Bias Aware Probabilistic Boolean Matrix Facortization

Abstract

Boolean matrix factorization (BMF) is a combinatorial problem arising from a wide range of applications including recommendation system, collaborative filtering, and dimensionality reduction. Currently, the noise model of existing BMF methods is often assumed to be homoscedastic; however, in real world data scenarios, the deviations of observed data from their true values are almost surely diverse due to stochastic noises, making each data point not equally suitable for fitting a model. In this case, it is not ideal to treat all data points equally distributed. Motivated by such observations, we introduce a probabilistic BMF model that recognizes the object- and feature-wise bias distribution respectively, called bias aware BMF (BABF). To the best of our knowledge, BABF is the first approach for Boolean decomposition with consideration of the feature-wise and object-wise bias in binary data. We conducted experiments on datasets with different levels of background noise, bias level, and sizes of the signal patterns, to test the effectiveness of our method in various scenarios. We demonstrated that our model outperforms the state-of-the-art factorization methods in both accuracy and efficiency in recovering the original datasets, and the inferred bias level is highly significantly correlated with true existing bias in both simulated and real world datasets.

1 INTRODUCTION

Boolean matrix is a one type of data representation with binary entries that originates from a wide range of applications including recommendation system, network analysis, collaborative filtering, and biological gene expression [Miettinen and Neumann, 2020, Balasubramaniam et al., 2018, Kocayusufoglu et al., 2018, Zhao et al., 2020, Liang et al., 2020]. The goal of **Boolean matrix factorization(BMF)** is to discover hidden patterns from binary data, where it finds a pair of low-rank binary matrices ($X \in \{0,1\}^{m\times k}$, $Y \in \{0,1\}^{k\times n}$) (Figure 1A,B,C), whose Boolean product approximates the original input matrix ($A \in \{0,1\}^{m\times n}$), i.e.,

$$A \sim X \otimes Y, \quad A_{ij} \sim \vee_{l=1}^{k} X_{il} \wedge Y_{lj}.$$

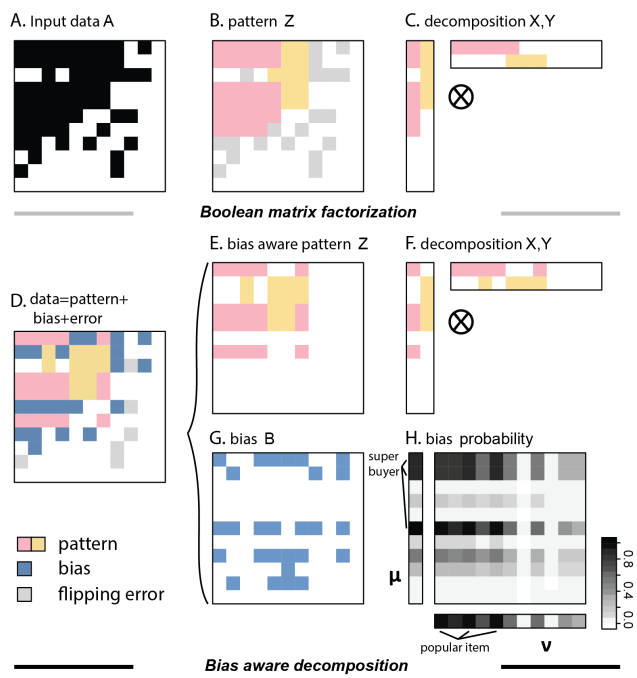

Figure 1: BMF with homoscedastic noise model (A-C) and bias aware BMF with column- and row-specific bias (D-H). H illustrates a biased data case in purchase history data.

Such low-rank decomposition can capture the local **dependency** between subsets of objects (row of A) and subsets of features (column of A). Specifically, in each rank-1 submatrix resulted from the decomposition into X, Y, i.e., $X_{:l} \otimes Y_{l:}$, it indicates a group of objects (i.e., nonzero entries

in $X_{:l}$) sharing the same behavior on a set of features (i.e., nonzero entries in $Y_{l:}$). Here we denote the overall pattern matrix as $Z := X \otimes Y$. For the background error distribution, existing BMF methods tend to assume homoscedastic error distribution, or a universal flipping error with a flipping rate of $p_f = p(A_{ij} = 0|Z_{ij} = 1) = p(A_{ij} = 1|Z_{ij} = 0)$. In other words, the objective of BMF is to find the a decomposition of A such that

$$A = (Z + E)\, mod\, 2\,;\ s.t. Z = X \otimes Y, p(E_{ij} = 1) = p_f$$

where Z, E minimize a certain cost function $\tau(Z, A) = |E| = |A \ominus (X \otimes Y)|$ (Figure 1A,B,C). Here, $mod2$ represents the modulo operation with a quotient of 2, and $|\cdot|$ represents a certain norm measure defined by the cost function $\tau(\cdot)$.

Unfortunately, the assumption on homoscedastic error distribution is often violated when applied to complex real-world data, where the individual objects or features may have its specific bias pattern that result in **heteroscedastic error distribution**. Existing BMF methods fail to account for such object- or feature-specific bias, which could severly impact our ability to identify the true pattern Z, as the error matrix E may display row- or column-specific bias [Wan et al., 2020a]. Take the online transaction records data as an example. The observed transaction records data from customers (row) and items (column) are constituted by three components: pattern, bias and flipping error (Figure 1D), meaning that aside from stochastic error, to determine whether or not a costumer would purchases a certain item, one should not only look at the purchase pattern that he/she belongs to (Figure 1E), but also his/her innate personal purchase preferences and the popularity of the item (Figure 1G). For example, a super-buyer, or someone with impulsive buying habits, is very likely to make a purchase regardless of the properties of the items; while a super-item, or a popular item, is also likely to be purchased by users with different characteristics (Figure 1B,H).

To mend the gap in binary data analysis, we propose **BABF** (**B**ias **A**ware **B**oolean matrix **F**actorization), the first tool to derive the latent binary pattern (Z), in the presence of individual row-wise and column-wise bias (Figure 1D-H), denoted as two real-valued probability vectors $\boldsymbol{\mu}$, $\boldsymbol{\nu}$, with $\boldsymbol{\mu}_i \in [0, 1]\, \forall i \in \{1, ..., m\}$ and $\boldsymbol{\nu}_j \in [0, 1]\, \forall j \in \{1, ..., n\}$. These two vectors represent processes that are object- and feature-specific, and are independent from the pattern generation process, or the homoscedastic background error. In other words, they capture the individual bias generation process that can't be captured by the existing model.

In this work, our contribution is three-fold: 1) BABF is the first method that considers a heteroscedastic error model resulted from object- and feature-specific bias, which is more suitable for modeling real world data; 2) BABF is a highly efficient algorithm in capturing the low rank structures in binary matrix in the presence of individual bias, and showed robust performance in deriving the true patterns across different data scenarios; 3) As a byproduct of pattern discovery, BABF-derived individual bias patterns are highly consistent with the true bias pattern in simulated data and reasonable in real world data, which may lead to practical interpretations depending on different application scenarios.

2 PROBLEM FORMULATION

In this section, we formally address our objective to derive the latent Boolean patterns while considering the individual row- and column-wise bias in a probabilistic framework. We first introduce the notations used across this paper, then report the existing probabilistic BMF framework in Ravanbakhsh et al. [2016], Rukat et al. [2017], and then our bias-aware BMF model, BABF.

2.1 NOTATION

Matrix, vector and scalar values are denoted by uppercase (A), bold lowercase (**a**) and lowercase character (a), respectively. The upper-script represents the dimension of the object (e.g. $A^{m\times n}$), while the lower-script indicates the element indices (e.g. i-th row: $A_{i:}$, j-th column: $A_{:j}$, and ij-th element: A_{ij}). $|\cdot|$ represents a certain type of norm measure. Under Boolean arithmetic, the *and*, *or*, and *not* operations are denoted by $\wedge$, $\vee$, and $\neg$. Subsequently, the Boolean element-wise sum and subtraction are defined as $X \oplus Y = X \vee Y$ and $X \ominus Y = (\neg X \vee Y) \wedge (X \vee \neg Y)$. The Boolean matrix product is defined as $Z = X \otimes Y$, where $Z_{ij} = \vee_{l=1}^{k} X_{ik} \wedge Y_{lj}$.

2.2 EXISTING HOMOSCEDASTIC BMF MODEL

Following Ravanbakhsh et al. [2016], Rukat et al. [2017], each observed entry in a matrix A, i.e. $A_{ij} \in \{0, 1\}$, is assumed to be generated from the latent pattern Z_{ij} with a homoscedastic error model with universal flipping probability p_f, where the likelihood function is defined as

$$p(A_{ij}|Z_{ij}) = \begin{cases} 1 - p_f,\ if\ A_{ij} = Z_{ij} \\ p_f,\ if\ A_{ij} \neq Z_{ij} \end{cases}$$

$$p(A|Z) = \prod_{ij}^{m\times n} p(A_{ij}|Z_{ij})$$

As $Z = X \otimes Y$, individual Bernoulli prior is applied on every element of X and Y, i.e.,

$$p(X) = \prod_{il}^{m\times k} p(X_{ij}) \quad p(Y) = \prod_{lj}^{k\times n} Y_{jl}$$

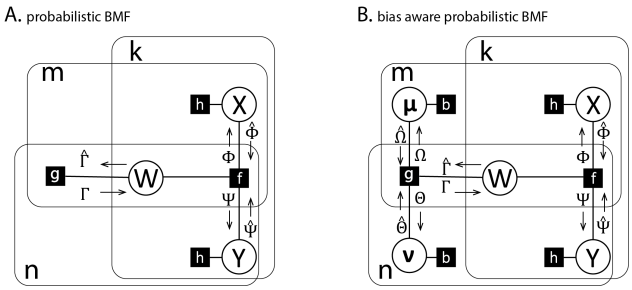

Figure 2: The factor graph representation of BMF and Bias-aware BMF. Noted, figure A is adopted from Ravanbakhsh et al. [2016]

Under this formulation, BMF is equivalent to a *Maximum A posterior (**MAP**)* inference problem of X and Y that maximize the overall likelihood function

$$p(X, Y|A) \propto p(X)p(Y)p(Z|X, Y)p(A|Z)$$

Following Ravanbakhsh et al. [2016], we assume identical Bernoulli prior on X, Y, represented by factor h, e.g., $h(X_{il}) = log(p(X_{il}))$, $h(Y_{lj}) = log(p(Y_{lj}))$. Here, $p(Z|X, Y)$ encodes the hard constraint that ensures the equality of the Boolean product, i.e., $Z = X \otimes Y$. By introducing an auxiliary tensor $W \in \{0, 1\}^{m\times n\times k}$, where $W_{ijl} = X_{il} \wedge Y_{lj},\ Z_{ij} = \vee_{l=1}^{k} W_{ijl}$, this hard constraint is dispersed onto each element in W, and can be reformulated as an identity constraint as

$$p(W_{ijl}|X_{il}, Y_{lj}) = \mathcal{I}(W_{ijl} = X_{il} \wedge Y_{lj})$$

where for $\mathcal{I}$, we have $\mathcal{I}(true) = 1$ and $\mathcal{I}(false) = 0$. Obviously, if $W_{ijl} \neq X_{il} \wedge Y_{lj}$, the factor $f(W_{ijl}, X_{il}, Y_{lj}) = log(p(W_{ijl}|X_{il}, Y_{lj}))$ will be evaluated to be $-\infty$. Finally, factor $g(\{W_{ijl}\}, \forall l \in \{1, ..., k\}) = log(p(A_{ij}|Z_{ij}))$ assess the likelihood of observed variable A_{ij} given the latent pattern Z_{ij}. Overall, we have the factor graph representation of the log-likelihood $p(X, Y|A)$ (Figure 2A, adopted from Ravanbakhsh et al. [2016]) as

$$log(p(X, Y|A)) = \sum_{ij}^{m\times n} h(X_{ij}) + \sum_{lj}^{k\times n} h(Y_{lj}) + \sum_{ijl}^{m\times n\times k} f(W_{ijl}, X_{il}, Y_{lj}) + \sum_{ij}^{m\times n} g(\{W_{ijl}\}_l)$$

Owing to the NP-hard complexity of BMF [Stockmeyer, 1975], it is intractable to infer the MAP of the log-likelihood. Alternatively, focusing on the marginal-MAP often yield good empirical success [Ravanbakhsh et al., 2016, Rukat et al., 2017], e.g.,

$$\arg\max_{X_{il}} log(p(X_{il}|A)) = \arg\max_{X_{il}} \sum_{X_{i'l'}\setminus X_{il}, Y_{l'j'}} log(p(X_{i'l'}, Y_{l'j'}|A))$$

Max-sum belief propagation and Gibbs sampling have been reported to achieve good performance under such strategy [Ravanbakhsh et al., 2016, Rukat et al., 2017].

2.3 PROPOSED BIAS AWARE BMF MODEL

The probabilistic BMF model presented above provides a good framework for us to account for the feature- and object-wise bias. Comparing with the homoscedastic setting, the core advancement of our work is to consider the observed data as generated from a process that is more realistic: aside from stochastic error, or the homoscedastic error distribution as in [Ravanbakhsh et al., 2016, Rukat et al., 2017], we consider that the observed data is generated not only from the latent pattern $Z = X \otimes Y$, but also from independent object/feature behavior process governed by a bias matrix $B \in \{0, 1\}^{m\times n}$, where B is determined by a row- and column-wise bias vector $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ in such way that

$$p_{B_{ij}} = p(B_{ij} = 1) = \boldsymbol{\mu}_i \boldsymbol{\nu}_j$$

And the generation process of A is hence

$$A = B \oplus ((Z + E)\, mod\, 2).$$

The new likelihood of each observations can be characterized in the following four scenarios:

$$\begin{aligned}
p(A_{ij} = 1|Z_{ij} = 0) &= 1 - (1 - p_f)(1 - \boldsymbol{\mu}_i \boldsymbol{\nu}_j) \\
p(A_{ij} = 0|Z_{ij} = 0) &= (1 - p_f)(1 - \boldsymbol{\mu}_i \boldsymbol{\nu}_j) \\
p(A_{ij} = 1|Z_{ij} = 1) &= 1 - p_f(1 - \boldsymbol{\mu}_i \boldsymbol{\nu}_j) \\
p(A_{ij} = 0|Z_{ij} = 1) &= p_f(1 - \boldsymbol{\mu}_i \boldsymbol{\nu}_j)
\end{aligned}$$

The new posterior probability could then be written as

$$p(X, Y, \boldsymbol{\mu}, \boldsymbol{\nu}|A) = p(X)p(Y)p(Z|X, Y)p(\boldsymbol{\mu})p(\boldsymbol{\nu})p(A|Z, \boldsymbol{\mu}, \boldsymbol{\nu})$$

Factor graph representation of the new posterior is shown in Figure 2B. Comparing to the existing probabilistic BMF model introduced in 2.2, the new factor graph involves the row- and column-wise bias vectors $\boldsymbol{\mu}, \boldsymbol{\nu}$. Given no prior knowledge of the two variables, we assume a uniform prior on $\boldsymbol{\mu}, \boldsymbol{\nu}$, thus factor $b(\boldsymbol{\mu}_i), b(\boldsymbol{\nu}_j)$ evaluate to 0 in the graph. And the likelihood factor g is also related to $\boldsymbol{\mu}, \boldsymbol{\nu}$ in the new formulation.

Same as in 2.2, the new bias aware formulation is still an NP-hard problem, and we also turn to find the marginal-MAP, which corresponds to optimally estimating individual variables, while the other variables are marginalized. In the next section, we introduce BABF algorithm to derive the decomposition.

3 THE ALGORITHM OF BABF

While we assume our observations A is generated from two sources, latent pattern Z and Bias B; these two sources themselves can be considered as independent from each other. Such independence is also reflected on the factor graph (Figure 2B). Though the likelihood factor g and the auxiliary variables W are involved with both $\{X, Y\}$ and $\{\boldsymbol{\mu}, \boldsymbol{\nu}\}$, the direct message update of $\{X, Y\}$ and $\{\boldsymbol{\mu}, \boldsymbol{\nu}\}$ are independent with each other. Conveniently, $\{X, Y, W\}$ and $\{\boldsymbol{\mu}, \boldsymbol{\nu}, W\}$ can be considered as two individual systems to be treated separately.

Algorithm 1: BABF, Bias Aware BMF

Inputs: A, k, p_X, p_Y, p_f, t_{all}, t_{MF}, t_{BI}

BABF:

while $t \leq t_{all}$ *and not converged messages*

$\boldsymbol{\mu}^{t+1}, \boldsymbol{\nu}^{t+1} \leftarrow$Bias_infer($A, X^t, Y^t, t_{BI}$)

$X^{t+1}, Y^{t+1} \leftarrow$ prob_BMF($A, k, p_X, p_Y, p_f, \boldsymbol{\mu}^{t+1}, \boldsymbol{\nu}^{t+1}, t_{MF}$)

end

Bias_infer:

$Z := X \otimes Y$

while $t \leq t_{BI}$ *and* $error_now < error_all$

$error_all := error_now$

$\boldsymbol{\mu}_i^{t+1} := \frac{\sum_{j=1, Z_{ij}=0}^{n} A_{ij} \boldsymbol{\nu}_j^t}{\sum_{j=1, Z_{ij}=0}^{n} \boldsymbol{\nu}_j^t} \forall i \in \{1, ..., m\}$

$\boldsymbol{\nu}_j^{t+1} := \frac{\sum_{i=1, Z_{ij}=0}^{m} A_{ij} \boldsymbol{\mu}_i^t}{\sum_{i=1, Z_{ij}=0}^{m} \boldsymbol{\mu}_i^t} \forall j \in \{1, ..., n\}$

$error_now := \sum_{ij, Z_{ij}=0} (A_{ij} - \boldsymbol{\mu}_i \boldsymbol{\nu}_j)^2$

end

prob_BMF:

$p(A_{ij}|Z_{ij}) \leftarrow$ calculate based on $\boldsymbol{\mu}, \boldsymbol{\nu}$.

Initialize Ψ_{ijl}^0, $\hat{\Psi}_{ijl}^0$, Φ_{ijl}^0, $\hat{\Phi}_{ijl}^0$, Γ_{ijl}^0, $\hat{\Gamma}_{ijl}^0, \forall i, j, l$

while $t \leq t_{MF}$ *and not converged messages*

$\Phi_{ijl}^{t+1} := \max(\Gamma_{ijl}^t + \hat{\Psi}_{ijl}^t, 0) - \max(\Psi_{ijl}^t, 0)$

$\Psi_{ijl}^{t+1} := \max(\Gamma_{ijl}^t + \hat{\Phi}_{ijl}^t, 0) - \max(\Phi_{ijl}^t, 0)$

$\hat{\Phi}_{ijl}^{t+1} := log(\frac{1-p_f}{p_f}) + \sum_{j' \neq j} \Phi_{ij'l}^t$

$\hat{\Psi}_{ijl}^{t+1} := log(\frac{1-p_f}{p_f}) + \sum_{i' \neq i} \Psi_{i'jl}^t$

$\Gamma_{ijl}^{t+1} := \min(log(\frac{p(A_{ij}|1)}{p(A_{ij}|0)} + \sum_{l' \neq l} \max(\Gamma_{ijl'}^t)), \max(0, -\max_{l' \neq l} \hat{\Gamma}_{ijl'}^t))$

$\hat{\Gamma}_{ijl}^{t+1} := \min(\hat{\Phi}_{ijl}^t + \hat{\Psi}_{ijl}^t, \hat{\Psi}_{ijl}^k, \hat{\Phi}_{ijl}^t)$

end

$$X_{il} = \begin{cases} 1 & \text{if } log(\frac{1-p_f}{p_f}) + \sum_{i=1}^{m} \Phi_{ijl}^t > 0 \\ 0 & otherwise. \end{cases}$$

$$Y_{lj} = \begin{cases} 1 & \text{if } log(\frac{1-p_f}{p_f}) + \sum_{i=1}^{m} \Psi_{ijl}^t > 0 \\ 0 & otherwise. \end{cases}$$

Outputs: $X, Y, \boldsymbol{\mu}, \boldsymbol{\nu}$

Here we introduce BABF in algorithm 1. BABF has two core components, prob_BMF and Bias_infer, corresponding to the derivations of $\{X, Y\}$ and $\{\boldsymbol{\mu}, \boldsymbol{\nu}\}$. Other than the input data A, BABF takes the pattern number parameters k, the Bernoulli prior of X, Y, filling error p_f and the maximum iterations for the overall algorithm as well as core components $(t_{all}, t_{MF}, t_{BI})$ as input, and outputs the decomposition X, Y and the bias vectors $\boldsymbol{\mu}, \boldsymbol{\nu}$.

3.1 PROB_BMF

When fixing bias vectors $\boldsymbol{\mu}, \boldsymbol{\nu}$, the only differences between bias aware BMF and the existing BMF model introduced in 2.2 is that each likelihood factor g_{ij} would evaluate to different probability assignments by referencing $\boldsymbol{\mu}_i, \boldsymbol{\nu}_j$. Following Ravanbakhsh et al. [2016], we utilize the max-sum belief propagation (BP) strategy to approximate the overall likelihood in prob_BMF. Correspondingly, the message Γ_{ijl} that propagates the likelihood information to auxiliary variable W would be different from Ravanbakhsh et al. [2016] with individualized probabilities. We introduced detailed derivations of the message passing process in the Bias aware factor graph (Figure 2B) in the Appendix.

3.2 BIAS_INFER

The inference of the marginal-MAP of $\boldsymbol{\mu}, \boldsymbol{\nu}$ is a non-trivial task even with accurate pattern information Z, as for any bias variable $\boldsymbol{\mu}_i$, any observation related to this variable is related to a different $\boldsymbol{\nu}_i$, and vice versa. To circumvent this computational challenge, we adopted two modifications. 1) we only consider the observations that are not covered by pattern Z for bias inference. We argue the pattern related observations have marginal contribution to the bias inference and could be omitted. 2) Instead of deriving exact MAP of likelihood, we treat this as an optimization problem, where we could utilize conventional loss functions to achieve the same objective that optimize the difference between $\boldsymbol{\mu}, \boldsymbol{\nu}$ and background information. Inspired from Wan et al. [2020a], we apply a modified mean square loss. Take $\boldsymbol{\mu}_i$ as an example, the loss function takes the form of

$$\Omega_i = \sum_{j=1, Z_{ij}=0}^{n} \boldsymbol{\nu}_j^t (A_{ij} - \boldsymbol{\mu}_i^t)^2$$

The most important benefit of this modified loss is that it ensures the probability of each $\boldsymbol{\mu}_i$ would be from [0,1] and still consider the impact from $\boldsymbol{\nu}_j$ for each observation A_{ij}. Moreover, it is with high computational feasibility as the updated $\boldsymbol{\mu}_i^{t+1}$ could be easily derived as $\boldsymbol{\mu}_i^{t+1} := \frac{\sum_{j=1, Z_{ij}=0}^{n} A_{ij} \boldsymbol{\nu}_j^t}{\sum_{j=1, Z_{ij}=0}^{n} \boldsymbol{\nu}_j^t}$ Correspondingly, $\boldsymbol{\nu}_j^{t+1} := \frac{\sum_{i=1, Z_{ij}=0}^{m} A_{ij} \boldsymbol{\mu}_i^t}{\sum_{i=1, Z_{ij}=0}^{n} \boldsymbol{\mu}_i^t}$. Here, we implement this strategy in Bias_infer. Empirically, it is robust

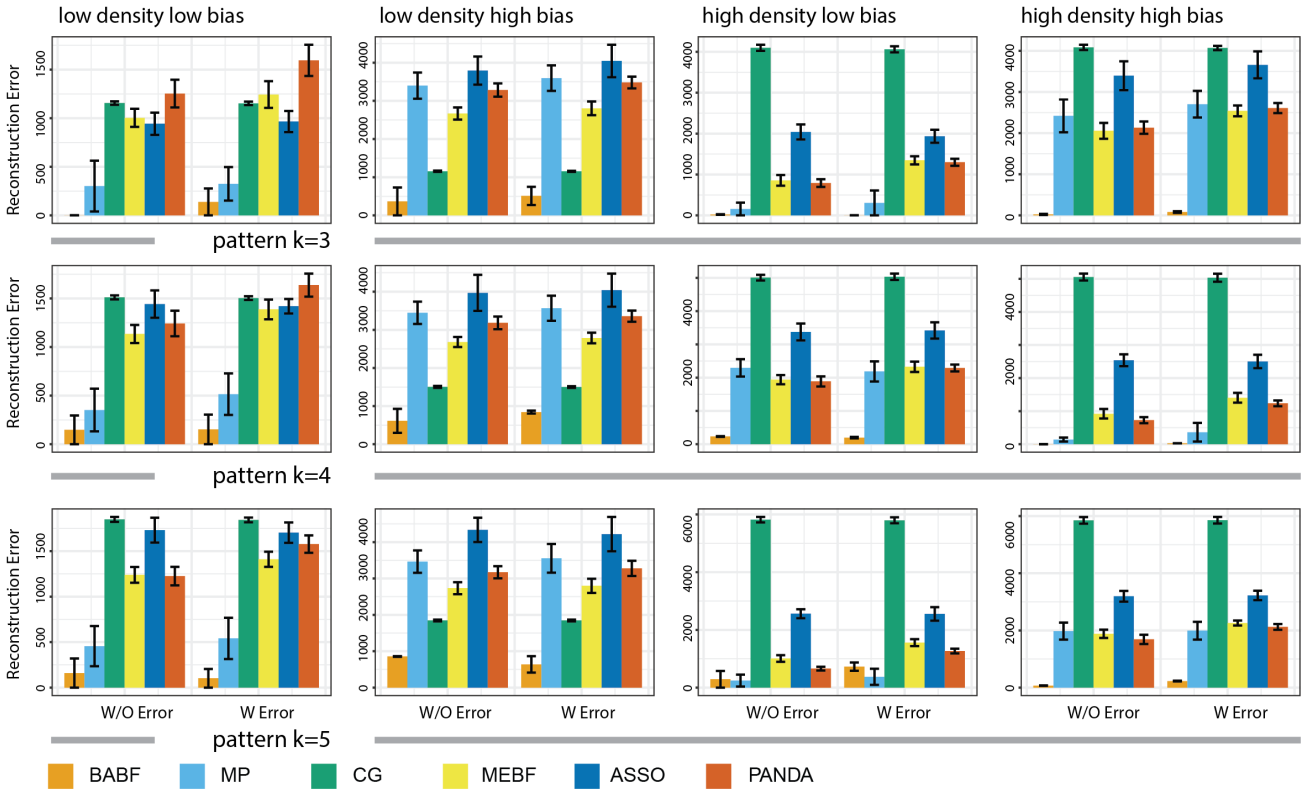

Figure 3: Performance comparison on simulated data

for the bias inference across different scenarios, which we will introduce in detail in the experiments section.

3.3 COMPLEXITY ANALYSIS

The computation cost of BABF depends on the core modules. For each iteration, prob_BMF will visit every variables in $\{X, Y, W\}$, and the calculation of the message update is at constant cost. Such that the cost of prob_BMF is bounded by the size of latent variables, i.e. $O(mnk)$. The consideration of mean square loss enables high computational feasibility to update the bias, that in each iteration of Bias_infer the computation is linear with data size, i.e., $O(mn)$. Thus, the computational cost of each interaction of BABF is $O(mnk)$.

4 EXPERIMENTS

We evaluate the performance of our bias aware model on both synthetic and real world datasets. We first introduce related methods for BMF and report the benchmark performance across different simulated data scenarios. We then highlight the practical use of BABF in our analysis of a movielens and gene expression data.

In this section we illustrate the performance of BABF, the first tool to simultaneously derive both $\{X, Y\}$ and $\{\boldsymbol{\mu}, \boldsymbol{\nu}\}$.

4.1 RELATED WORK

In addition to the probabilistic methods introduced above [Ravanbakhsh et al., 2016, Rukat et al., 2017], different heuristic methods have been developed to solve the BMF problem. Previously Wan et al. [2020a] systematically discussed the bias issue in BMF, but their focus is to explore the identifiability of the patterns in the presence of bias in the noise model. For the rest of the methods, none of them considered the heteroscesdastic issue of the error distribution. Among these methods, ASSO represents a series of work from Miettinen et al [Miettinen et al., 2008, Miettinen and Vreeken, 2011, Karaev et al., 2015, Tatti and Miettinen, 2019]. ASSO first generates a pool of column basis from row-wise correlation matrix, and iteratively search for the best column and row basis following a pre-defined cost function. PANDA is another series of heuristic methods that embed the cost function in the search of top_k core patterns [Lucchese et al., 2010, 2013]. Formal Concept Analysis also showed empirical success in BMF [Belohlavek and Trnecka, 2015, 2018, Belohlavek et al., 2019]. More recently, Wan et al. [2020b] proposed a fast algorithm by formulating submatrix pattern identification in a geometric perspective. Kovacs et al. [2020] formulates BMF as an integer program problem and utilize column generation framework to search for the best solutions. Here, we benchmark the performance of BABF with MP [Ravanbakhsh et al., 2016], CG [Kovacs

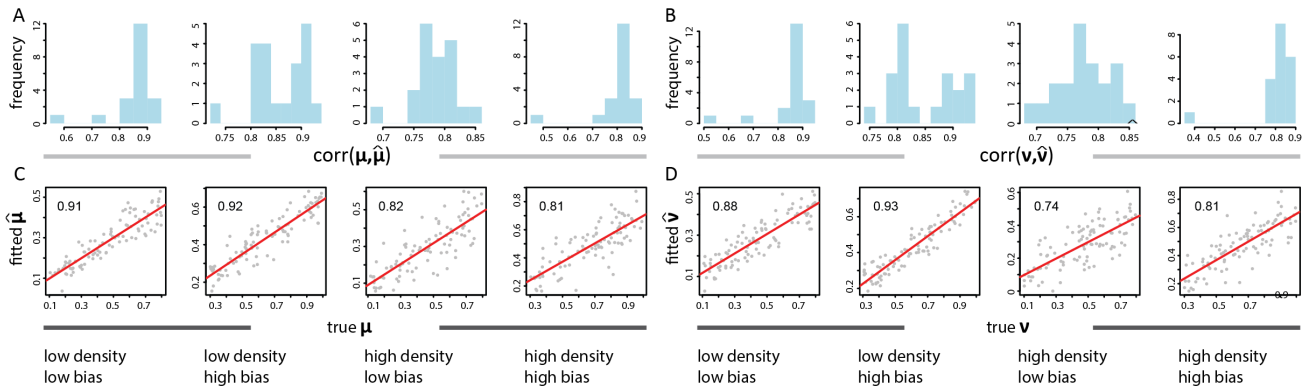

Figure 4: BABF inferred bias is highly correlated with ground truth bias

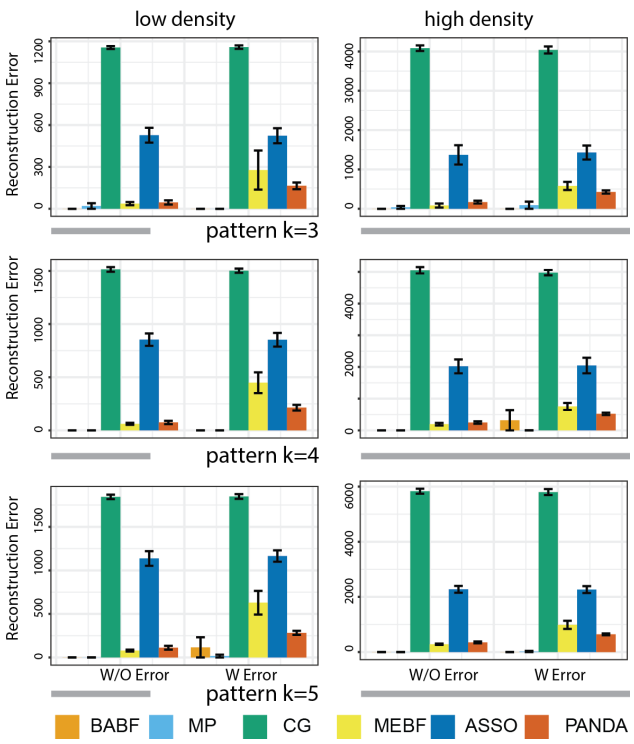

Figure 5: Performance comparison on simulated data without individual bias

et al., 2020], MEBF [Wan et al., 2020b], ASSO [Miettinen et al., 2008] and PANDA [Lucchese et al., 2010] and believe that this set of methods represent the state-of-the-art performance of BMF in different perspectives.

4.2 BENCHMARK ON SIMULATED DATA

We simulate an observed binary matrix A by the following model:

$$A = B \oplus ((Z + E)\, mod\, 2).$$

Here, B, Z, E represent the column-/row-wise bias matrix, pattern matrix and error matrix respectively. Each entry in $B, E \in \{0,1\}^{m\times n}$ is simulated to follow Bernoulli distribution with success probabilities $p(B_{ij}) \propto \boldsymbol{\mu}_i \boldsymbol{\nu}_j$ and $p(E_{ij}) = p_f$. The latent pattern matrix is generated by $Z = X \otimes Y$, where $X \in \{0,1\}^{m\times k}, Y \in \{0,1\}^{k\times n}$, and entries in X, Y also follow Bernoulli distributions with success probabilities $p(X_{il}) = p_X$ and $p(Y_{lj}) = p_Y$. To comprehensively evaluate the methods, we generate varied data scenarios by considering different pattern numbers ($k \in \{3,4,5\}$), and flipping error ($p_f \in \{0, 0.05\}$). We also use different levels of p_X, p_Y to simulate pattern matrices of different density levels, where low density has $p_X = p_Y = 0.2$ while high density has $p_X = p_Y = 0.4$. The bias level is controlled by $\boldsymbol{\mu}, \boldsymbol{\nu}$. In case of low bias, we sample every $\boldsymbol{\mu}_i$, $\boldsymbol{\nu}_j$ uniformly from $[0.1, 0.8]$, which yields a overall bias level of $p_{\bar{B}_{ij}} \sim 0.2$. For the high bias case, $\boldsymbol{\mu}_i$, $\boldsymbol{\nu}_j$ is sampled from $[0.3, 0.9]$ that result a bias probabilities of each variable at $p_{\bar{B}_{ij}} \sim 0.36$. Altogether, we resulted in 24 data scenarios. For each scenarios, we set $m = n = 100$ and simulate 20 replicates.

4.2.1 Performance on reconstruction error

We report the benchmark results in Figure 3. We utilize default setting of the benchmarking methods in our analysis. As for BABF, we assume the prior of X, Y as Bernoulli distribution with $p_X = p_Y = 0.5$ and a flipping error of $p_f = 0.01$. The maximum iterations of t_{all}, t_{BI}, t_{MF} are set at 20, 5 and 50.

For each method, we compare their performance using reconstruction_error, i.e, $|\hat{Z} - Z|$ as evaluation metric. Here, $|\cdot|$ represents the L_1 norm, and $\hat{Z}$ denotes the derived pattern matrix by each method. Lower reconstruction error indicates a better performance. It is anticipated that heuristic approaches like CG, MEBF, ASSO, PANDA would show varied performances respect to different data scenarios as different bias level would result in different impact on their underlying heuristic assumptions. Probabilistic method MP showed an overall stable performance but still struggles with

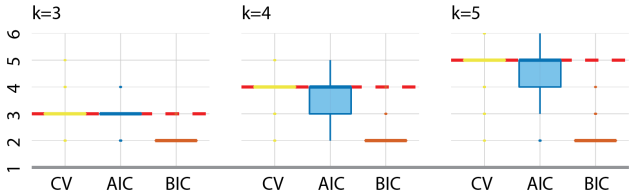

Figure 6: Model selection of pattern number k

high bias level. As expected, BABF achieves the most desirable performance with different bias levels, which highlight the importance to consider individual bias. Additionally, BABF revealed its robustness towards different data scenarios.

4.2.2 Evaluate inferred bias

We explore whether BABF could reliably recover the bias levels $\boldsymbol{\mu}, \boldsymbol{\nu}$. Here, we denote BABF inferred row- and column-wise bias as $\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\nu}}$. Since it is easy to find a scalar value r, s.t., $\boldsymbol{\mu}_i \cdot \boldsymbol{\nu}_j = r\hat{\boldsymbol{\mu}}_i \cdot \frac{1}{r}\hat{\boldsymbol{\nu}}_j$, we do not seek to directly compare the difference of values between $\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i$, or $\boldsymbol{\nu}_j, \hat{\boldsymbol{\nu}}_j$, but instead analyze the correlation between the inferred bias and true bias for every input matrix A, i.e., $corr(\boldsymbol{\mu}, \hat{\boldsymbol{\mu}}), corr(\boldsymbol{\nu}, \hat{\boldsymbol{\nu}})$. We report the correlation results across different data scenarios with pattern number ($k = 4$) in Figure 4. Every scenario has 20 replications. Figure 4A,B show the row- and column-wise bias across different data scenarios. In most cases, BABF inferred bias achieved over 0.8 correlation with ground truth. Even in the worst case, the correlation is as high as 0.4. To give a more intuitive idea, we reveal the inferred bias and true bias of the first input matrix from each scenario as an example in Figure 4C,D. The high correlation suggests desirable performance of BABF to infer the individual bias associated with the objects and the features.

4.2.3 Performance on data without bias

Next, we wish to test how BABF performs on data without bias ($\boldsymbol{\mu}_i = \boldsymbol{\nu}_j = 0, \forall i, j$). In other words, we want to see if BABF is only suitable to model biased data scenarios. In Figure 3, we report the reconstruction error of the methods across 12 data scenarios all without background bias. In general, BABF and MP showed reliable performance. In some high density cases, BABF performs slightly worse than MP, but the difference is only marginal. Overall, BABF showed robust performance towards different data scenarios.

4.2.4 Selection of the pattern number

In our setting, the pattern number k is the most important hyper-parameter that directly determines the number of variables in the factor graph. Under the probabilistic framework, we could utilize different statistic metric to select the most optimal pattern numbers. Here we test three metrics, including cross validation (CV), Akaike information criterion (AIC) and Bayesian information criterion (BIC) for the model selection of k. For CV, we use 90% of the data for fitting and the rest 10% for testing [Kohavi et al., 1995]. For AIC and BIC, we utilize the formulation in Stoica and Selen [2004]. For all the methods, we evaluate the metrics on $k = \{2, ..., 6\}$ and select the best k following their formulation. We tested above metrics across all 24 data scenarios and report the pattern number selections results in box plots (Figure 6). Here red dash marked the ground truth of k. Overall, CV showed consistently accurate selection of pattern number, with only marginal derivations for a small number of cases. AIC and BIC are impacted by the size of input data given a rather large number of variables in the model. Particular, BIC tends to always identify the smallest k for the model.

4.3 ANALYSIS ON REAL-WORLD DATA

We tested the performance of BABF on three real world datasets, movie lens data from Harper and Konstan [2015] and two biological gene expression datasets, head and neck cancer and melanoma single cell RNAseq (scRNA-seq) data from Puram et al. [2017], Tirosh et al. [2016]. The choice of the dataset as well as the pre-processing procedure follows previous works [Rukat et al., 2017, Wan et al., 2020c]. In movie lens data we have 943 users, that rated/not rated 1682 films. In head and neck, and melanoma data, we have 5902 cells that express/not express 7954 genes, and 4486 cells that express/not express 8210 genes. For each dataset, we first identify the number of patterns $k \in \{2, 20\}$ through cross validation, which yield 5 patterns in movie lens, 3 patterns in melanoma and 6 patterns in head and neck. BABF is then applied to retrieve $\hat{X}, \hat{Y}$, $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\nu}}$ following the specific pattern number for each dataset. We mainly focus on addressing two questions: 1. Would the consideration of individual bias benefit our interpretation of real world data? 2. Does the inferred bias carry any practical meaning?

4.3.1 Data interpretation

Since the underlying true patterns of real world data is not accessible, instead of comparing decomposed pattern $\hat{Z}$ with input matrix A, where A is constituted of not only the true pattern matrix Z but also likely noise matrix E and bias matrix B, we use a likelihood metric. Specifically, we evaluate the goodness of model fitting as the overall likelihood, where a larger likelihood indicating a better fitting of the data. We compare the likelihood of BABF with the probabilistic BMF method MP. Similar to Ravanbakhsh et al. [2016], Rukat et al. [2017], we investigate the methods' performance by only keeping a certain percentage of the observations, called observation level, while masking the rest of the observa-

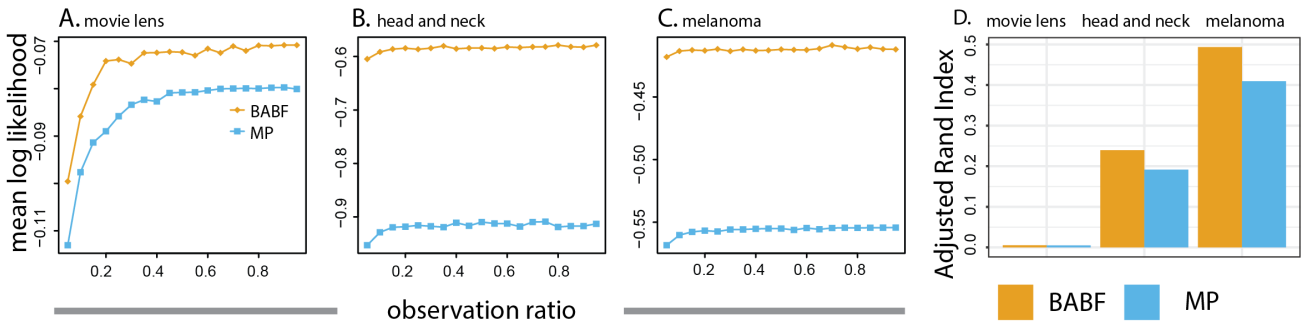

Figure 7: Goodness of fitting on the decomposition for real-world data

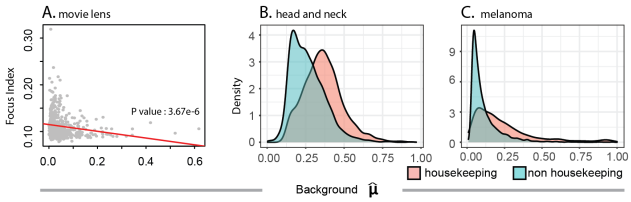

Figure 8: Interpretation analysis on inferred bias

tions. At every observation level, we replicate the analysis for five times and report the mean log-likelihood value. We report the likelihood results in Figure 7A,B,C. On all three datasets, BABF showed higher overall likelihood compared with MP, which suggests that the individual bias assumption is more realistic for real-world data, that movie viewers or cells could be vastly different from each other even in the same pattern group, and such bias is independent with the latent pattern. This advocates the necessity to consider the individual bias in the BMF problem.

To further test the interpretability of the patterns, we examined how the patterns coincide with cell type labels in the two expression datasets, and the movie genres in the movielens dataset, using adjusted rand index, where a higher value corresponds to a greater similarity [Rand, 1971]. Figure 7D shows the peformance of BABF and MP on three datasets. Though both BABF and MP perform poorly on movie lens data, the decomposition from BABF showed higher similarity with given labels in both biological data, which partially revealed a better decomposition of BABF compared with MP.

4.3.2 Practical meaning of inferred bias

The individual bias assumption allows BABF to outperform or have comparable performance with the existing BMF methods, whether such bias is present or not. Here, we want to understand whether the inferred bias could reflect certain practical meaning. Inspired by Wan et al. [2020a], in movie lens data, we want to explore the inferred bias on individual user with their taste on movie types. In our hypothesis, if a user only focus on certain genres of movie, then their behavior could be majorly explained by pattern information Z, while with less effect from B. Here we design the *focus index* to quantify such effect. Specifically if a user watched a numbers of movies in c categories, i.e., $a = b_1 + ... + b_c$, the focus index of this user is calculated as $focus_index := \sum_{i=1}^{c} (\frac{b_i}{a})^2$. As anticipated, inferred bias is negatively correlated with the focus index (Figure 8A, $corr = -0.19,\ p = 3.67e-6$). This significant negative correlation revealed that the inferred bias partially revealed certain taste of the movie viewers.

In the case of gene expression, we focus on two group of genes, housekeeping genes and non-housekeeping genes [Eisenberg and Levanon, 2013]. As the name revealed, housekeeping genes are to maintain the basic activities of the cells, that each cell, regardless of their cell types, will all express these genes. On the other hand, non-housekeeping genes will be the ones that reflect the cell-type specificity. For example, T cells will express T cell marker genes like CD3D,CD3E [Call et al., 2002, Wan et al., 2019]. Figure 8B,C are the density plot visualization of the inferred bias on housekeeping and non-housekeeping genes. As expected, housekeeping genes have a much bigger effect from bias as their expression behavior is not related with any patterns. On the other hand, since the non-housekeeping genes revealed the specificity of the cell, its behavior is largely covered by the latent pattern, such that we witness a small bias in $\hat{\boldsymbol{\mu}}$ on both datasets.

5 CONCLUSION

In this paper, we propose a bias aware model, BABF, which is the first algorithm to derive Boolean matrix decomposition in the presence of individual object- and feature-wise bias. Compared with other methods, BABF is a highly efficient approach, which not only results in good approximation of the true binary pattern with low reconstruction error, but also infers individual bias with high consistency with ground truth. The bias inference from BABF could lead to interesting interpretations depending on different data scenarios.

References

Thirunavukarasu Balasubramaniam, Richi Nayak, and Chau Yuen. People to people recommendation using coupled nonnegative boolean matrix factorization. In *Proceedings of the 2018 International Conference on Soft-computing and Network Security (ICSNS)*, pages 223–227. Institute of Electrical and Electronics Engineers Inc., 2018.

Radim Belohlavek and Martin Trnecka. From-below approximations in boolean matrix factorization: Geometry and new algorithm. *Journal of Computer and System Sciences*, 81(8):1678–1697, 2015.

Radim Belohlavek and Martin Trnecka. A new algorithm for boolean matrix factorization which admits overcovering. *Discrete Applied Mathematics*, 249:36–52, 2018.

Radim Belohlavek, Jan Outrata, and Martin Trnecka. Factorizing boolean matrices using formal concepts and iterative usage of essential entries. *Information Sciences*, 489:37–49, 2019.

Matthew E Call, Jason Pyrdol, Martin Wiedmann, and Kai W Wucherpfennig. The organizing principle in the formation of the t cell receptor-cd3 complex. *Cell*, 111 (7):967–979, 2002.

Eli Eisenberg and Erez Y Levanon. Human housekeeping genes, revisited. *TRENDS in Genetics*, 29(10):569–574, 2013.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

Sanjar Karaev, Pauli Miettinen, and Jilles Vreeken. Getting to know the unknown unknowns: Destructive-noise resistant boolean matrix factorization. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 325–333. SIAM, 2015.

Furkan Kocayusufoglu, Minh X Hoang, and Ambuj K Singh. Summarizing network processes with network-constrained boolean matrix factorization. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 237–246. IEEE, 2018.

Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

Reka A Kovacs, Oktay Gunluk, and Raphael A Hauser. Binary matrix factorisation via column generation. *arXiv preprint arXiv:2011.04457*, 2020.

Lifan Liang, Kunju Zhu, and Songjian Lu. Bem: mining coregulation patterns in transcriptomics via boolean matrix factorization. *Bioinformatics*, 36(13):4030–4037, 2020.

Claudio Lucchese, Salvatore Orlando, and Raffaele Perego. Mining top-k patterns from binary datasets in presence of noise. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 165–176. SIAM, 2010.

Claudio Lucchese, Salvatore Orlando, and Raffaele Perego. A unifying framework for mining approximate top-k binary patterns. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):2900–2913, 2013.

Pauli Miettinen and Stefan Neumann. Recent developments in boolean matrix factorization. *arXiv preprint arXiv:2012.03127*, 2020.

Pauli Miettinen and Jilles Vreeken. Model order selection for boolean matrix factorization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 51–59, 2011.

Pauli Miettinen, Taneli Mielikäinen, Aristides Gionis, Gautam Das, and Heikki Mannila. The discrete basis problem. *IEEE transactions on knowledge and data engineering*, 20(10):1348–1362, 2008.

Sidharth V Puram, Itay Tirosh, Anuraag S Parikh, Anoop P Patel, Keren Yizhak, Shawn Gillespie, Christopher Rodman, Christina L Luo, Edmund A Mroz, Kevin S Emerick, et al. Single-cell transcriptomic analysis of primary and metastatic tumor ecosystems in head and neck cancer. *Cell*, 171(7):1611–1624, 2017.

William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

Siamak Ravanbakhsh, Barnabás Póczos, and Russell Greiner. Boolean matrix factorization and noisy completion via message passing. In *International Conference on Machine Learning*, pages 945–954. PMLR, 2016.

Tammo Rukat, Chris C Holmes, Michalis K Titsias, and Christopher Yau. Bayesian boolean matrix factorisation. In *International conference on machine learning*, pages 2969–2978. PMLR, 2017.

Larry J Stockmeyer. *The set basis problem is NP-complete*. IBM Thomas J. Watson Research Division, 1975.

Petre Stoica and Yngve Selen. Model-order selection: a review of information criterion rules. *IEEE Signal Processing Magazine*, 21(4):36–47, 2004.

Nikolaj Tatti and Pauli Miettinen. Boolean matrix factorization meets consecutive ones property. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 729–737. SIAM, 2019.

Itay Tirosh, Benjamin Izar, Sanjay M Prakadan, Marc H Wadsworth, Daniel Treacy, John J Trombetta, Asaf Rotem, Christopher Rodman, Christine Lian, George Murphy, et al. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell rna-seq. *Science*, 352(6282):189–196, 2016.

Changlin Wan, Wennan Chang, Yu Zhang, Fenil Shah, Xiaoyu Lu, Yong Zang, Anru Zhang, Sha Cao, Melissa L Fishel, Qin Ma, et al. Ltmg: a novel statistical modeling of transcriptional expression states in single-cell rna-seq data. *Nucleic acids research*, 47(18):e111–e111, 2019.

Changlin Wan, Wennan Chang, Tong Zhao, Sha Cao, and Chi Zhang. Denoising individual bias for fairer binary submatrix detection. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2245–2248, 2020a.

Changlin Wan, Wennan Chang, Tong Zhao, Mengya Li, Sha Cao, and Chi Zhang. Fast and efficient boolean matrix factorization by geometric segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6086–6093, 2020b.

Changlin Wan, Dongya Jia, Yue Zhao, Wennan Chang, Sha Cao, Xiao Wang, and Chi Zhang. A data denoising approach to optimize functional clustering of single cell rna-sequencing data. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 217–222. IEEE, 2020c.

Sha Zhao, Gang Pan, Jianrong Tao, Zhiling Luo, Shijian Li, and Zhaohui Wu. Understanding smartphone users from installed app lists using boolean matrix factorization. *IEEE Transactions on Cybernetics*, 2020.

Appendix

A MESSAGE UPDATE

Despite the dense connections in the factor graph, max-sum belief propagation achieved admirable performance in the case of approximate the MAP of Boolean matrix factorization [Ravanbakhsh et al., 2016]. Here we also utilize this strategy that not only derive the MAP of matrix decomposition X and Y, but also infer the background row- and column-wise bias $\boldsymbol{\mu}, \boldsymbol{\nu}$. Though the information of $\boldsymbol{\mu}, \boldsymbol{\nu}$ and X, Y communicates through likelihood factor g and auxiliary variable W, their independence of each other resulted in disconnected message update between $\boldsymbol{\mu}, \boldsymbol{\nu}$ and X, Y. Conveniently, $\{X,\, Y,\, W\}$ and $\boldsymbol{\mu}, \boldsymbol{\nu},\, W\}$ can be considered as two separate systems. In this paper we focus on the message update of $\{\boldsymbol{\mu}, \boldsymbol{\nu},\, W\}$, and adopt the algorithm in Ravanbakhsh et al. [2016] for $\{X,\, Y,\, W\}$.

A.0.1 update X,Y,W

Variables to factor message.

Conveniently, all the variables in $\{X, Y, W\}$ are binary variables ($X_{il},\, Y_{lj},\, W_{ijl} \in \{0, 1\}$). Following the notation in Ravanbakhsh et al. [2016], we denote the message between factors and variables as $\mathbf{m}$ (e.g., $\mathbf{m}_{X_{il} \to f_{ijl}}(X_{ij}) : \{0, 1\} \to \mathcal{R}$). Max-sum BP is utilized to calculate the outgoing message, while consideration all incoming messages from neighbor factors, despite the receiving one, e.g.,

$$\mathbf{m}_{X_{il} \to f_{ijl}}(X_{ij})^{t+1} = \mathbf{m}_{h_{il} \to X_{ill}}(X_{il})^t + \sum_{j' \neq j} \mathbf{m}_{f_{ij'l} \to X_{il}}(X_{il})^t$$

Our objective is to achieve the maximum likelihood, which align with the difference between the message of $\mathbf{m}_{X_{il} \to f_{ijl}}(X_{ij} = 1)$ and $\mathbf{m}_{X_{il} \to f_{ijl}}(X_{ij} = 0)$, i.e.,

$$\hat{\Phi} = \mathbf{m}_{X_{il} \to f_{ijl}}(1) - \mathbf{m}_{X_{il} \to f_{ijl}}(0)$$

In the case of individual variable X_{il} to the factor f_{ijl}

$$\begin{aligned}\hat{\Phi}_{ijl}^{t+1} &= (\mathbf{m}_{h_{il} \to X_{ill}}(1)^t + \sum_{j' \neq j} \mathbf{m}_{f_{ij'l} \to X_{il}}(1)^t) \\ &- (\mathbf{m}_{h_{il} \to X_{ill}}(0)^t + \sum_{j' \neq j} \mathbf{m}_{f_{ij'l} \to X_{il}}(0)^t) \\ &= log(\frac{p(X_{il} = 1)}{p(X_{il} = 0)}) + \sum_{j' \neq j} \Phi_{ij'l}^t\end{aligned}$$

Similarly, the message $\hat{\Psi}$ can be derived as

$$\hat{\Psi}_{ijl} = log(\frac{p(Y_{lj} = 1)}{p(Y_{lj} = 1)}) + \sum_{i' \neq i} \Psi_{i'jl}^t$$

For W, since each variable W_{ijl} has exact two factor neighbors $g_{ij},\ f_{ijl}$, the message from W_{ijk} to either factors is the message from the other factor, i.e.,

$$\mathbf{m}_{W_{ijl}\to g_{ij}}(W_{ijl}) = \mathbf{m}_{f_{ijl}\to W_{ijl}}(W_{ijl})$$

$$\mathbf{m}_{g_{ij}\to W_{ijl}}(W_{ijl}) = \mathbf{m}_{W_{ijl}\to f_{ijl}}(W_{ijl})$$

We will discuss in detail of the message involve factor g in next section.

factor to variable message

For factor h, it only connect to the single variable X_{il} or Y_{lj}, which works as prior knowledge for the sparsity of X and Y, where their information is passed through

$$h_{il}(X_{il}=1) - h_{il}(X_{il}=0) = log(\frac{p(X_{il}=1)}{p(X_{il}=0)})$$

$$h_{lj}(Y_{lj}=1) - h_{lj}(Y_{lj}=0) = log(\frac{p(Y_{lj}=1)}{p(Y_{lj}=0)})$$

Factor f links X, Y with the auxiliary variable W, that ensures $W_{ijl} = X_{il} \wedge Y_{lj}$, i.e.,

$$f(X_{il}, Y_{lj}, W_{ijl}) = log(\mathcal{I}(W_{ijl} = X_{il} \wedge Y_{lj}))$$

Notably, $f(X_{il}, Y_{lj}, W_{ijl}) \to -\infty$ if $W_{ijl} \neq X_{il} \wedge Y_{lj}$. Such that it restrict the message scenarios when passing the information from f to X, Y. Here, we use $\mathbf{m}_{f_{ijl}\to X_{il}(X_{il})}$ as example, where $\mathbf{m}_{f_{ijl}\to Y_{lj}(Y_{lj})}$ can be similarly derived. For X_{il} to equal to 1, if $Y_{lj}=1$, restricted by f, $W_{ijl}=1$, and if $Y_{lj}=0$, $W_{ijl}=0$, thus,

$$\begin{aligned}&\mathbf{m}_{f_{ijl}\to X_{il}}(1)^{t+1} = max(\mathbf{m}_{Y_{lj}\to f_{ill}}(1)^t + \\ &\mathbf{m}_{W_{ijl}\to f_{ijl}}(1)^t, \quad \mathbf{m}_{Y_{lj}\to f_{ill}}(0)^t + \mathbf{m}_{W_{ijl}\to f_{ijl}}(0)^t)\end{aligned}$$

While if $X_{il}=0$, $W_{ijl}=0$ regardless the value of Y_{lj}, i.e.,

$$\begin{aligned}&\mathbf{m}_{f_{ijl}\to X_{il}}(0)^{t+1} = max(\mathbf{m}_{Y_{lj}\to f_{ill}}(1)^t + \\ &\mathbf{m}_{W_{ijl}\to f_{ijl}}(0)^t, \quad \mathbf{m}_{Y_{lj}\to f_{ill}}(1)^t + \mathbf{m}_{W_{ijl}\to f_{ijl}}(0)^t)\end{aligned}$$

Since $\hat{\Psi}_{ijl} = \mathbf{m}_{Y_{lj}\to f_{ijl}}(1) - \mathbf{m}_{Y_{lj}\to f_{ijl}}(0)$, and $\Gamma_{ijl} = \mathbf{m}_{W_{ijl}\to f_{ijl}}(1) - \mathbf{m}_{W_{ijl}\to f_{ijl}}(0)$ the message from f to X can be derived as

$$\begin{aligned}\Phi_{ijl} &= \mathbf{m}_{f_{ijl}\to X_{il}}(1) - \mathbf{m}_{f_{ijl}\to X_{il}}(0) \\ &= max(\Gamma_{ijl} + \hat{\Psi}_{ijl}, 0) - max(\hat{\Psi}_{ijl}, 0)\end{aligned}$$

Similarly, we have

$$\begin{aligned}\Psi_{ijl} &= \mathbf{m}_{f_{ijl}\to Y_{il}}(1) - \mathbf{m}_{f_{ijl}\to Y_{il}}(0) \\ &= max(\Gamma_{ijl} + \hat{\Phi}_{ijl}, 0) - max(\hat{\Phi}_{ijl}, 0)\end{aligned}$$

Following the same strategy, while considering the message from factor f to variable W, if $W_{ijl}=1$, $X_{il}=Y_{lj}=1$, whereas if $W_{ijl}=0$, either X_{il} or Y_{lj} should equal to zero, i.e.,

$$\mathbf{m}_{f_{ijl}\to W_{ijl}}(1)^{t+1} = \mathbf{m}_{Y_{lj}\to f_{ill}}(1)^t + \mathbf{m}_{X_{il}\to f_{ijl}}(1)^t$$

$$\begin{aligned}&\mathbf{m}_{f_{ijl}\to W_{ijl}}(0)^{t+1} = max(\mathbf{m}_{Y_{lj}\to f_{ill}}(1)^t + \\ &\mathbf{m}_{X_{il}\to f_{ijl}}(0)^t, \mathbf{m}_{Y_{lj}\to f_{ill}}(0)^t + \mathbf{m}_{X_{il}\to f_{ijl}}(1)^t, \\ &\mathbf{m}_{Y_{lj}\to f_{ill}}(0)^t + \mathbf{m}_{X_{il}\to f_{ijl}}(0)^t)\end{aligned}$$

Such that

$$\begin{aligned}\hat{\Gamma}_{ijk} &= \mathbf{m}_{f_{ijl}\to W_{ijl}}(1) - \mathbf{m}_{f_{ijl}\to W_{ijl}}(0) \\ &= min(\hat{\Phi}_{ijl} + \hat{\Psi}_{ijl}, \hat{\Phi}_{ijl}, \hat{\Psi}_{ijl})\end{aligned}$$

A.1 UPDATE μ, ν, W

In the previous section, we have derived the messages passing between the X, Y and W. In this section, we derive the message passing between μ, $\mathbf{v}$ and W, where they all related to the likelihood factor g. Also, different with binary variable W, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are Bernoulli variable, that the simplified singleton message does not applied for their message update. We first reinstate the log likelihood function of each element (A_{ij}) that represent the factor g.

$$\begin{aligned}&p(A_{ij}=1|Z_{ij}=0) = 1-(1-p_f)(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j) \\ &p(A_{ij}=0|Z_{ij}=0) = (1-p_f)(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j) \\ &p(A_{ij}=1|Z_{ij}=1) = 1-p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j) \\ &p(A_{ij}=0|Z_{ij}=1) = p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)\end{aligned}$$

In the case of Bernoulli variables $\boldsymbol{\mu}_i$, the incoming message from factor g to $\boldsymbol{\mu}_i$ is certainly the likelihood information,

$$\Omega_i = log(\prod_{j=1}^{n} p(A_{ij}))$$

while the message from $\boldsymbol{\mu}_i$ to g would be the MAP of the posterior distribution, i.e.,

$$\begin{aligned}\hat{\Omega}_i &= \arg\max_{\boldsymbol{\mu}_i} log(\prod_{j=1}^{n} p(A_{ij})p(\boldsymbol{\mu}_i)) \\ &= \arg\max_{\boldsymbol{\mu}_i}(\sum_{j=1}^{n} log(p(A_{ij})) + b_i(\boldsymbol{\mu}_i))\end{aligned}$$

Given no knowledge on the bias before hand, here we impose a uniform prior on the Bernoulli variable, such that $b_i(\boldsymbol{\mu}_i) = 0$. In addition, the log posterior is related to 4

situations,

$$\Omega_i = \sum_{j=1,A_{ij}=1,Z_{ij}=0}^{n} log(1-(1-p_f)(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)) + \sum_{j=1,A_{ij}=1,Z_{ij}=1}^{n} log(1-p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)) + \sum_{j=1,A_{ij}=0,Z_{ij}=0}^{n} log((1-p_f)(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)) + \sum_{j=1,A_{ij}=0,Z_{ij}=1}^{n} log(p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)$$

Here we assume $P_f \to 0$, such that $p_f(1-\boldsymbol{\mu}_i\boldsymbol{v}_i) \to 0$, and both $\sum_{j=1,A_{ij}=1,Z_{ij}=1}^{n} log(1-p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j))$ and $\sum_{j=1,A_{ij}=0,Z_{ij}=1}^{n} log(p_f(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)$ can be approximate by a constant that does not contribute to the inference of $\hat{\Omega}_i$. Also $(1-p_f)(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)$ can be approximated by $(1-\boldsymbol{\mu}_i\boldsymbol{\nu}_j)$. It also has practical meanings, that for the inference of background bias, we only consider the values that are not covered by the latent pattern X, Y. While our objective is to infer $\boldsymbol{\mu}_i$ that better reflect the background information of $A_{i:}$. However, it is still non-trivial to derive $\hat{\Omega}_i$ as every observation is related to a different $\boldsymbol{v}_j$. Instead of deriving exact MAP of likelihood, we treat this as an optimization problem, where we could utilize conventional loss function to achieve the same objective that optimize the difference between $\boldsymbol{\mu}_i$ with $A_{i:}$. Here, we apply a modified mean square loss, i.e.,

$$\Omega = \sum_{j=1,Z_{ij}=0}^{n} \boldsymbol{v}_j(A_{ij}-\boldsymbol{\mu}_i)^2$$

The most important benefit of this modified loss is that it ensures the probability of each $\boldsymbol{\mu}_i$ would be from $[0,1]$ and still consider the impact from $\boldsymbol{v}_j$ for each observations. Conveniently, $\hat{\Omega}_i$ is inferred from the derivative of Ω, i.e.,

$$\hat{\Omega} = \arg\max_{\boldsymbol{\mu}_i} \Omega = \frac{\sum_{j=1,Z_{ij}=0}^{n} A_{ij}v_j}{\sum_{j=1,Z_{ij}=0}^{n} v_j}$$

Similarly, we have

$$\Theta_j = \sum_{i=1,Z_{ij}=0}^{m} \boldsymbol{\mu}_i(A_{ij}-\boldsymbol{v}_j)^2$$

$$\hat{\Theta}_j = \frac{\sum_{i=1,Z_{ij}=0}^{m} A_{ij}\boldsymbol{\mu}_i}{\sum_{i=1,Z_{ij}=0}^{n} \boldsymbol{\mu}_i}$$

Now we have derived all messages in the likelihood despite $\Gamma_{ijl} : \mathbf{m}_{g_{ij}\to W_{ijl}}$ that passed the information from the likelihood factor to each of auxiliary variable W_{ijl}. Overall, the message take the form of

$$\mathbf{m}_{g_{ij}\to W_{ijl}}(W_{ijl})^{t+1} = \max_{W_{ijl'},l'\neq l}(g_{ij}(Z_{ij},\boldsymbol{\mu}_i,\boldsymbol{v}_j) + \sum_{l'\neq l}\mathbf{m}_{W_{ijl'}\to g_{ij}}(W_{ijl'})^t)$$

When updating W_{ijl}, we consider two scenarios: 1. $Z_{ij} = \vee_{l=1}^{k} W_{ijl} = 1$ with likelihood factor $p(A_{ij}|Z_{ij}=1)$ and 2. $\vee_{l=1}^{k} W_{ijl} = 0$, $p(A_{ij}|Z_{ij}=0)$.

$W_{ijl} = 1$ falls into the situation of scenarios 1, that no matter the value of $W_{ijl'}$, $Z_{ij} = \vee W_{ijl} = 1$. The message for $W_{ijl} = 1$ can be derived as

$$\begin{aligned}\mathbf{m}_{g_{ij}\to W_{ijl}}(1) &= \max_{W_{ijl'},l'\neq l}(g_{ij}(Z_{ij},\boldsymbol{\mu}_i,\boldsymbol{v}_j) \\ &+ \sum_{l'\neq l}\mathbf{m}_{W_{ijl'}\to g_{ij}}(W_{ijl'})^t) \\ &= log(p_(A_{ij}|1)) \\ &+ \sum_{l'\neq l} max(\mathbf{m}_{W_{ijl'}\to g_{ij}}(1), \mathbf{m}_{W_{ijl'}\to g_{ij}}(0))\end{aligned}$$

$W_{ijl} = 0$ could involve both cases. If $Z_{ij} = 0$, all $W_{ijl'} = 0$, i.e.,

$$\mathbf{m}_{g_{ij}\to W_{ijl}}(0) = log(p_(A_{ij}|0)) + \sum_{l'\neq l}\mathbf{m}_{W_{ijl'}\to g_{ij}}(0)$$

If $Z_{ij} = 1$, at least one of $W_{ijl'}$ equal to zero. To achieve the maximum likelihood, the $W_{ijl'}$ with the maximum likelihood difference on 0 or 1 should be set as 1, we denote it as W_{ijl^*}, where $l^* = \arg\max_{l'\neq l}(\mathbf{m}_{W_{ijl'}\to g_{ij}}(1) - \mathbf{m}_{W_{ijl'}\to g_{ij}}(0))$, such that we have

$$\begin{aligned}\mathbf{m}_{g_{ij}\to W_{ijl}}(0) &= log(p_(A_{ij}|1)) \\ &+ \sum_{l'\neq l} max(\mathbf{m}_{W_{ijl'}\to g_{ij}}(1), \mathbf{m}_{W_{ijl'}\to g_{ij}}(0)) \\ &- \mathbf{m}_{W_{ijl*}\to g_{ij}}(0)\end{aligned}$$

Taken together,

$$\begin{aligned}\mathbf{m}_{g_{ij}\to W_{ijl}}(0) &= max(log(p_(A_{ij}|0)) \\ &+ \sum_{l'\neq l}\mathbf{m}_{W_{ijl'}\to g_{ij}}(0), log(p_(A_{ij}|1)) \\ &+ \sum_{l'\neq l} max(\mathbf{m}_{W_{ijl'}\to g_{ij}}(1), \mathbf{m}_{W_{ijl'}\to g_{ij}}(0)) \\ &- \mathbf{m}_{W_{ijl*}\to g_{ij}}(0))\end{aligned}$$

Therefore

$$
\begin{aligned}
&\Gamma_{ijl} = \mathbf{m}_{g_{ij} \to W_{ijl}}(1) - \mathbf{m}_{g_{ij} \to W_{ijl}}(0) \\
&= log(p_(A_{ij}|1)) \\
&+ \sum_{l' \neq l} max(\mathbf{m}_{W_{ijl'} \to g_{ij}}(1), \mathbf{m}_{W_{ijl'} \to g_{ij}}(0)) \\
&- max(log(p_(A_{ij}|0)) \\
&+ \sum_{l' \neq l} \mathbf{m}_{W_{ijl'} \to g_{ij}}(0), log(p_(A_{ij}|1)) \\
&+ \sum_{l' \neq l} max(\mathbf{m}_{W_{ijl'} \to g_{ij}}(1), \mathbf{m}_{W_{ijl'} \to g_{ij}}(0)) \\
&- \mathbf{m}_{W_{ijl*} \to g_{ij}}(0)) \\
&= min(log(\frac{p(A_{ij}|1)}{p(A_{ij}|0)}) + \sum_{l' \neq l} max(0, \hat{\Gamma}^t_{ijl'}), \\
&max(0, -max_{l' \neq l} \hat{\Gamma}^t_{ijl'}))
\end{aligned}
$$