$W^{\text{new}} \cdot H^{\text{new}} = (W \cdot J) \cdot \left(J^{-1} \cdot H\right) = W \cdot H \tag{45}$$

As the Poisson log-likelihood NMF depends only on the reconstruction $R = WH$ (Eq. 2), the cost of the two solutions will be identical, and therefore the KL-NMF problem is not identifiable.

This is important in a few ways: First, measuring the quality of a solution, we want to compare the solution directly to ground truth parameters (if exist). We also compare two different solutions from different models. More crucially, we want, for example, to be able to infer component patterns and use them to characterize states in the system (e.g. cell-types). If the components are not unique in a way that changes the real-life interpretation of them, it significantly harms the interpretability of the model.

We start by considering permutation and non-negative normalization matrices (diagonal matrices with non-negative elements). Since such matrices are invertible and their inverse is also non-negative [Plemmons and Cline, 1972], multiplying an NMF solution by them (as detailed in Eq. 45) will result in an equivalent solution. However, we can easily normalize NMF solutions to alleviate ambiguity of normalization and permutation and get *essentially unique* solutions that are comparable between models and runs: We use a reference solution to determine the order of the components (for synthetic data - according to the ground truth, for real data - some predefined order), and match the components order of the new solution using linear sum assignment. We then reorder the components in the solutions by multiplying H and W with a corresponding permutation matrix. For normalization, we normalize each component by a factor d_k by multiplying H and W with a corresponding normalization matrix. This results in an equivalent solution.

For VarNMF solutions we use the same process, except for the normalization of the components that requires a different treatment: We want to scale the distributions of $\mathcal{H}$ by some factors $d_1, ..., d_K$. Following Lemma 4, we can simply scale the parameters B_k by the inverse d_k^{-1}, or equivalently multiply B by the inverse of the corresponding normalization matrix. This will result in an equivalent solution if we normalize W accordingly.

The choice of normalization depends on the data. For synthetic data, we normalize each component to have the same mean value (over the features) as the matching ground truth component. For real data, we use a healthy reference — a vector that represent the typical signal of healthy samples (obtained from [Sadeh et al., 2021]). We choose a group of house-keeping genes, that are known to have high signal and low variation between cell-types and tissues (Sadeh et al.). We then normalize each component to have the same median value these house-keeping genes as the healthy reference. This normalization procedure is meant to anchor all components' house-keeping genes (that have similar behavior in different cell-types) to a similar location.

Other matrices for which Eq. 44 holds can be used to scale only NMF and not VarNMF solutions, as it will cause different components to become correlated, thus the scaled solution will be distinct from the original solution (in which all features are

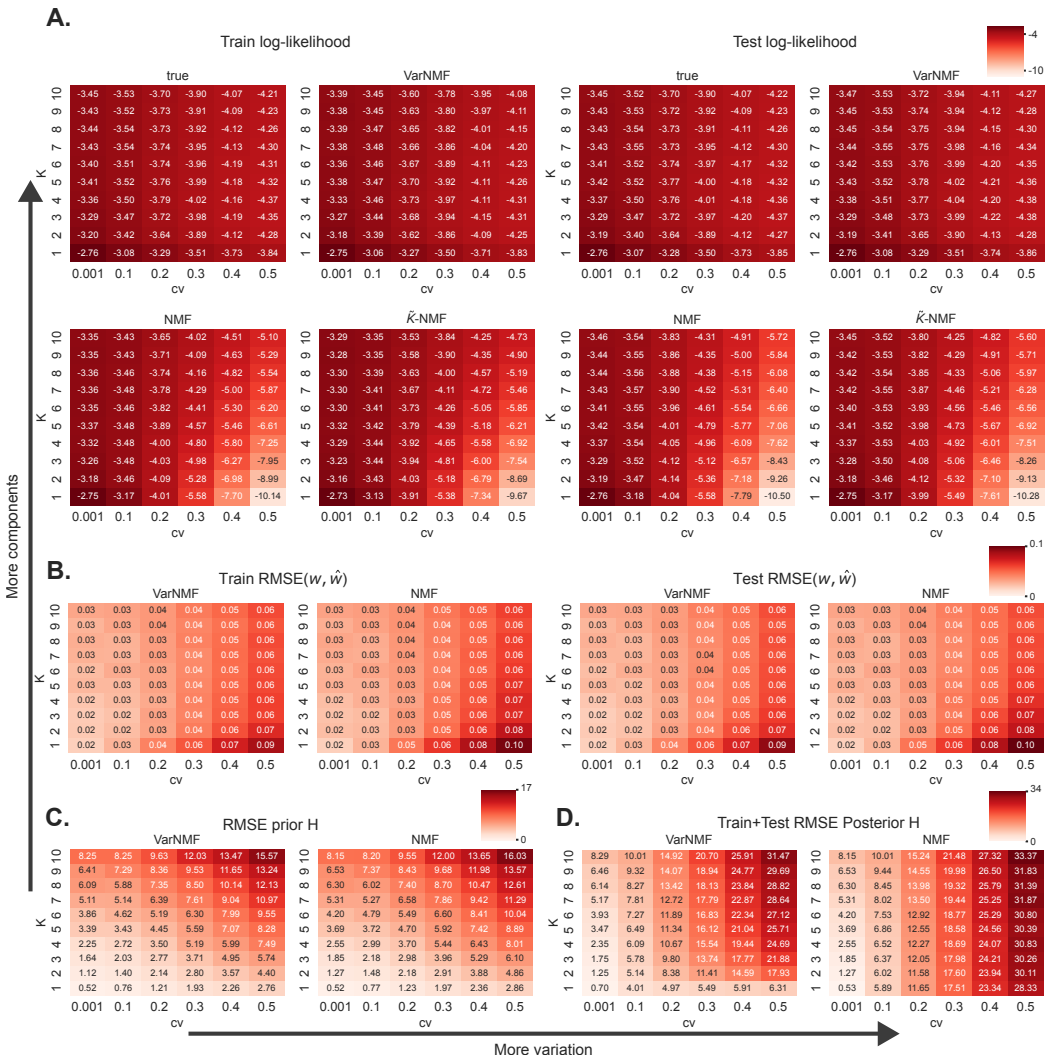


Fig. 10. Decomposing synthetic data with $K = 1, \ldots, 10$ components: A) Train and test log-likelihood of the ground truth parameters and three models - NMF, VarNMF and $\tilde{K}$-NMF (NMF with higher degrees of freedom than VarNMF), versus the coefficient of variation of the dataset and versus K. The log-likelihood values are normalized to nats/observation. B) Root mean square error (RMSE) of the the train and test weights estimated by VarNMF and NMF versus the ground truth weights. C) RMSE of the mean of the distributions estimated by VarNMF and the constant components estimated by NMF versus the mean of the ground truth distributions. D) RMSE the per-sample posterior expected signal estimated by VarNMF and the constant components estimated by NMF versus the ground truth $\mathcal{H}[i]$. The shown values are the mean over $T = 10$ runs.

independent). These matrices can sometimes exists, and the conditions for essential uniqueness of the NMF solutions are complicated [Donoho and Stodden, 2003]. Fortunately, in practice, following some basic sparseness properties, it is likely that NMF will have an essentially unique solution [Huang et al., 2013]. In our case, these conditions usually hold, and we will assume the NMF solution (normalized for permutations and normalization matrices) is unique.

Generalization

The train log-likelihood measure evaluates the ability of an algorithm to fit a dataset, and is a popular measure in the world of probabilistic modeling and unsupervised learning (which includes NMF research). However, in our case and in many other situations, the overall goal is extract an accurate distribution of features (e.g. genes). While the mixing weights are specific to each sample, the learned distributions are general and should fit all samples, including unseen ones. Therefore, we want measure the generalization abilities of our model on a new dataset sampled using the same distribution.

To test the generalization abilities of VarNMF, we use the learned distributions to learn a new W^{test} for a test dataset V^{test} by applying EMcAB (Appendix 9.5). Similarly, for NMF, we use the learned constant components H to learn W^{test} by applying the NMF Multiplicative update rule [Lee and Seung, 2000] on V^{test} and keeping H constant. In both cases we report the test log-likelihood to be the log-likelihood for the test data with the train distributions or components and the test mixing weights.

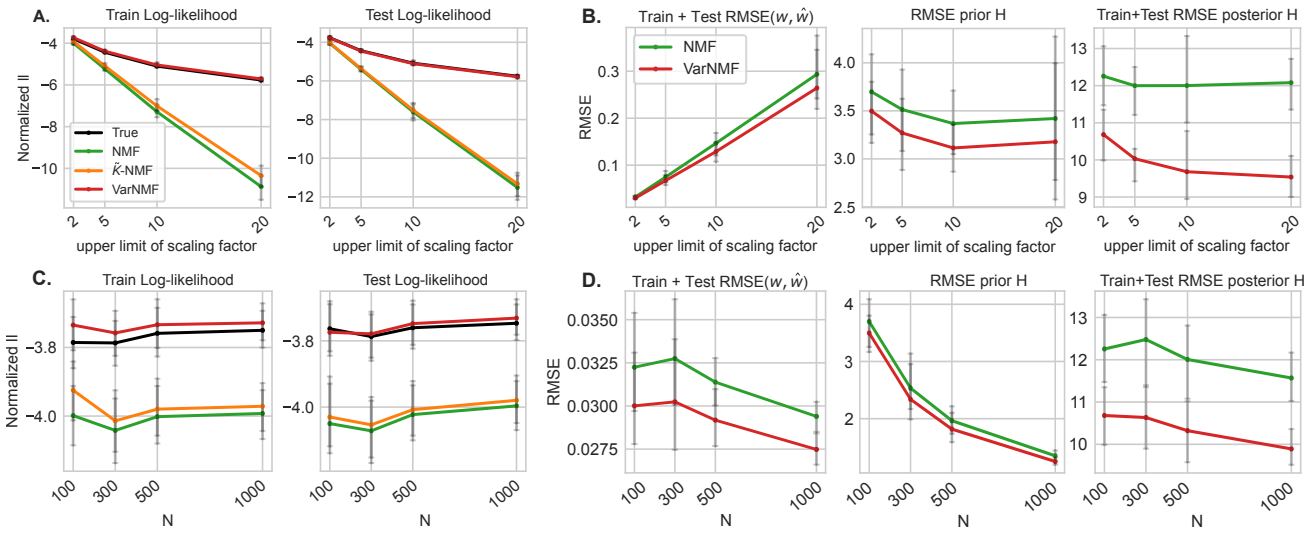


Fig. 11. Robustness to sampling noise (top) and N (bottom) - Decomposing synthetic data with increasing sampling noise and increasing number of samples N: A,B) Train and test log-likelihood of the ground truth parameters and three models - NMF, VarNMF and $\tilde{K}$-NMF, versus A) the upper limit of the scaling factors distribution and versus B) the number of samples N. The log-likelihood values are normalized to nats/observation. C,D) RMSE of the different parameters versus C) the upper limit of the scaling factors distribution and versus D) the number of samples N. Left - RMSE of the train and test weights estimated by VarNMF and NMF versus the ground truth weights. Middle - RMSE of the mean of the distributions estimated by VarNMF and the constant components estimated by NMF versus the mean of the ground truth distributions. Right - RMSE the per-sample posterior expected signal estimated by VarNMF and the constant components estimated by NMF versus the ground truth $\mathcal{H}[i]$. The shown values are the mean over $T = 10$ runs of datasets with $K = 4$ components and coefficient of variation cv $= 0.2$

3. Synthetic-Data

Results for different K values

We examine the results NMF and VarNMF on synthetic datasets with increasing source variation and increasing number of sources (Section 4.1 for more details). The models are given the correct number of source K. In Fig. 10A we observe a decline in NMF train and test log-likelihood results as the levels of variation in the dataset (controlled by the coefficient of variation of the ground truth source distributions) increase. This is true for all values of K (#Components in the dataset) but is most apparent for small values of K. We conclude that for that VarNMF better captures the datasets distribution in the presence of high component variation.

Next, we examine the learned parameters of each model against the ground truth. Starting with the learned weights of NMF and VarNMF against the ground truth (Fig. 10B), the two models have almost identical values, with performances decreasing with the level of variation. As for learned components against the ground truth, the VarNMF component means are closer to the ground truth means than NMF's estimates (Fig. 10C) but generally the trends look similar: The RMSE values increase with the levels of variation and with the number of components. This suggests that component variation between samples increases the complexity of the data and hinder both algorithms effort to extract the ground truth weights and mean of the sources distributions. However, comparing the ground truth per-sample contribution of source k, $\mathcal{H}[i]_k$, to the VarNMF posterior expected signal versus NMF constant components (Fig. 10D), the VarNMF RMSE is similar to the NMF values for datasets with no variation (cv$= 0.001$), but outperform the NMF solution when this variation increases, and NMF perform poorly for every value of K.

Robustness to sampling noise and number of samples

In Fig 11 we examine the effect of increasing sampling noise on the performances of NMF and VarNMF. This is done by increasing the limits of the uniform distribution from which we sample the per-sample scaling factors $\lambda[i]$ (Section 4.1 for more details). Specifically, applying VarNMF and NMF to datasets with scaling factors sampled from $\mathcal{U}\left(\left[\frac{l}{2}, l\right]\right)$ for $l = 2, 5, 10, 20$, results in an expected drop in performances. However, the NMF performances drop significantly more than those of VarNMF, suggesting that the advantage of VarNMF is robust to sampling noise. Additionally, when increasing the number of samples N, we get better estimations of the ground truth parameters, but also a consistent advantage of VarNMF over NMF (Fig 11).

Results with wrong K values

A worry in practice is that we would not have access to the correct number of component. To examine the sensitivity of the reconstruction to wrong values of K, we created synthetic datasets with trueK $= 4$ sources and increasing levels of source variation (coefficient of variation $= 0.001, 0.2, 0.5$, as in Section 4.1). We then decomposed these datasets using $K = 1, \ldots, 10$ components with NMF, VarNMF, and also NMF with additional components to compensate for difference in number of free parameters, Fig 12. When K is small, All models suffer from big gaps in log-likelihood scores. These gaps decrease as K increases, and plateau at $K = 4 =$ trueK, as expected. However, in cases with higher source variation, both NMF models have substantial gap from the ground truth, even for larger values of K. In contrast, VarNMF does not suffer from this gap and achieves the true log-likelihood

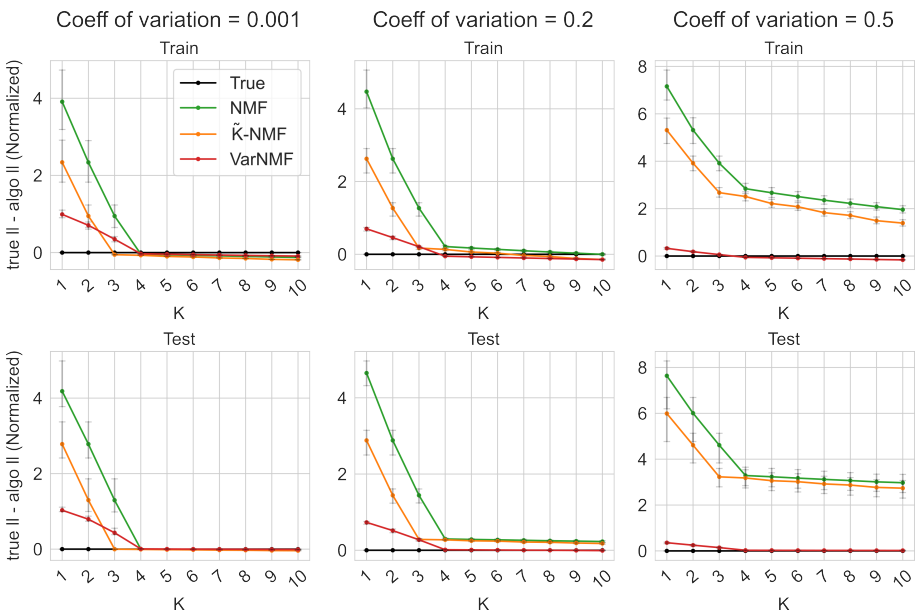


Fig. 12. Decomposing synthetic data with trueK $= 4$ sources using $K = 1, \ldots, 10$ components: Train (top) and test (bottom) log-likelihood of the ground truth parameters and three models - NMF, VarNMF and $\tilde{K}$-NMF (NMF with higher degrees of freedom than VarNMF, i.e. higher number of components), versus the number of components used in the decomposition, for three coefficient of variation values. The log-likelihood values are normalized to nats/observation. The shown values are the mean over $T = 10$ runs. Shown are the differences from true log-likelihood on the same data, to account for likelihood differences that arise from the data sampling process.

using $K \geq 4$ components. This suggests that in settings with high source variation, VarNMF offers a better fit for the data than NMF, even when decomposing with the wrong number of components.

4. cfChIP-data

We collected a dataset of cfChIP-seq samples from Sadeh et al. [2021] and Fialkoff et al. [2022]. This data includes plasma samples of 80 healthy subjects, 139 small-cell lung cancer (SCLC) patients, and 86 colorectal cancer (CRC) patients. This data has two representations.

- Normalized reads - gene values after normalization, According to Sadeh et al. [2021] the normalization ensure that the samples agree on a set of reference genes.
- Raw counts - gene values are the observed count in each sample.

We used the normalized reads representation for selecting genes (below) and for PCA analysis (Fig. 5). For the actual data processing we used raw counts, and let the model fit scaling parameter per sample.

Choosing dataset features

To choose relevant features we first excluded genes that are positioned on chromosomes X or Y (sex-specific genes) and putative genes (ORFs without a name, pseudogenes and such).

For each gene we computed several statistics across the entire dataset — mean μ_g, variance σ_g^2, coefficient of variation $\eta_g = \sigma_g/\mu_g$, number of times they are above 0 n_g, and maximal value m_g. Based on these we defined three groups of genes:

- Housekeeping-like genes [6029 genes] – Genes with high levels ($\mu_g > 50$) and low variability ($\eta_g < 0.5$)
- Variable genes [5119 genes] – genes that do not appear in the first group and have high variation ($\eta_g > 0.25$), expression in more than 50 samples ($n_g > 50$), and some observations above a threshold ($m_g > 10$).
- Remaining non-excluded genes [5764 genes]

We reasoned that the housekeeping-like genes provide stability and anchor the estimation of values. The variable genes provide a chance to identify interesting phenomena. Thus we randomly selected 5000 variable genes, and added 1000 randomly selected genes from each of the two other groups.

Testing gene-lists versus curated databases

To choose genes that differ in one component k from the rest, we first discard genes that have a value lower than 10 in the specific component (or in the mean of the component, in the case of VarNMF). We then calculate the median value across all components

Table 1. Enrichr results for NMF solution, $K = 4$

Component	Database	Term	Overlap	Adj. p-value
1	BioPlanet 2019	Hemostasis pathway	12/468	3.52e-09
37 genes	BioPlanet 2019	Platelet activation signaling and aggregation	8/205	2.19e-07
2	ARCHS4 Tissues	PERIPHERAL BLOOD	165/2316	8.93e-75
297 genes	ARCHS4 Tissues	NEUTROPHIL	140/2316	9.49e-52
	ARCHS4 Tissues	MACROPHAGE	137/2316	1.88e-48
3	Cancer Cell Line Enc.	CORL279 LUNG	116/465	1.10e-84
464 genes	Cancer Cell Line Enc.	CORL24 LUNG	117/569	1.98e-75
	Cancer Cell Line Enc.	NCIH446 LUNG	94/327	3.20e-74
4	Cancer Cell Line Enc.	SNU283 LARGE INTESTINE	58/189	5.99e-55
329 genes	Cancer Cell Line Enc.	CL40 LARGE INTESTINEE	55/210	5.63e-48
	Cancer Cell Line Enc.	SW1463 LARGE INTESTINE	49/177	6.22e-44

Table 2. Enrichr results for VarNMF solution, $K = 4$

Component	Database	Term	Overlap	Adj. p-value
1	Reactome 2022	Hemostasis	10/576	3.11e-07
28 genes	BioPlanet 2019	Hemostasis pathway	11/468	2.12e-09
	BioPlanet 2019	Platelet Activation, Signaling and Aggregation	8/205	1.91e-08
2	ARCHS4 Tissues	PERIPHERAL BLOOD	86/2316	1.24e-52
118 genes	ARCHS4 Tissues	NEUTROPHIL	76/2316	1.45e-40
	ARCHS4 Tissues	GRANULOCYTE	71/2316	3.56e-35
	ARCHS4 Tissues	MACROPHAGE	66/2316	4.05e-30
3	Cancer Cell Line Enc.	CORL279 LUNG	155/465	8.09e-108
700 genes	Cancer Cell Line Enc.	NCIH446 LUNG	122/327	1.50e-90
	Cancer Cell Line Enc.	CORL24 LUNG	150/569	6.80e-88
4	Cancer Cell Line Enc.	JHH5 LIVER	89/245	5.58e-88
383 genes	Cancer Cell Line Enc.	C3A LIVER	95/387	3.12e-76
	Cancer Cell Line Enc.	SNU283 LARGE INTESTINE	53/189	9.57e-45
	Cancer Cell Line Enc.	SW1463 LARGE INTESTINE	51/177	1.03e-43

for each gene, and sort the genes in descending order based on the fold-change increase observed in component k relative to the calculated median. Genes with less than a 2-fold increase are excluded.

Next, we test whether these genes are significantly over-represented in curated genes-lists associated with a specific tissue or cell-type (using the Enrichr tool, Chen et al. [2013] for details). The top results for each component of the NMF and VarNMF solutions with $K = 4$ components are presented in Tables 1, 2. For the "healthy" components we used reference data from human tissues and cell-types (*Reactome 2022*, *BioPlanet 2019* and *ARCHS4 Tissues*). For the disease-associated components, we made use of the *Cancer Cell-Line Encyclopedia* (CCLE) which contains data from cells of various cancers.

Results for the first three components are similar between the two models. The first two components' genes are strongly enriched for Platelets and Neutrophils, which are the two main sources of cell-free DNA in healthy samples [Sadeh et al., 2021]. The second component is also enriched for Macrophages that differentiate from Monocytes which are also found in high concentrations in cell-free DNA from healthy samples. The third component (with non zeros weights mostly in SCLC patients) is enriched specifically for SCLC-derived cell-lines in both the NMF and VarNMF solutions. This indicates that this component indeed represents the tumor derived cell-free DNA in the SCLC patients plasma. The forth component (with non zeros weights mostly in CRC patients) displays different associations between the two models: The NMF solution is enriched exclusively for colon-derived cell-lines, aligning with the CRC patients diagnosis. On the other hand, the VarNMF solution is enriched for both liver and colon derived cell-lines. The former enrichment may reflect liver metastasis which exist in many of the CRC patients in this cohort.

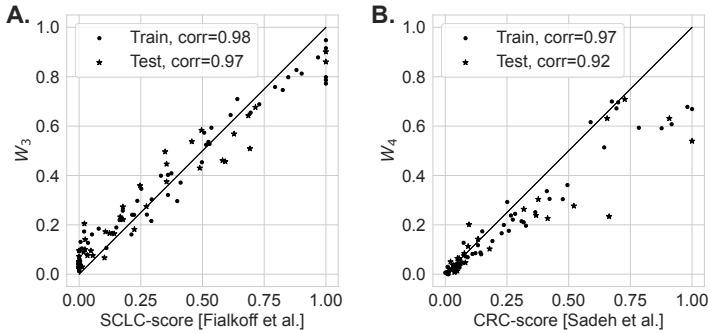


Fig. 13. Correlations of the weights estimated by VarNMF for components #3 (A) and #4 (B) versus supervised estimations of disease load [Fialkoff et al., 2022, Sadeh et al., 2021]

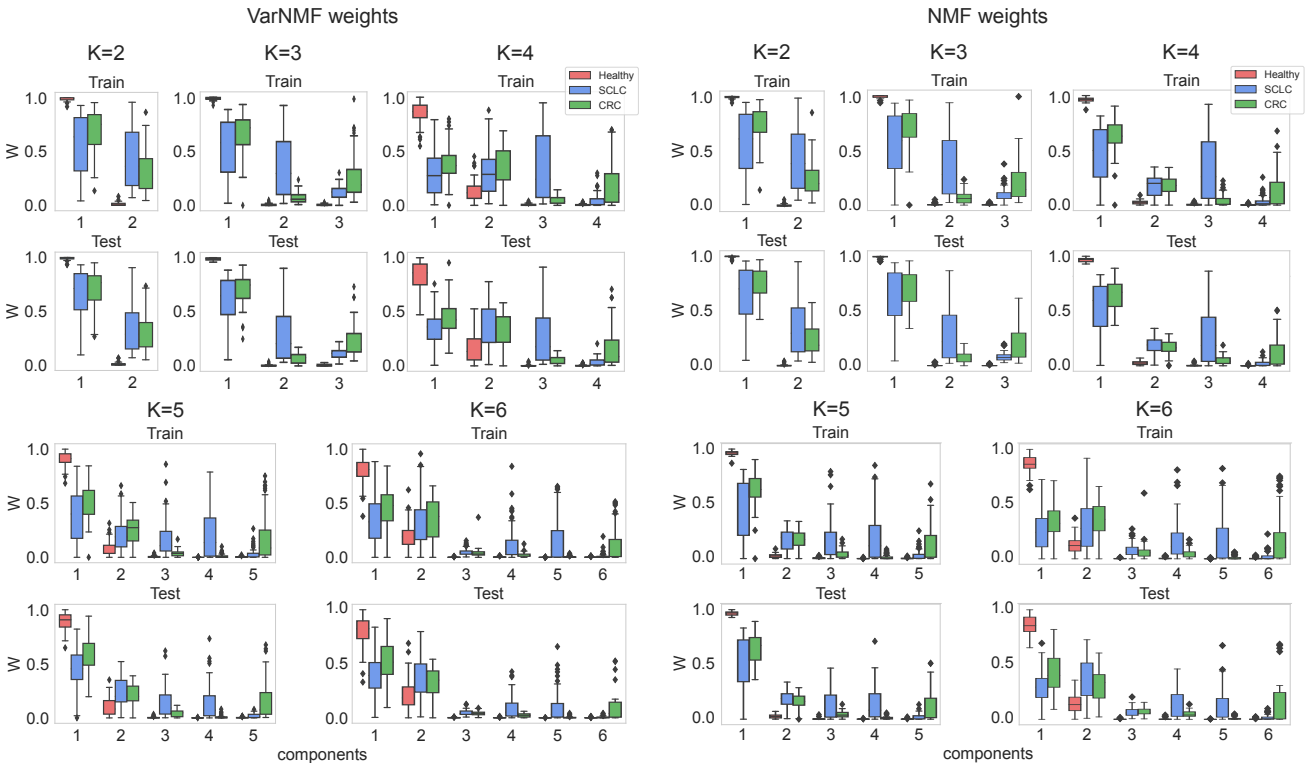


Fig. 14. Decomposing cell-free ChIP-seq data with VarNMF (left) and NMF (right): The estimated train and test weights for each component in the $K = 2, \ldots, 6$ solutions, aggregated by sample sub-cohort (healthy, SCLC and CRC). Weights are normalized so that each sample has a total weight of 1.

Results of NMF and VarNMF for a range of Ks

We examine decomposition of the NMF and VarNMF models and algorithm on cfChiP-seq dataset with increasing number of components $K = 2, \ldots, 6$ (Section 4.2 for more details). In Fig. 14 we look at the resulting weights. We observe similar results in the train and test datasets as well as for both NMF and VarNMF. However, in the NMF solution we observe a single healthy component (that is high in healthy samples but also appear in a range of weights in the cancer samples) whereas VarNMF displays two. Additionally, starting from $K = 3$, there are at least two components that are cancer-specific (that have non-zero weights mostly in one cancer type). The one SCLC-associated component of $K = 4$ splits into two SCLC-associated components starting from $K = 5$, and the CRC-associated component remains unique in all solutions.

Lastly, we observe a shared-cancer component in the $K = 4, 5$ solutions. In VarNMF it is also shared with healthy samples and is associated with genes of blood-related cell-types, which is expected (Appendix 11.2). In NMF, however, this component is not shared with healthy samples, but is still associated with these cell-types. Starting from $K = 6$, the two models results in two healthy components, and one shared cancer component. This suggests that both models extract a component that represent shared features between the cancers.

We conclude that both VarNMF and NMF result in similar weights, with one or two healthy components that reflect the healthy cell-free material in the plasma, one or two cancer-specific components for each type of cancer, that represent the disease contribution to the plasma, and possibly one component that is shared between the cancers.