Orthogonal nonnegative matrix tri-factorization based on Tweedie distributions

Hiroyasu Abe¹ · Hiroshi Yadohisa²

Received: 3 April 2016 / Revised: 25 August 2018 / Accepted: 15 October 2018 /
Published online: 25 October 2018
© The Author(s) 2018

Abstract

Orthogonal nonnegative matrix tri-factorization (ONMTF) is a biclustering method using a given nonnegative data matrix and has been applied to document-term clustering, collaborative filtering, and so on. In previously proposed ONMTF methods, it is assumed that the error distribution is normal. However, the assumption of normal distribution is not always appropriate for nonnegative data. In this paper, we propose three new ONMTF methods, which respectively employ the following error distributions: normal, Poisson, and compound Poisson. To develop the new methods, we adopt a k -means based algorithm but not a multiplicative updating algorithm, which was the main method used for obtaining estimators in previous methods. A simulation study and an application involving document-term matrices demonstrate that our method can outperform previous methods, in terms of the goodness of clustering and in the estimation of the factor matrix.

Keywords Orthogonal nonnegative matrix tri-factorization · Biclustering · Tweedie family · Compound Poisson distribution · Spherical k -means

Mathematics Subject Classification 15A23 Factorization of matrices · 62H30 Classification and discrimination; cluster analysis · 68T10 Pattern recognition, speech recognition

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s11634-018-0348-8>) contains supplementary material, which is available to authorized users.

✉ Hiroyasu Abe
hiro00012@gmail.com

Hiroshi Yadohisa
hyadohis@mail.doshisha.ac.jp

¹ Kyoto University, 54 Kawahara-cho, Shogoin, Sakyo-ku, Kyoto-fu 606-8507, Japan

² Doshisha University, 1-3 Tatara Miyakodani, Kyotanabe-shi, Kyoto-fu 610-0394, Japan

1 Introduction

Nonnegative matrix factorization (NMF), which is a dimension reduction technique for decomposing a data matrix into two factor matrices, in both of which all entries are nonnegative, has been applied to many fields and extended to various forms (Lee and Seung 1999, 2001; Berry et al. 2007; Wang and Zhang 2013). One of best-known extensions is orthogonal NMF (ONMF), which imposes column orthogonality on one side's nonnegative factor matrix (Ding et al. 2006; Yoo and Choi 2008; Choi 2008; Yoo and Choi 2010a; Li et al. 2010; Pompili et al. 2014; Mirzal 2014; Kimura et al. 2014). Because a nonnegative column orthogonal matrix plays a role analogous to an indicator matrix in k -means clustering, and in fact one can obtain the sparse factor matrix from ONMF, it has mainly been adopted for nearest-neighbor clustering tasks such as document and term clustering (Mauthner et al. 2010; Kim et al. 2011; Wang et al. 2016). Another extended version of NMF is nonnegative matrix tri-factorization (NMTF), which decomposes a nonnegative matrix into three nonnegative factor matrices. Because constraint-free NMTF is known to be generally equivalent to NMF, some constraints are often imposed. One popular constraint is column orthogonality of the left- and right-side nonnegative factor matrices, similar to ONMF. This NMTF is referred to as orthogonal NMTF (ONMTF) (Ding et al. 2006; Yoo and Choi 2010b, 2009). Owing to the relationship between column orthogonal nonnegative factor matrices and clustering mentioned above, ONMTF is considered to be a biclustering method. The objective of biclustering is to simultaneously detect any row or column clusters of a data matrix (Govaert and Nadif 2013). The sample objects and variables are classified using data matrix at a time. It has been adopted for use in document-term clustering, collaborative filtering, etc. (Costa and Ortale 2014; Chen et al. 2009).

In ONMF and ONMTF, it is often assumed that the error is normal or Poisson distributed. However, in NMF, various algorithms have been proposed based on various error distributions, including generalized ones. Tweedie family distributions are well-known generalized distributions for NMF (Févotte and Idier 2011; Nakano et al. 2010). It uses the index parameter β to identify the distribution, and includes normal ($\beta = 2$), Poisson ($\beta = 1$), compound Poisson (CP) ($\beta \in (0, 1)$), gamma ($\beta = 0$), and inverse normal ($\beta = -1$) as special cases (Jørgensen 1997; Dunn and Smyth 2001). The assumption of a Tweedie family distribution as the error distribution implies that the error criterion has the form of a β -divergence (Simsekli et al. 2013; Tan and Févotte 2013). β -divergence is a generalized divergence that includes Euclidean distance ($\beta = 2$), KL-divergence ($\beta = 1$), and Itakura–Saito divergence ($\beta = 0$) (Cichocki and Amari 2010). One of the merits of Tweedie family distributions is their flexibility with real nonnegative data. The assumption of normality for nonnegative data means that the variance is the same whenever the expected value is small or large. However, one can adjust the relationship between expected value and variance in a Tweedie distribution assumption by changing β , because variance is proportional to a power of the expected value, such as in $V(y) = \phi E(y)^{2-\beta}$, where $V(y)$ and $E(y)$ are the expected value and variance, respectively, of the random variable y , and ϕ is the dispersion parameter in the Tweedie family distribution. For example, if we assume a CP distribution (i.e., choose β in $(0, 1)$), it is implicitly suggested that the variance will be large when the

expected value is also large. In fact, a CP distribution is defined as a Poisson mixture of gamma distributions; it is absolutely continuous on the positive axis and has a positive mass at zero, from the aspect of the generative model. In other words, a CP distributed random variable is the sum of n independent identically gamma-distributed random variables, and the number of random variables n is Poisson distributed. This assumption is familiar with a gross summation of the nonnegative values and associated with various types of real-world nonnegative data (e.g., precipitation, insurance, and purchase volume data) (Ohnishi and Dunn 2007; Smyth and Jørgensen 2002). From an NMF parameter estimation aspect, this assumption is related to robust estimation in the presence of outliers that have extremely large positive values (Li et al. 2017; Virtanen et al. 2015; Carabias-Orti et al. 2013; Weninger and Schuller 2012; Févotte et al. 2009; Virtanen 2007).

In this paper, we propose a new ONMTF method that generalizes the error distribution to a Tweedie family distribution. Our new method has two advantages. First, it facilitates interpretations from nonnegative data matrices by simultaneously detecting the clusters of row and column objects and the relationships among them. Second, the assumption of the error distribution is plausible with some nonnegative data, which leads to robust estimation against extremely large positive values.

One of the ways to develop the new method is to derive an iterative algorithm for estimating factor matrices using the same techniques as the previous ONMTF methods proposed by Ding et al. (2006) and Yoo and Choi (2010b). In both methods, the factor matrices are estimated using a multiplicative updating algorithm, in which they are iteratively updated by element-wise multiplication. However, this algorithm suffers from two drawbacks. First, column orthogonality is approximately (not precisely) obtained despite the column orthogonality constraints. Second, although the objective function value tends to be non-increasing in the early stages, it is not exactly monotonically non-increasing. These problems are caused by the difficulty of obtaining the optimal nonnegative column orthogonal factor matrices by constrained optimization. Mirzal (2014) pointed out the second drawback in the multiplicative algorithm for ONMF, and proposed a new convergent ONMF method using an additive updating algorithm; however, there is no guarantee that an orthogonal factor matrix will be obtained. Kimura et al. (2014) proposed a new ONMF method using a hierarchical alternating least-squares algorithm; this algorithm is faster than the previous multiplicative algorithms, but still has these two drawbacks. On the other hand, Pompili et al. (2014) proposed a k -means like method for ONMF, which exactly maintains orthogonality and ensures monotonicity for the objective function. Pompili et al. (2014) found that the optimization problem of ONMF is similar to that of spherical k -means introduced by Banerjee et al. (2003), and referred to this problem as a weighted variant of the spherical k -means (WSK). This is an ONMF method, not ONMTF, and its error distribution is normal. Thus, we extend it to the ONMTF methods based on normal and Tweedie family distributions. Of course, our ONMTF method guarantees orthogonality and the monotonically non-increasing property of the objective function value.

Several biclustering methods based on k -means have been proposed in the literature as double k -means (Vichi 2001; Wang et al. 2011; Van Mechelen et al. 2004). These methods consider the following hard clustering problem: each object belongs to exactly one cluster, and only belongs or does not belong, like 0 or 1. Although we

also consider a hard clustering problem, our proposed methods identify a membership degree. In that sense, our methods are somewhat more flexible than the double k -means. Li and Peng (2005) relaxed the hard double k -means method using orthogonal constrained factor matrices instead of indicator matrices. However, they did not impose nonnegativity constraints on these matrices. Moreover, these former double k -means approaches did not consider non-normal error distributions. The contribution of our work is the proposal of flexible hard bi-clustering approaches that consider the nonnegativity constraints and robust estimation with respect to extremely large positive values.

First, we introduce ONMTF-N, our new ONMTF method based on normal error distribution; this method is an ONMTF version of ONMF by Pompili et al. (2014). We then introduce two other new ONMTF methods (ONMTF-P and ONMTF-CP), which are respectively based on Poisson and CP error distributions. Poisson distribution is often used for integer data, e.g., count data, especially in the linear regression framework on the assumption that the expected value and the variance are equal; thereby, ONMTF-P can also be applied for count data, such as a contingency table. On the other hand, ONMTF-CP can be applied to nonnegative data that are collected as a summation of nonnegative values, such as a matrix containing the purchase volumes of individual customers in a store. To the best of our knowledge, ONMTF based on a CP distribution has not been proposed in any previous studies. Our two simulation studies demonstrate the increased consistency of ONMTF-N (compared to ONMTF based on the multiplicative algorithm), and the robustness of ONMTF-CP. In addition, we apply our ONMTF to document-term matrix to examine its goodness of clustering.

Because the previous methods are not relatively accurate, as shown in a simulation described later on, the estimates obtained by these methods can result in misinterpretations. However, our methods are more reliable because the estimates obtained by our methods are more accurate as compared to the previous methods. Moreover, our methods, especially ONMTF-P and ONMTF-CP, can be applied to real-world data because nonnegative data cannot be assumed to have a normal distribution; for example, purchase volume data contains a few extremely large positive values, which indicates that a large volume was purchased by only a few people. In such situations, a right-tailed distribution is more appropriate than a normal distribution. Therefore, our methods can significantly contribute in solving problems encountered in real-world data.

The notation employed in this paper is as follows. A matrix is represented in upper-case (bold type), e.g., $\mathbf{M}$; its i, j element is represented in lowercase, e.g., m_{ij} . An element of a complicated matrix is expressed as $[\cdot]_{ij}$. Further, $\mathbf{m}_i$ and $\mathbf{m}_{(j)}$ are column vectors with elements m_{ij} of the i -th row and the j -th column of $\mathbf{M}$, respectively. We use the prime symbol and ‘ -1 ’ to express a transpose matrix and inverse matrix, e.g., $\mathbf{M}'$ and $\mathbf{M}^{-1}$, respectively. The trace and diagonal parts of a square matrix $\mathbf{M}$ are denoted by $\text{tr}(\mathbf{M})$ and $\text{diag}(\mathbf{M})$, respectively. The Euclidean norm of a matrix or vector is represented as $\|\mathbf{M}\| = \sqrt{\text{tr}(\mathbf{M}'\mathbf{M})}$. $\mathbf{D}_{\mathbf{M}}$ is a diagonal matrix in which each diagonal element is $\|\mathbf{m}_{(j)}\|$, while $\Delta(\mathbf{M})$ is the vector with the absolute values of elements in eigenvector $\mathbf{v}$ with the largest eigenvalue of square matrix $\mathbf{M}$: $\mathbf{M}\mathbf{v} = \lambda\mathbf{v}$. Finally, $\mathbb{R}_+^{n \times p}$ is a set of $n \times p$ nonnegative matrices.

We introduce herein our bi-clustering problem as a motivation for our ONMTF models, which are presented in later sections, and discuss the connection between the

bi-clustering problem and ONMTF. Let y_{ij} be the i, j element of a nonnegative data matrix $Y \in \mathbb{R}_+^{n \times p}$ and $R = \{r_1, \dots, r_n\}$ and $C = \{c_1, \dots, c_p\}$ be the sets of the n row and p column objects, respectively. One of our aims is to detect a k -partition $\mathcal{R} = \{R_1, \dots, R_k\}$ of R and an ℓ -partition $\mathcal{C} = \{C_1, \dots, C_\ell\}$ of C , where R_m and C_q are the m - and q -th classes of $\mathcal{R}$ and $\mathcal{C}$, respectively. Classes R_m and C_q are defined as sets of row and column objects, respectively, and $\mathcal{R}$ and $\mathcal{C}$ are sets of k and ℓ disjoint non-empty classes that cover R and C , respectively. This definition implies that an object that belongs to one class does not belong to any other classes (i.e., when $r_i \in R_m$ then $r_i \notin R_{m^*}$ for all $m^* \neq m$ and when $c_j \in C_q$ then $c_j \notin C_{q^*}$ for all $q^* \neq q$). We refer to R_m as an “ m -th row cluster” and C_q as a “ q -th column cluster.” We consider the so-called membership of objects for a cluster to which each of the objects belong to. Let $F = (f_{im})$ be the $n \times k$ membership matrix of row objects such that $f_{im} > 0$ when $r_i \in R_m$, while $f_{im} = 0$ when $r_i \notin R_m$. Let $A = (a_{jq})$ be the $p \times \ell$ membership matrix of column objects such that $a_{jq} > 0$ when $c_j \in C_q$, while $a_{jq} = 0$ when $c_j \notin C_q$. Note that this definition leads to the orthogonality of F and A : $f'_{(m)} f_{(m^*)} = 0$ for all $m^* \neq m$ and $a'_{(q)} a_{(q^*)} = 0$ for all $q^* \neq q$. If $f_{(m)}$ and $a_{(q)}$ for all m, q have the unit length, the orthogonality is changed to orthonormality: $F'F = I_k$ and $A'A = I_\ell$. We also consider a relationship between row and column clusters. Let $S = (s_{mq})$ be a $k \times \ell$ matrix such that $s_{mq} > 0$ for all entries. This study aims to estimate the best unknown parameters $\theta = \{\mathcal{R}, \mathcal{C}, F, A, S\}$ by given Y . We consider the following approximation problem for this aim: the best θ is obtained such that for all i, j there exist $m : R_m \ni r_i$ and $q : C_q \ni c_j$ such that $y_{ij} \approx x_{ij} = f_{im}s_{mq}a_{jq}$. From the definition of the membership matrices F and A , we can rewrite the approximation problem as $y_{ij} \approx x_{ij} = \sum_{i=1}^n \sum_{j=1}^p f_{im}s_{mq}a_{jq}$. In matrix forms, we can describe it as $Y \approx X = FSA'$. Note that F and A have orthogonality, and all entries in all matrices are nonnegative. Hence, the approximation problem is equivalent to the ONMTF problem. We consider herein that the numbers of classes k and ℓ are chosen in advance.

2 Orthogonal NMTF based on normal error distribution

In this section, a new method for NMTF, namely, ONMTF-N, is introduced. This method is based on the WWSK algorithm proposed by Pompili et al. (2014); it is a fundamental algorithm for ONMTF methods described in the following sections, where it is assumed that data follows Poisson and CP distributions.

To formulate the bi-clustering problem as the approximation problem described in Sect. 1, we define the objective function by the following Euclidean distance:

$$Q_N(\theta) = d_{\text{EUC}}(Y, FSA') = \|Y - FSA'\|^2/2, \quad (1)$$

and define ONMTF-N as the following optimization problem with respect to θ :

$$\begin{aligned} & \min Q_N(\theta) \\ & \text{subject to } F \in \mathbb{R}_+^{n \times k}, S \in \mathbb{R}_+^{k \times \ell}, A \in \mathbb{R}_+^{p \times \ell}, F'F = I_k, A'A = I_\ell. \end{aligned} \quad (2)$$

We set the two orthogonal constraints for $\mathbf{F}$ and $\mathbf{A}$ according to the discussion in Sect. 1.

This optimization problem is formally derived from a maximum likelihood (ML) problem under the assumption of normality: $y_{ij} \sim N(x_{ij}, \sigma^2)$ for all i, j independently, where $N(\mu, \sigma^2)$ is a normal distribution with mean μ and variance σ^2 , which is why we named the method “ONMTF-N.” However, this assumption is incompatible because we consider that the given data matrix only contains nonnegative entries. For this issue, we start with (2) to describe the ONMTF-N problem. The explanation on the other two methods that will be described later starts with the ML problem.

The objective function (1) is invariant under the changing of length of each row vector of $\mathbf{S}$ because the following is satisfied:

$$2Q_N = \|\mathbf{Y} - \mathbf{F}\mathbf{S}\mathbf{A}'\|^2 = \|\mathbf{Y} - \mathbf{F}\mathbf{D}_{\mathbf{S}'}\mathbf{D}_{\mathbf{S}'}^{-1}\mathbf{S}\mathbf{A}'\|^2 = \|\mathbf{Y} - \mathbf{F}^*\mathbf{S}^*\mathbf{A}'\|^2, \quad (3)$$

where $\mathbf{D}_{\mathbf{S}'}$ is the $k \times k$ diagonal matrix in which each diagonal element is $\|\mathbf{s}_m\|$ ($m = 1, \dots, k$), $\mathbf{F}^* = \mathbf{F}\mathbf{D}_{\mathbf{S}'}$, and $\mathbf{S}^* = \mathbf{D}_{\mathbf{S}'}^{-1}\mathbf{S}$. Now,

$$\text{diag}(\mathbf{S}^*\mathbf{S}^{*\prime}) = \mathbf{I}_k \quad (4)$$

is satisfied and $\mathbf{f}_{(m)}^*$ ($m = 1, \dots, k$) no longer has to be a unit length. According to (3) and (4), the optimization problem (2) is the same as

$$\begin{aligned} \min Q_N(\theta) \text{ subject to } & \mathbf{F} \in \mathbb{R}_+^{n \times k}, \mathbf{S} \in \mathbb{R}_+^{k \times \ell}, \mathbf{A} \in \mathbb{R}_+^{p \times \ell}, \text{diag}(\mathbf{S}\mathbf{S}') = \mathbf{I}_k, \\ & \mathbf{f}_{(m)}' \mathbf{f}_{(m^*)} = 0 \ (m \neq m^*), \mathbf{A}'\mathbf{A} = \mathbf{I}_\ell. \end{aligned} \quad (5)$$

The parameters are updated using alternating optimization. First, we derive the update equation of $\mathbf{F}$ given $\mathcal{R}$, $\mathbf{S}$, and $\mathbf{A}$. Note that (1) can be expanded such that

$$\begin{aligned} 2Q_N &= \sum_{i=1}^n \left\| \mathbf{y}_i - \sum_{m=1}^k f_{im} \mathbf{A} \mathbf{s}_m \right\|^2 = \sum_{m=1}^k \sum_{r_i \in R_m} \left\| \mathbf{y}_i - f_{im} \mathbf{A} \mathbf{s}_m \right\|^2 \\ &= \sum_{m=1}^k \sum_{r_i \in R_m} \left\{ \mathbf{y}_i' \mathbf{y}_i - 2f_{im} \mathbf{y}_i' \mathbf{A} \mathbf{s}_m + f_{im}^2 \mathbf{s}_m' \mathbf{A}' \mathbf{A} \mathbf{s}_m \right\} \\ &= \sum_{m=1}^k \sum_{r_i \in R_m} \left\{ \mathbf{y}_i' \mathbf{y}_i - 2f_{im} \mathbf{y}_i' \mathbf{A} \mathbf{s}_m + f_{im}^2 \right\} \quad (\because (4) \text{ and } \mathbf{A}'\mathbf{A} = \mathbf{I}_\ell). \end{aligned} \quad (6)$$

Hence, the minimizer of f_{im} for (6) is

$$f_{im} = \mathbf{y}_i' \mathbf{A} \mathbf{s}_m \quad (r_i \in R_m; m = 1, \dots, k). \quad (7)$$

Substituting (7) into (6) and rearranging the terms proportional to the parameters, we obtain

$$Q_N^*(\mathcal{R}, \mathbf{S}) = \sum_{m=1}^k \sum_{r_i \in R_m} \left\{ -(\mathbf{y}_i' \mathbf{A} \mathbf{s}_m)^2 \right\}. \quad (8)$$

Therefore, the problem of minimizing Q_N is the same as the problem of minimizing Q_N^* . Note that (8) can be observed to be similar to the objective function of WVS K when $A'y_i$ ($i = 1, \dots, n$) and s_m ($m = 1, \dots, k$) are considered as n data vectors and k normalized centroids in $\mathbb{R}_+^\ell$, respectively. In addition, when all n data vectors are normalized (i.e., $\|A'y_i\|^2 = 1$ ($i = 1, \dots, n$)), (8) is observed to be the same as the objective function of spherical k -means in $\mathbb{R}_+^\ell$. Given S and A , the minimizers of $\mathcal{R}$ for Q_N^* are derived by, e.g., a k -means algorithm such that

$$R_m = \left\{ r_i \mid \operatorname{argmax}_{m^*=1, \dots, k} (y_i' A s_{m^*}) = m \right\} \quad (m = 1, \dots, k). \quad (9)$$

Because (8) can be rewritten such that

$$Q_N^* = - \sum_{m=1}^k \|Y_m A s_m\|^2 = - \sum_{m=1}^k \operatorname{tr} \{ s_m' A' Y_m' Y_m A s_m \}, \quad (10)$$

where Y_m ($m = 1, \dots, k$) is a $|R_m| \times p$ matrix consisting of the row vectors y_i' , such that $r_i \in R_m$. The minimizer of S given $\mathcal{R}$ and A can be obtained by an eigenvector of $A' Y_m' Y_m A$ as follows:

$$s_m = \Delta(A' Y_m' Y_m A) \quad (m = 1, \dots, k). \quad (11)$$

Finally, the minimizer of F given $\mathcal{R}$, S , and A can be expressed from (7) as

$$f_{im} = \begin{cases} y_i' A s_m & (r_i \in R_m) \\ 0 & (r_i \notin R_m) \end{cases} \quad (i = 1, \dots, n; \quad m = 1, \dots, k). \quad (12)$$

If we regard the model as $Y' \approx A S' F'$, the update equations of $\mathcal{C}$, S , and A can be derived similarly to (9), (11), and (12) as follows:

$$C_q = \left\{ c_j \mid \operatorname{argmax}_{q^*=1, \dots, \ell} \{ y_{(j)}' F s_{(q^*)} \} = q \right\} \quad (q = 1, \dots, \ell) \quad (13)$$

$$s_{(q)} = \Delta(F' Y_{(q)} Y_{(q)}' F) \quad (q = 1, \dots, \ell) \quad (14)$$

$$a_{jq} = \begin{cases} y_{(j)}' F s_{(q)} & (c_j \in C_q) \\ 0 & (c_j \notin C_q) \end{cases} \quad (j = 1, \dots, p; \quad q = 1, \dots, \ell), \quad (15)$$

where $Y_{(q)}$ ($q = 1, \dots, \ell$) is an $n \times |C_q|$ matrix consisting of the column vectors $y_{(j)}$, such that $c_j \in C_q$.

From (9), (11), (12), (13), (14) and (15), we can formulate an alternating optimization algorithm for model ONMTF-N as **Algorithm 2.1** given below: Note that lines 7 and 8 mean the normalization of the length of each column vector of F and the

modification of the length of each row vector of S maintaining the objective function value, respectively. These two steps are needed because the update of C_q in step 9 is under the assumption of $F'F = I_k$. This holds for steps 12 and 13.

Algorithm 2.1 ONMTF-N

```

1: Input  $Y$  and stopping threshold  $\tau$ .
2: Initialize  $S$  and  $A$  as satisfying  $A'A = I_\ell$ .
3: repeat
4:    $R_m \leftarrow \left\{ r_i \mid \operatorname{argmax}_{m^*=1,\dots,k} \{y'_i A s_{m^*}\} = m \right\} \quad (m = 1, \dots, k)$ 
5:    $s_m \leftarrow \Delta(A'Y'_m) \quad (m = 1, \dots, k)$ 
6:    $f_{im} \leftarrow \begin{cases} y'_i A s_m & (r_i \in R_m) \\ 0 & (r_i \notin R_m) \end{cases} \quad (i = 1, \dots, n; m = 1, \dots, k)$ 
7:    $S \leftarrow D_F S$ 
8:    $F \leftarrow F D_F^{-1}$ 
9:    $C_q \leftarrow \left\{ c_j \mid \operatorname{argmax}_{q^*=1,\dots,\ell} \{y'_{(j)} F s_{(q^*)}\} = q \right\} \quad (q = 1, \dots, \ell)$ 
10:   $s_{(q)} \leftarrow \Delta(F'Y_{(q)}) \quad (q = 1, \dots, \ell)$ 
11:   $a_{jq} \leftarrow \begin{cases} y'_{(j)} F s_{(q)} & (c_j \in C_q) \\ 0 & (c_j \notin C_q) \end{cases} \quad (j = 1, \dots, p; q = 1, \dots, \ell)$ 
12:   $S \leftarrow S D_A$ 
13:   $A \leftarrow A D_A^{-1}$ 
14:   $Q \leftarrow d_{\text{EUC}}(Y, FSA')$ 
15: until the difference between the previous and the current  $Q$  is less than  $\tau$ .
  
```

In some former double k -means approaches (e.g., Vichi (2001); Wang et al. (2011); Li and Peng (2005)), the middle factor matrix S is updated such that $S = F'YA$ or $S = (F'F)^{-1}F'YA(A'A)^{-1}$. We can utilize this update in our methods, and it may lead to a better performance of the estimation accuracy or the computation time. However, we adopt the two-time update described in **Algorithm 2.1** to emphasize ONMTF-N as an expansion of the WVSF (Pompili et al. 2014). This expansion provides the idea of ONMTF-P and ONMTF-CP introduced in the sections that follow. Approaches for updating S within the algorithm will be investigated in the future.

3 Orthogonal NMTF based on Poisson error distribution

In this section, we introduce a new ONMTF method, namely, ONMTF-P. It is a modified version of ONMTF-N described in Sect. 2, and it assumes that data follow a Poisson distribution. Although a multiplicative updating algorithm for NMTF under this assumption was proposed by Yoo and Choi (2009), the orthogonal constraints were not imposed on it. In contrast, our algorithm is not based on a multiplicative updating algorithm but on the WVSF algorithm, and the orthogonality constraints are imposed on it. Unfortunately, we only provide the model and the algorithm of ONMTF-P because of space limitations. For the derivation of the update equations,

please see Abe H, Yadohisa H (2017) Supplementary material to “Orthogonal non-negative matrix tri-factorization based on Tweedie distributions.”.

In formulating the bi-clustering problem described in Sect. 1, we consider y_{ij} as the independent Poisson distributed random variables with density:

$$f(y_{ij}|x_{ij}) = x_{ij}^{y_{ij}} \exp \{-x_{ij}\} / y_{ij}! \quad (i = 1, \dots, n; \quad j = 1, \dots, p), \quad (16)$$

where $x_{ij} = \sum_{i=1}^n \sum_{j=1}^p f_{im} s_{mq} a_{jq}$. The objective function to be minimized is derived from (16) by an ML procedure as follows:

$$\begin{aligned} Q_P(\theta) &= d_{KL}(\mathbf{Y}, \mathbf{FSA}') = \sum_{i=1}^n \sum_{j=1}^p (y_{ij} \log(y_{ij}/x_{ij}) - y_{ij} + x_{ij}) \\ &= \sum_{i=1}^n \sum_{j=1}^p \left[-y_{ij} \log \left\{ \sum_{m=1}^k \sum_{q=1}^{\ell} f_{im} s_{mq} a_{jq} \right\} + \sum_{m=1}^k \sum_{q=1}^{\ell} f_{im} s_{mq} a_{jq} \right] + \text{cst.} \end{aligned} \quad (17)$$

where $d_{KL}(\cdot, \cdot)$ is KL-divergence, and “cst” is a constant with respect to θ to be estimated. Hence, the optimization problem of ONMTF-P is

$$\begin{aligned} \min Q_P(\theta) \text{ subject to } & \mathbf{F} \in \mathbb{R}_+^{n \times k}, \quad \mathbf{S} \in \mathbb{R}_+^{k \times \ell}, \quad \mathbf{A} \in \mathbb{R}_+^{p \times \ell}, \\ & \mathbf{f}'_{(m)} \mathbf{f}_{(m^*)} = 0 \quad (m \neq m^*), \quad \mathbf{a}'_{(q)} \mathbf{a}_{(q^*)} = 0 \quad (q \neq q^*). \end{aligned} \quad (18)$$

The two constraints at the bottom are from the bi-clustering problem described in Sect. 1. The procedure to solve the optimization problem (18) of ONMTF-P is completed as **Algorithm 3.1**. Here, the λ_m ($m = 1, \dots, k$) and λ_q^* ($q = 1, \dots, \ell$) are defined as:

$$\lambda_m = \sum_{\gamma=1}^p \sum_{q=1}^{\ell} s_{mq} a_{\gamma q} \quad \text{and} \quad \lambda_q^* = \sum_{i=1}^n \sum_{m=1}^k s_{mq} f_{im}. \quad (19)$$

4 Orthogonal NMTF based on CP error distribution

In this section, we introduce the other new method for ONMTF, namely, ONMTF-CP, where it is assumed that data follows a CP distribution. This method is also based on the WVK algorithm, as in the case of ONMTF-N and ONMTF-P described in Sects. 2 and 3, respectively. ONMTF-CP has a hyperparameter β that determines the robustness of estimation against the extremely large positive values. It is noteworthy that we derive a new auxiliary function for updating the middle factor matrix $\mathbf{S}$.

In formulating the bi-clustering problem described in Sect. 1, we consider y_{ij} as the independent CP distributed random variables with density:

$$f(y_{ij}|x_{ij}, \phi, \beta) = h_{\beta}(y_{ij}, \phi) \exp \left\{ \frac{1}{\phi} \left(\frac{1}{\beta - 1} y_{ij} x_{ij}^{\beta-1} - \frac{1}{\beta} x_{ij}^{\beta} \right) \right\}$$

Algorithm 3.1 ONMTF-P

1: Input $\mathbf{Y}$ and stopping threshold τ .
 2: Initialize $\mathbf{A}$ and $\mathbf{S}$ as satisfying $\mathbf{a}'_{(q)} \mathbf{a}_{(q^*)} = 0$ ($q \neq q^*$).
 3: **repeat**
 4: $R_m \leftarrow \left\{ r_i \mid \underset{m^*=1, \dots, k}{\operatorname{argmax}} \left\{ \sum_{j=1}^p y_{ij} \log \left\{ \frac{[\mathbf{S}\mathbf{A}']_{m^*j}}{\sum_{\gamma=1}^p [\mathbf{S}\mathbf{A}']_{m^*\gamma}} \right\} \right\} = m \right\}$ ($m = 1, \dots, k$)
 5: $s_{mq} \leftarrow \frac{\lambda_m \sum_{r_i \in R_m} \sum_{c_j \in C_q} y_{ij}}{\left(\sum_{c_j \in C_q} \sum_{i=1}^n y_{ij} \right) \left(\sum_{i=1}^n f_{im} \right)}$ ($m = 1, \dots, k; q = 1, \dots, \ell$)
 6: $f_{im} \leftarrow \begin{cases} \frac{\sum_{j=1}^p y_{ij}}{\sum_{j=1}^p [\mathbf{S}\mathbf{A}']_{mj}} & (r_i \in R_m) \\ 0 & (r_i \notin R_m) \end{cases}$ ($i = 1, \dots, n; m = 1, \dots, k$)
 7: $C_q \leftarrow \left\{ c_j \mid \underset{q^*=1, \dots, \ell}{\operatorname{argmax}} \left\{ \sum_{i=1}^n y_{ij} \log \left\{ \frac{[\mathbf{F}\mathbf{S}]_{iq^*}}{\sum_{v=1}^n [\mathbf{F}\mathbf{S}]_{vq^*}} \right\} \right\} = q \right\}$ ($q = 1, \dots, \ell$)
 8: $s_{mq} \leftarrow \frac{\lambda_q^* \sum_{r_i \in R_m} \sum_{c_j \in C_q} y_{ij}}{\left(\sum_{r_i \in R_m} \sum_{j=1}^p y_{ij} \right) \left(\sum_{j=1}^p a_{jq} \right)}$ ($m = 1, \dots, k; q = 1, \dots, \ell$)
 9: $a_{jq} \leftarrow \begin{cases} \frac{\sum_{i=1}^n y_{ij}}{\sum_{i=1}^n [\mathbf{F}\mathbf{S}]_{iq}} & (c_j \in C_q) \\ 0 & (c_j \notin C_q) \end{cases}$ ($j = 1, \dots, p; q = 1, \dots, \ell$)
 10: $Q \leftarrow d_{\text{KL}}(\mathbf{Y}, \mathbf{F}\mathbf{S}\mathbf{A}')$
 11: **until** the difference between the previous and the current Q is less than τ .

$$(i = 1, \dots, n; j = 1, \dots, p), \quad (20)$$

where $x_{ij} = \sum_{i=1}^n \sum_{j=1}^p f_{im} s_{mq} a_{jq}$, ϕ is a dispersion parameter, and $\beta \in (0, 1)$ is an index parameter that determines the shape of the distribution. The objective function to be minimized is derived from (20) by a ML procedure as follows:

$$\begin{aligned} Q_{\text{CP}}(\theta) = d_{\beta}(\mathbf{Y}, \mathbf{X}) &= \sum_{i=1}^n \sum_{j=1}^p \left(y_{ij} \frac{y_{ij}^{\beta-1} - x_{ij}^{\beta-1}}{\beta - 1} - \frac{y_{ij}^{\beta} - x_{ij}^{\beta}}{\beta} \right) \\ &= \frac{1}{\beta} \sum_{i=1}^n \sum_{j=1}^p \left(\sum_{m=1}^k \sum_{q=1}^{\ell} f_{im} s_{mq} a_{jq} \right)^{\beta} \\ &\quad - \frac{1}{\beta - 1} \sum_{i=1}^n \sum_{j=1}^p y_{ij} \left(\sum_{m=1}^k \sum_{q=1}^{\ell} f_{im} s_{mq} a_{jq} \right)^{\beta-1} + \text{cst.} \end{aligned} \quad (21)$$

where $d_{\beta}(\cdot, \cdot)$ is a β -divergence with parameter β that corresponding to the index parameter of CP (20). Hence, the optimization problem of ONMTF-CP is

$$\begin{aligned} \min Q_{\text{CP}}(\theta) \text{ subject to } &\mathbf{F} \in \mathbb{R}_+^{n \times k}, \mathbf{S} \in \mathbb{R}_+^{k \times \ell}, \mathbf{A} \in \mathbb{R}_+^{p \times \ell}, \\ &\mathbf{f}'_{(m)} \mathbf{f}_{(m^*)} = 0 \ (m \neq m^*), \mathbf{a}'_{(q)} \mathbf{a}_{(q^*)} = 0 \ (q \neq q^*). \end{aligned} \quad (22)$$

The two constraints at the bottom are from the bi-clustering problem described in Sect. 1.

The parameters are updated using alternating optimization. First, we derive the update equation of $\mathbf{F}$ given $\mathcal{R}$, $\mathbf{S}$, and $\mathbf{A}$. Note that (21) can be rewritten as

$$\begin{aligned} Q_{\text{CP}} &= \sum_{i=1}^n \sum_{j=1}^p \left\{ \frac{1}{\beta} \left(\sum_{m=1}^k f_{im} [\mathbf{SA}']_{mj} \right)^{\beta} \right. \\ &\quad \left. - \frac{1}{\beta-1} y_{ij} \left(\sum_{m=1}^k f_{im} [\mathbf{SA}']_{mj} \right)^{\beta-1} \right\} + \text{cst} \\ &= \sum_{m=1}^k \sum_{r_i \in R_m} \left\{ \frac{1}{\beta} f_{im}^{\beta} \sum_{j=1}^p [\mathbf{SA}']_{mj}^{\beta} - \frac{1}{\beta-1} f_{im}^{\beta-1} \sum_{j=1}^p y_{ij} [\mathbf{SA}']_{mj}^{\beta-1} \right\} + \text{cst}. \end{aligned} \quad (23)$$

Hence, the minimizer of f_{im} for (23) is

$$f_{im} = \frac{\sum_{j=1}^p y_{ij} [\mathbf{SA}']_{mj}^{\beta-1}}{\sum_{j=1}^p [\mathbf{SA}']_{mj}^{\beta}} \quad (r_i \in R_m \ m = 1, \dots, k). \quad (24)$$

Substituting (24) into (23) and rearranging the terms proportional to the parameters, we obtain

$$Q_{\text{CP}}^*(\mathcal{R}, \mathbf{S}) = -\frac{1}{\beta(\beta-1)} \sum_{m=1}^k \sum_{r_i \in R_m} \frac{\left\{ \sum_{j=1}^p y_{ij} [\mathbf{SA}']_{mj}^{\beta-1} \right\}^{\beta}}{\left\{ \sum_{j=1}^p [\mathbf{SA}']_{mj}^{\beta} \right\}^{\beta-1}}. \quad (25)$$

Therefore, the problem of minimizing Q_{CP} is the same as the problem of minimizing Q_{CP}^* . Given $\mathbf{S}$ and $\mathbf{A}$, the minimizers of $\mathcal{R}$ for Q_{CP}^* are derived by, e.g., a k -means algorithm such that

$$R_m = \left\{ r_i \left| \operatorname{argmin}_{m^*=1, \dots, k} \left\{ \frac{\left\{ \sum_{j=1}^p y_{ij} [\mathbf{SA}']_{m^*j}^{\beta-1} \right\}^{\beta}}{\left\{ \sum_{j=1}^p [\mathbf{SA}']_{m^*j}^{\beta} \right\}^{\beta-1}} \right\} = m \right. \right\} \quad (m = 1, \dots, k). \quad (26)$$

Then, we derive the optimal $\mathbf{S}$ for $Q_{\text{CP}}^*(\mathcal{R}, \mathbf{S})$ given $\mathcal{R}$ and $\mathbf{A}$. Note that (25) is rewritten as

$$Q_{\text{CP}}^* = -\frac{1}{\beta(\beta-1)} \sum_{m=1}^k \sum_{r_i \in R_m} \frac{\left\{ \sum_{q=1}^{\ell} \sum_{c_j \in C_q} y_{ij} s_{mq}^{\beta-1} a_{jq}^{\beta-1} \right\}^{\beta}}{\left\{ \sum_{q=1}^{\ell} \sum_{c_j \in C_q} s_{mq}^{\beta} a_{jq}^{\beta} \right\}^{\beta-1}}. \quad (27)$$

It is difficult to directly obtain the minimizer of s_{mq} for (27) because the summation of s_{mq} is in the power function. However, we can obtain the optimal s_{mq} using the

auxiliary function method, as in the case of ONMTF-P. First, we can find that the following bivariate function

$$f(x, y) = -\frac{1}{\beta(\beta-1)}(y^\beta/x^{\beta-1}) \quad ((x, y) \in \mathbb{R}_+^2; \beta \in (0, 1)) \quad (28)$$

is involved in equation (27). We need the following corollary to obtain the appropriate inequality for (28):

Corollary 1 *The bivariate function $f(x, y)$ in (28) is concave if $(x, y) \in \mathbb{R}_+^2$ and $\beta > 0$.*

Proof The first- and second-order differentials of $f(x, y)$ are as follows:

$$f_x(x, y) = (1/\beta)y^\beta x^{-\beta}, \quad (29)$$

$$f_y(x, y) = (1/(1-\beta))y^{\beta-1}x^{1-\beta}, \quad (30)$$

$$f_{xx}(x, y) = -y^\beta x^{-\beta-1}, \quad (31)$$

$$f_{yy}(x, y) = -y^{\beta-2}x^{1-\beta}, \text{ and} \quad (32)$$

$$f_{xy}(x, y) = y^{\beta-1}x^{-\beta}. \quad (33)$$

From (31), (32), and (33), we have

$$f_{xx}(x, y) < 0, \quad f_{yy}(x, y) < 0, \text{ and } f_{xx}(x, y)f_{yy}(x, y) - \{f_{xy}(x, y)\}^2 = 0. \quad (34)$$

Hence, the function $f(x, y)$ is concave. $\square$

Therefore, we have

$$f(x, y) \leq f(\lambda, \eta) + f_x(\lambda, \eta)(x - \lambda) + f_y(\lambda, \eta)(y - \eta) \quad (35)$$

for any λ and η with equality if and only if $x = \lambda$ and $y = \eta$. From this inequality, we obtain the following auxiliary function of (27) for s_{mq} :

$$\begin{aligned} Q_{\text{CP}}^{\text{aux}}(\mathbf{S}, \mathbf{S}^*) &= \sum_{m=1}^k \sum_{r_i \in R_m} \left\{ -\frac{1}{\beta(\beta-1)} \frac{\eta_{im}^\beta}{\lambda_m^{\beta-1}} + \frac{1}{\beta} \left(\frac{\eta_{im}}{\lambda_m} \right)^\beta \left(\sum_{q=1}^\ell \sum_{c_j \in C_q} s_{mq}^\beta a_{jq}^\beta - \lambda_m \right) \right. \\ &\quad \left. + \frac{1}{1-\beta} \left(\frac{\eta_{im}}{\lambda_m} \right)^{\beta-1} \left(\sum_{q=1}^\ell \sum_{c_j \in C_q} y_{ij} s_{mq}^{\beta-1} a_{jq}^{\beta-1} - \eta_{im} \right) \right\} \end{aligned} \quad (36)$$

with

$$\lambda_m = \sum_{q=1}^\ell \sum_{c_j \in C_q} s_{mq}^* a_{jq}^\beta, \quad \text{and} \quad \eta_{im} = \sum_{q=1}^\ell \sum_{c_j \in C_q} y_{ij} s_{mq}^{*\beta-1} a_{jq}^{\beta-1}$$

$$(i = 1, \dots, n; m = 1, \dots, k), \quad (37)$$

where s_{mq}^* is the current value of s_{mq} . Let $Q_{CP}^*(S)$ be the same function as $Q_{CP}^*(\mathcal{R}, S)$. We then have:

$$Q_{CP}^{\text{aux}}(S, S^*) \geq Q_{CP}^*(S) \quad \text{and} \quad Q_{CP}^{\text{aux}}(S, S) = Q_{CP}^*(S) \quad (38)$$

for any S and $S^* = (s_{mq}^*)$, and if we can find $\hat{S} = \underset{S}{\operatorname{argmin}} Q_{CP}^{\text{aux}}(S, S^*)$, we have

$$Q_{CP}^*(S^*) = Q_{CP}^{\text{aux}}(S^*, S^*) \geq Q_{CP}^{\text{aux}}(\hat{S}, S^*) \geq Q_{CP}^*(\hat{S}). \quad (39)$$

Hence, the minimizer of S for (36) is at least the optimal S , which is monotonically non-increasing for (27). We obtain the following update equation of s_{mq} as a minimizer for $Q_{CP}^{\text{aux}}(S, S^*)$:

$$s_{mq} = \frac{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{im}/\lambda_m)^{\beta-1} y_{ij} a_{jq}^{\beta-1}}{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{im}/\lambda_m)^{\beta} a_{jq}^{\beta}}, \quad (40)$$

Finally, the minimizer of F given $\mathcal{R}, S$, and A can be expressed from (24) as follows:

$$f_{im} = \begin{cases} \frac{\sum_{j=1}^p y_{ij} [\mathbf{SA}']_{mj}^{\beta-1}}{\sum_{j=1}^p [\mathbf{SA}']_{mj}^{\beta}} & (r_i \in R_m) \\ 0 & (r_i \notin R_m) \end{cases} \quad (i = 1, \dots, n; m = 1, \dots, k). \quad (41)$$

If we regard the model as $\mathbf{Y}' \approx \mathbf{AS}'\mathbf{F}'$, the update equations of $\mathcal{C}, S$, and A can be derived similarly to (26), (40), and (41) as follows:

$$C_q = \left\{ c_j \left| \underset{q^*=1, \dots, \ell}{\operatorname{argmin}} \left\{ \frac{\left\{ \sum_{i=1}^n y_{ij} [\mathbf{FS}]_{iq^*}^{\beta-1} \right\}^{\beta}}{\left\{ \sum_{i=1}^n [\mathbf{FS}]_{iq^*}^{\beta} \right\}^{\beta-1}} \right\} = q \right. \right\} \quad (q = 1, \dots, \ell), \quad (42)$$

$$s_{mq} = \frac{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{jq}^*/\lambda_q^*)^{\beta-1} y_{ij} f_{im}^{\beta-1}}{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{jq}^*/\lambda_q^*)^{\beta} f_{im}^{\beta}} \quad (m = 1, \dots, k; q = 1, \dots, \ell)$$

$$\text{with } \lambda_q^* = \sum_{m=1}^k \sum_{r_i \in R_m} s_{mq}^* f_{im}^{\beta}, \quad \eta_{jq}^* = \sum_{m=1}^k \sum_{r_i \in R_m} y_{ij} s_{mq}^{*\beta-1} f_{im}^{\beta-1} \quad (j = 1, \dots, p; q = 1, \dots, \ell), \quad (43)$$

$$a_{jq} = \begin{cases} \frac{\sum_{i=1}^n y_{ij} [\mathbf{FS}]_{iq}^{\beta-1}}{\sum_{i=1}^n [\mathbf{FS}]_{iq}^{\beta}} & (c_j \in C_q) \\ 0 & (c_j \notin C_q) \end{cases} \quad (j = 1, \dots, p; q = 1, \dots, \ell). \quad (44)$$

A novelty of this paper is the derivation of the updating equation for S using the auxiliary function obtained by an inequality of the bivariate concave function in ONMTF-CP. To the best of our knowledge, the corollary on the inequality and its proof have not yet been presented in previous studies.

From (26), (40), (41), (42), (43), and (44), we can formulate an alternating optimization algorithm for model ONMTF-CP as **Algorithm 4.1**. Here, λ_m , η_{im} , λ_q^* , and η_{jq}^* are defined as the same formulas in (37) and (44), in which s_{mq}^* is converted to s_{mq} .

Algorithm 4.1 ONMTF-CP

- 1: Input Y , β , and stopping threshold τ .
 - 2: Initialize A and S as satisfying $a'_{(q)} a_{(q^*)} = 0$ ($q \neq q^*$).
 - 3: **repeat**
 - 4: $R_m \leftarrow \left\{ r_i \mid \underset{m^*=1, \dots, k}{\operatorname{argmin}} \left\{ \frac{\left\{ \sum_{j=1}^p y_{ij} [SA']_{m^*j}^{\beta-1} \right\}^\beta}{\left\{ \sum_{j=1}^p [SA']_{m^*j}^\beta \right\}^{\beta-1}} \right\} = m \right\}$ ($m = 1, \dots, k$)
 - 5: $s_{mq} \leftarrow \frac{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{im}/\lambda_{im})^{\beta-1} y_{ij} (a_{jq})^{\beta-1}}{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{im}/\lambda_{im})^\beta (a_{jq})^\beta}$ ($m = 1, \dots, k; q = 1, \dots, \ell$)
 - 6: $f_{im} \leftarrow \begin{cases} \frac{\sum_{j=1}^p y_{ij} [SA']_{mj}^{\beta-1}}{\sum_{j=1}^p [SA']_{mj}^\beta} & (r_i \in R_m) \\ 0 & (i \notin R_m) \end{cases}$
 - 7: $C_q \leftarrow \left\{ c_j \mid \underset{q^*=1, \dots, \ell}{\operatorname{argmin}} \left\{ \frac{\left\{ \sum_{i=1}^n y_{ij} [FS]_{iq^*}^{\beta-1} \right\}^\beta}{\left\{ \sum_{i=1}^n [FS]_{iq^*}^\beta \right\}^{\beta-1}} \right\} = q \right\}$ ($q = 1, \dots, \ell$)
 - 8: $s_{mq} \leftarrow \frac{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{jq}^*/\lambda_{jq}^*)^{\beta-1} y_{ij} (f_{im})^{\beta-1}}{\sum_{r_i \in R_m} \sum_{c_j \in C_q} (\eta_{jq}^*/\lambda_{jq}^*)^\beta (f_{im})^\beta}$ ($m = 1, \dots, k; q = 1, \dots, \ell$)
 - 9: $a_{jq} \leftarrow \begin{cases} \frac{\sum_{i=1}^n y_{ij} [FS]_{iq}^{\beta-1}}{\sum_{i=1}^n [FS]_{iq}^\beta} & (c_j \in C_q) \\ 0 & (c_j \notin C_q) \end{cases}$
 - 10: $Q \leftarrow d_\beta(Y, FSA')$
 - 11: **until** the difference between previous and current Q is less than τ .
-

4.1 Some issues

The proposed ONMTF methods introduced earlier are based on the k -means algorithm from the row and column sides. In other words, these methods have the disadvantages of double k -means clustering: initialization, local minima, empty clusters. As for the initialization, we randomly assign each column object to a cluster in $\mathcal{C}$ and use an exponential random number as an initial value of a_{jq} ($c_j \in C_q$) for all j and all entries of S . Note that the initial $\mathcal{R}$ and F are not randomly given, but updated by the randomly initialized $\mathcal{C}$, A , and S . Concerning the local minima and empty clusters,

it is recommended to run these algorithms with efficiently large random starts. We restart the update iteration from another random start if the empty clusters occur. We select the estimates with the least objective function value from the non-empty cluster estimates. Any initialization with the k -means (e.g., Xue et al. (2008)) is not used in our proposed methods because of the two following reasons: 1) its initialization also needs initialization to compute and 2) its initialization does not always lead to the best estimates; therefore, we use large random starts.

5 Simulation studies

In this section, we describe two simulation studies. The first study compares ONMTF-N and previous ONMTF methods in terms of estimation accuracy. The second study analyzes the characteristics of the estimates given by ONMTF-N, ONMTF-P, and ONMTF-CP.

5.1 Simulation study 1: estimation accuracy of ONMTF-N

We conduct a simulation study to examine the estimation accuracy of ONMTF-N. We compare ONMTF-N with the previous ONMTF techniques proposed by Ding et al. (2006) and Yoo and Choi (2010b). We refer to their methods as “Ding’s method” and “Yoo’s method,” respectively. The details of the procedure are presented as follows: First, we generate synthetic data matrix as $Y = \tilde{F}^* \tilde{S}^* \tilde{A}^{*'} + E$, where $\tilde{F}^*$, $\tilde{S}^*$, and $\tilde{A}^*$ denote three nonnormalized true factor matrices generated as described below and E denotes the error matrix whose entries are normal random number from $N(0, \sigma^2)$. The entries are converted to zero if $y_{ij} < 0$. We generate each entries of these three nonnormalized true factor matrices $\tilde{F}^*$, $\tilde{S}^*$, and $\tilde{A}^*$ as follows:

$$\begin{cases} \tilde{f}_{im}^* \sim Ex(\mu^{1/3}) & (r_i \in \tilde{R}_m) \\ \tilde{f}_{im}^* = 0 & (r_i \notin \tilde{R}_m) \end{cases} \quad (i = 1, \dots, n; \quad m = 1, \dots, k), \quad (45)$$

$$\begin{cases} \tilde{a}_{jm}^* \sim Ex(\mu^{1/3}) & (c_j \in \tilde{C}_q) \\ \tilde{a}_{jm}^* = 0 & (c_j \notin \tilde{C}_q) \end{cases} \quad (j = 1, \dots, p; \quad q = 1, \dots, \ell), \quad (46)$$

$$\tilde{s}_{mq}^* \sim Ex(\mu^{1/3}) \quad (m = 1, \dots, k; \quad q = 1, \dots, \ell), \quad (47)$$

where $Ex(\cdot)$ denotes an exponential distribution, μ denotes the expected value of each entry of Y , and $\tilde{R} = \{\tilde{R}_1, \dots, \tilde{R}_k\}$ and $\tilde{C} = \{\tilde{C}_1, \dots, \tilde{C}_\ell\}$ denote true partitions of row and column objects. These partitions are determined using discrete uniform random numbers. It is noted that the μ undoubtedly represents the expected value of y_{ij} from the following equation:

$$\begin{aligned} E[y_{ij}] &= E[\tilde{f}_{im}^* \tilde{s}_{mq}^* \tilde{a}_{jq}^*] = E[\tilde{f}_{im}^*] E[\tilde{s}_{mq}^*] E[\tilde{a}_{jq}^*] = (\mu^{1/3})^3 = \mu \\ &\quad (i = 1, \dots, n; \quad j = 1, \dots, p; \quad \tilde{R}_m \ni r_i; \quad \tilde{C}_q \ni c_j). \end{aligned} \quad (48)$$

Here, we get the normalized true factor matrices, $\tilde{F}$, $\tilde{S}$, and $\tilde{A}$ for the later step as follows:

$$\tilde{F} = \tilde{F}^* D_{\tilde{F}^*}^{-1}, \quad \tilde{A} = \tilde{A}^* D_{\tilde{A}^*}^{-1}, \quad \tilde{S} = D_{\tilde{F}^*} \tilde{S}^* D_{\tilde{A}^*}. \quad (49)$$

where $D_{\tilde{F}^*}$ and $D_{\tilde{A}^*}$ denote the $k \times k$ and $\ell \times \ell$ diagonal matrix, in which each diagonal element is $\|\tilde{f}_m^*\|$ ($m = 1, \dots, k$) and $\|\tilde{a}_q^*\|$ ($q = 1, \dots, \ell$), respectively. Second, we execute our proposed method ONMTF-N and the previous methods (Ding's and Yoo's methods) using given Y , initialized factor matrices, τ , and ν , where τ denotes the threshold for the stopping algorithm and ν denotes the maximum number of iterative cycles. Here, we use the initial inputs $\tilde{R}$, $\tilde{C}$, $\tilde{F}$, $\tilde{S}$, and $\tilde{A}$ for ONMTF-N which is generated in the same manner of generating true ones as described above. In this initialization, instead of the μ , $\bar{y} = \sum_{i=1}^n \sum_{j=1}^p y_{ij} / (np)$ is used in generating exponential random numbers. For Ding's and Yoo's methods, we generate initial inputs as follows:

$$\begin{aligned} \tilde{f}_{im}, \tilde{s}_{mq}, \tilde{a}_{jq} &\sim Ex(\bar{y}^{1/3}) \\ (i = 1, \dots, n; j = 1, \dots, p; m = 1, \dots, k; q = 1, \dots, \ell) \end{aligned} \quad (50)$$

From this execution, we get the estimated parameters, that is, estimated partitions of row and column objects $\hat{\mathcal{R}} = \{\hat{R}_1, \dots, \hat{R}_k\}$ and $\hat{\mathcal{C}} = \{\hat{C}_1, \dots, \hat{C}_\ell\}$, respectively, and three factor matrices $\hat{F}$, $\hat{S}$, and $\hat{A}$. Note that the factor matrices are normalized such as (49). Finally, we measure the similarity (or dissimilarity) between each of the five pairs of true and the estimated parameters. We calculate the following five indices: (a) $\text{ARI}(\tilde{\mathcal{R}}, \hat{\mathcal{R}})$, (b) $\text{ARI}(\tilde{\mathcal{C}}, \hat{\mathcal{C}})$, (c) $\|\tilde{F} - \hat{F}\|/(nk)$, (d) $\|\tilde{A} - \hat{A}\|/(p\ell)$, and (e) $\|\tilde{S} - \hat{S}\|/(k\ell)$, where $\text{ARI}(\cdot, \cdot)$ denotes the adjusted Rand index (ARI) (Hubert and Arabie 1985), which is a similarity measure between two partitions of objects. ARI is 1 if the two partitions are completely the same and is close to 0 if they are different. For the three factor matrices, we calculate the mean square error (MSE). The parameters for generating the synthetic data are set as follows: $(n, p, k, \ell) = (1000, 600, 5, 3)$, $\sigma = 1$, $\mu = 10$, $\tau = np \times 10^{-7}$, $\nu = 1000$. For more results of the other conditions, please see Abe H, Yadohisa H (2017) Supplementary material to "Orthogonal nonnegative matrix tri-factorization based on Tweedie distributions." It is noted that the true numbers of row and column clusters, and the estimated ones, are the same as k and ℓ , respectively. We generate 100 synthetic data matrices; from among the candidates of the estimates given by 20 executions of each ONMTF, we select the best estimates for which the objective function value is minimized. The results are shown in Fig. 1. ONMTF-N has the highest ARI for row clusters (see (a)), followed by Yoo's and Ding's methods in that order. In contrast, ARIs for the column clusters (see (b)) of both Ding's and Yoo's methods are as large as those of ONMTF-N. This result suggests that all three methods can accurately detect the smaller side clusters (in our simulation, this is the column side), but two previous methods cannot accurately detect the larger side clusters. The results of the MSE for the row and column factor matrices (see the (c) and (d)) are similar to the ARI results. The MSE for the middle factor matrix (see (e)) obtained by our method is smaller than those obtained by the other

Fig. 1 Boxplots of five indices for the 3 methods. Each of the panels represents **a** $\text{ARI}(\hat{\mathcal{R}}, \hat{\mathcal{R}})$, **b** $\text{ARI}(\hat{\mathcal{C}}, \hat{\mathcal{C}})$, **c** $\|\tilde{\mathbf{F}} - \hat{\mathbf{F}}\|/(nk)$, **d** $\|\tilde{\mathbf{A}} - \hat{\mathbf{A}}\|/(p\ell)$, and **e** $\|\tilde{\mathbf{S}} - \hat{\mathbf{S}}\|/(k\ell)$. The left, middle, and right boxplots represent Ding's, Yoo's, and our proposed method, respectively. The vertical axis ranges are not same because the magnitude of f_{im} and a_{jq} differs depending on n and p

Fig. 2 Graph of $d_\beta(x, y)$ for various β values. The left side represents $y = 10$ and the right side represents $y = 100$. The horizontal axis denotes x and the vertical axis denotes $d_\beta(x, y)$

two methods. Overall, from this simulation study, we can conclude that ONMTF-N provides better estimation consistency than the other two ONMTF methods.

5.2 Simulation study 2: robustness of ONMTF-CP against extremely large positive values

We conducted another simulation study to demonstrate the characteristics of the estimates given by ONMTF-N, ONMTF-P, and ONMTF-CP. As mentioned in previous sections, it is assumed that y_{ij} follows normal, Poisson, and CP distributions, respectively, in these three ONMTF methods. These distributions belong to the Tweedie family, which is described by (20), and the value of β determines the distribution: it is normal if $\beta = 2$, Poisson if $\beta = 1$, and CP if $\beta \in (0, 1)$. The index parameter β is related to the robust estimation against extremely large positive values. Figure 2 shows β -divergence, which is derived from the log-likelihood of Tweedie distributions for various β values when $y = 10$ or $y = 100$. The β -divergence around a small y is larger than that around a large y , except $\beta = 2$. This means that extremely large positive values in the data are not considered for parameter estimation when β is smaller than 2. To examine these characteristics in ONMTF, we measure the estimation accuracy of ONMTF-N, ONMTF-P, and ONMTF-CP for synthetic data matrices generated using normal, Poisson, and CP distributions of data. The accuracy is calculated using the ARI between true clusters and estimated clusters of row and column objects.

We now explain how to generate a synthetic data matrix. First, we generate $\tilde{\mathcal{R}}, \tilde{\mathbf{F}}^*, \tilde{\mathcal{C}}, \tilde{\mathbf{A}}^*$, and $\tilde{\mathbf{S}}^*$ as in Sect. 5.1. Next, we generate each element of the synthetic data matrix $\mathbf{Y}$ as a random number from $y_{ij} \sim TW(x_{ij}, \phi, \tilde{\beta})$, where $TW(x, \phi, \beta)$ denotes a Tweedie distribution, and the mean x_{ij} is the corresponding i, j element of

$X = \tilde{F}^* \tilde{S}^* \tilde{A}^{*/}$. Note that $TW(x_{ij}, \phi, \tilde{\beta})$ is normal if $\tilde{\beta} = 2$ and Poisson if $\tilde{\beta} = 1$. Negative values of y_{ij} can be generated when $\tilde{\beta} = 2$. In this case, y_{ij} is converted to zero. The parameters for generating synthetic data are set as follows: $(n, p, k, \ell) = (100, 100, 5, 5)$, $\phi = 2$, $\tilde{\beta} = \{2, 1, 0.8, 0.5, 0.2\}$, $\mu = 10$, $\tau = np \times 10^{-7}$ (threshold for the stopping algorithm), and $v = 1000$ (the maximum number of iterative cycles.) It is noted that the true numbers of row and column clusters, and the estimated ones, are the same as k and ℓ , respectively. We generate 100 synthetic data matrices for each of five conditions. Then, from among the estimate candidates given by 20 executions, we select the best estimates, $\hat{\mathcal{R}}$, for which the objective function value is minimized. We then calculate $\text{ARI}(\tilde{\mathcal{R}}, \hat{\mathcal{R}})$ of each ONMTF. We execute ONMTF-CP for three cases: $\beta = \{0.2, 0.5, 0.8\}$. We refer to the procedures as ONMTF-CP2, ONMTF-CP5, and ONMTF-CP8, respectively. The results are shown in Fig. 3. Note that we do not show the result of ARI for column clustering because it is very similar to that for row clustering. When $\tilde{\beta} = 2$ (normal), ONMTF-N has the best accuracy, followed by ONMTF-P, ONMTF-CP8, ONMTF-CP5, and ONMTF-CP2, in that order. When $\tilde{\beta} = 0.5$, ONMTF-N has the worst accuracy; when $\tilde{\beta} = 0.2$, the accuracy deteriorates in the order of ONMTF-N, ONMTF-P, and ONMTF-CP8. Because more extremely large positive values are generated from a CP distribution with small $\tilde{\beta}$ values, these results imply that ONMTF-N, ONMTF-P, and ONMTF-CP procedures with relatively larger β values do not fit a data matrix containing some extremely large positive values. This does not mean that an ONMTF-CP procedure with a small β value is the best in any case. It may be worse for a data matrix having a normal error, as shown in the case of $\tilde{\beta} = 2$ in Fig. 3. However, it may be preferable to use ONMTF-CP because the inaccuracy of ONMTF-CP with small β values is smaller than that of ONMTF-N for a data matrix containing some extremely large positive values.

6 Applications

In this section, we describe an application involving a document-term data matrix. The data is used to compare the goodness of clustering of ONMTF-N with those of the previous ONMTF methods and other methods used for the document clustering task. In addition, we attempt to apply our methods to point-of-sale data to check the differences among ONMTF-N, ONMTF-P, and ONMTF-CP. However, the application is not described herein because of space limitations. For the application, please see Abe H, Yadohisa H (2017) Supplementary material to “Orthogonal nonnegative matrix tri-factorization based on Tweedie distributions.”

We describe an application involving a matrix containing document-term data to compare the goodness of clustering of ONMTF-N with previous ONMTF, Ding’s and Yoo’s methods, and those of methods used previously for the document clustering task. Two reasons are considered to apply the three ONMTF methods for document-term data: (1) the ONMTF is compatible with the document-term clustering; (2) these two previous ONMTF methods have some problems, as described above, and we are interested in its performance in real data application. In addition to the three ONMTF methods based on normal distribution, we also applied various methods described in

Fig. 3 $ARI(\hat{\mathcal{R}}, \hat{\mathcal{R}})$ obtained by five ONMTF methods for five conditions. The five boxplots for ONMTF-N, ONMTF-P, ONMTF-CP8, ONMTF-CP5, and ONMTF-CP2 are lined up from left to right in that order in each of the panels

Table 1 List of methods for comparison in the application for document clustering

Method	Abbreviation	Paper
ONMTF by Ding et al. (2006)	Ding	Ding et al. (2006)
ONMTF by Yoo and Choi (2010b)	Yoo	Yoo and Choi (2010b)
Double K-means	DK	Vichi (2001)
LP-FNMTF	LP	Wang et al. (2011)
ONMTF-N	N	Proposed
ONMTF-P	P	Proposed
ONMTF-CP	CP	Proposed
Graph modularity maximization	Mod	Ailem et al. (2016)
SPKM	SP	Banerjee et al. (2003)
WVSPKM	WV	Pompili et al. (2014)

Table 1. Double k -means (DK), LP-FNMTF (LP), and Graph modularity maximization (Mod) were applied as the base line biclustering methods. LP was proposed by Wang et al. (2011) as extended double k -means. Mod was proposed by Ailem et al. (2016) for direct maximization of bipartite graph modularity. We applied ONMTF-P (P) and ONMTF-CP (CP8, CP5, and CP2) to confirm the robust performance of them for the document clustering task. SPKM (SP) and WVSPKM (WV) were also applied for comparison with one-side clustering with biclustering.

The data matrices we used were obtained from the open data CLUTO¹ website. Table 2 lists the selected data matrices and statistics. The list of datasets in Table 2 are ordered by the number of elements. The $tr11$, $tr12$, $tr23$, $tr31$, $tr41$, and $tr45$ datasets are derived from the TREC² collections. The true categories of the documents in the $tr31$ and $tr41$ datasets are obtained by particular queries. The $re0$ and $re1$ datasets are

¹ <http://glaros.dtc.umn.edu/gkhome/views/cluto>.

² <http://trec.nist.gov/>.

Table 2 Stats of some text-word datasets in CLUTO

Data	Documents	Terms	Classes	Elements	Nonzero elements	Ratio of nonzero elements	Total words (%)
tr23	204	5832	6	1189728	78609	6.61	493387
tr12	313	5804	8	1816652	85640	4.71	311111
tr11	414	6429	9	2661606	116613	4.38	437143
re0	1504	2886	13	4340544	77808	1.79	128671
fbis	2463	2000	17	4926000	393386	7.99	1063914
tr45	690	8261	10	5700090	193605	3.40	646537
re1	1657	3758	25	6227006	87328	1.40	142680
tr41	878	7454	10	6544612	171509	2.62	357606
tr31	927	10128	7	9388656	248903	2.65	892795
wap	1560	8460	20	13197600	220482	1.67	337521
k1a	2340	21839	20	51103260	349792	0.68	530374
k1b	2340	21839	6	51103260	349792	0.68	530374
hitech	2301	22498	6	51767898	346881	0.67	549664

from the Reuters-21578 text categorization test collection, distribution 1.0.³ The *fbis* data set is from the Foreign Broadcast Information Service data of TREC-5. The *hitech* is a dataset of San Jose Mercury Newspaper articles and contains documents about computers, electronics, health, medicine, research, and technology. The *k1a*, *k1b*, and *wap* datasets are used for the WebACE project (Boley et al. 1999) and contain web pages in various subject directories of Yahoo!.⁴ Datasets *k1a* and *k1b* contain the same documents, but the true labels are different.

We conduct tf-idf conversion for all data matrices. The intuitive idea for the document clustering is to classify the documents using “unique words” that appear in each of the documents. Although we count the appearance frequency of every word in each of the documents, the counting does not represent the “uniqueness” of the word. Well used words (e.g., “run”, “come”, and “get”) are not important for document clustering. We can add weight to words, which do not appear in many documents, by tf-idf conversion. Using tf-idf values provides a well intuitive document clustering. The row and column objects are assigned to documents and words, respectively; then, F and A can be considered as the factor matrices of documents and words, respectively. The number of document clusters (the number of columns of F , e.g., k) is set to the number of document classes provided in advance, and the number of word clusters (the number of columns of A , e.g., ℓ) is set to 10 for all the data matrices. Although the best number of word cluster must be estimated from the dataset, we set the number to 10 in advance to avoid the time consuming computation. Note that the $k=\ell$ in Mod algorithm, that is, the numbers of word and document clusters are same. The convergence threshold is set as $\tau = 10^{-5} \|Y\|^2/2$ for all the methods except for co-clustering based on graph modularity whose computation is repeated until no change is observed in its objective function value. The goodness of clustering is measured using the ARI between the given clusters and estimated clusters of the documents. It is noted that some clusters occasionally become empty in the update iteration for some methods. In this case, we restart the update iteration from another initial parameter. From among the candidates of the estimates given by 10 executions of each method, we select the best estimates for which the objective function value is minimized. The initial inputs of each method is as follows: for Ding, Yoo, N, P, and CP methods, we use initial inputs as described in Sect. 5.1. For DK, we generate initial partition of row objects as described in Sect. 5.1 and initial middle factor matrix (which means center of cluster) such as $s_{mq} \sim Ex(\bar{y})$. For SP, we generate the initial right side factor matrix (which means the normalized center of cluster) as follows: first we generate random number $a_{jk} \sim Ex(1)$ and then normalize square norm of each row of the A to 1. For WV, we generate the initial right side factor matrix such as $a_{jk} \sim Ex(\bar{y}^{1/2})$. For LP, we generate initial partition of row and column objects as described in Sect. 5.1. For Mod, we use the default initialization procedure of the “CoclustMod” function which is distributed in Python package “coclust.”⁵ We apply the LP in the same way described in Wang et al. (2011). We use nearest neighbor graph as affinity matrices. We try to compute with all patterns of hyperparameters: the neighborhood size $k = \{1, 2, \dots, 10\}$ and each of the regu-

³ <http://www.daviddlewis.com/resources/testcollections/reuters21578/>.

⁴ <http://www.yahoo.com/>.

⁵ <http://coclust.readthedocs.io/en/v0.2.1/>.

Table 3 ARI between the given and estimated clusters of the documents

data	Ding	Yoo	DK	LP	N	P	CP8	CP5	CP2	Mod	SP	WV
tr23	0.02	<i>0.24</i>	0.22	0.09	0.21	0.14	0.06	0.02	0.02	0.13	0.28	*0.22
tr12	<i>0.59</i>	0.60	0.18	0.13	0.57	0.12	0.04	0.01	0.01	0.12	0.36	*0.58
tr11	*0.60	<i>0.63</i>	0.24	0.29	0.69	0.13	0.01	0.01	0.00	0.26	0.52	0.59
re0	*0.18	0.21	0.08	0.12	0.16	0.08	0.06	0.09	0.06	0.10	<i>0.18</i>	0.17
fbis	0.31	0.29	0.24	0.20	0.32	0.36	*0.36	0.32	0.17	0.48	<i>0.36</i>	0.33
tr45	0.42	0.32	0.22	0.10	0.64	0.44	0.32	0.03	0.02	0.38	<i>0.62</i>	*0.54
re1	0.12	0.06	0.10	*0.12	0.11	0.10	0.06	0.04	0.02	0.10	<i>0.25</i>	0.42
tr41	0.33	0.21	0.32	0.12	<i>0.43</i>	0.40	0.30	0.01	0.01	0.42	0.49	*0.43
tr31	*0.41	0.10	0.22	0.04	<i>0.55</i>	0.30	0.26	0.02	0.01	0.40	0.38	0.60
wap	*0.30	0.26	0.25	0.11	0.27	0.26	0.24	0.13	0.02	0.45	<i>0.43</i>	0.29
k1a	0.33	0.44	0.29	0.14	0.21	0.25	0.16	0.01	0.01	<i>0.39</i>	0.32	*0.34
k1b	0.42	0.69	0.33	0.06	<i>0.50</i>	0.17	0.00	0.00	0.00	0.19	0.45	*0.50
hitech	<i>0.20</i>	0.00	0.07	0.01	*0.12	0.08	0.00	0.00	0.00	0.11	0.26	0.12

The first, second, and third values are denoted by bold case, italics, and star “*”, respectively

larization parameters, α and β (which is different from β of Tweedie distribution or β -divergence), is $\{0.1, 1, 10, 100, 500, 1000\}$. We select the hyperparameter pattern with the highest average of ARI for relatively small datasets (*tr23*, *tr12*, *tr11*, *re0*, *fbis*, and *tr45*) and then we apply for the remaining datasets with the hyperparameter pattern. The selected pattern is $k = 9$, $\alpha = 100$, and $\beta = 500$. The result is in Table 3.

Table 3 shows ARI of the 12 methods for each dataset. Yoo’s method has four datasets (*tr12*, *re0*, *k1a*, and *k1b*) with the best ARI and this number is the largest amongst those of the 12 methods. The second best performance is shown in SP, which has three datasets (*tr23*, *tr41*, and *hitech*) with the best ARI and the method has high accuracy in many datasets. The third best performance is shown in N, Mod, and WV, each of which show the best ARI in two datasets. Although Yoo’s method seems to show the best performance from the perspective of which method shows the best ARI in any of the datasets, the method has poor ARI on some datasets, e.g., *re1*, *tr41*, *tr31*, and *hitech*. In fact, Yoo’s method shows higher performance than N in no more than 5 datasets of all 13 datasets. On the other hand, N is not poor for almost all datasets in the biclustering methods. However, it is poorer than one-side clustering methods, SP and WV. A possible reason for the result is as follows. Applying one-side clustering approach to document-term clustering task leads to the clustering of documents and terms into the same number of clusters; on the other hand, biclustering approach leads to clustering of the two into a different number of clusters. In the biclustering approach, a number of term clusters may be short or too much for clustering documents, and this mismatch could cause the poorer performance of N in comparison to one-side clustering methods. We also show the worse performance of P and CPs, which are robust methods, in comparison to that of other methods. The cause of this result can be robustness against the large positive values. We use a document-term matrix converted using tf-idf conversion, which strongly weights terms in a few documents.

The entries of such terms have a large positive value. The effect of the weighted entries disappear when these robust ONMTFs are used, implying that an ONMTF based on the Euclidean distance is not used, and interpretable clusters cannot be obtained. Note that good performance of the robust ONMTF can be obtained using new standardizations other than tf-idf conversion. However, we will undertake this challenge in our future works. DK and LP have the poor result in most of the datasets. The accuracy of LP can be improved by selection more appropriate hyperparameters for each dataset.

We obtain the results of the computational time, degree of approximation (measured as Euclidean distance at convergence), and convergence behavior. However, we do not report these herein because of space limitation. Please see Abe H, Yadohisa H (2017) Supplementary material to “Orthogonal nonnegative matrix tri-factorization based on Tweedie distributions for the results.”

We now show the estimates given by ONMTF-N using the *kla* datasets as an example to demonstrate the features from the bi-clustering aspect. The *kla* dataset consists of web news documents obtained from the Reuters news service in October, 1997 (Boley 1998). The documents in *kla* are labeled in advance under six categories: “business,” “entertainment,” “health,” “politics,” “sports,” and “tech”. Table 4 presents the estimated 10 clusters of the terms given by ONMTF-N. For comparison, we also show them by Yoo’s method in Table 5. Table 6 shows the center factor matrices S that enable us to grasp the relationship between the estimated document and the term clusters. We label the document clusters herein as DC 1 to DC 6 and the term clusters as TC 1 to TC 10.

We now focus on the term clustering. Each of the term cluster obtained using the two methods could be interpreted as Table 7. We find that both methods extract similar term clusters from each other (e.g., TC 1, 2, 4, 5, and 6 of both.) However, TC 3, which is strongly related to DC 3 (see Table 6) in both methods, has different terms. TC 3 of ONMTF-N seems to mean the chart of movie by the box office and that of Yoo’s method seems to mean sports. Unfortunately, the detail of document clusters cannot be shown due to space limitations, but in fact, DC 3 of ONMTF-N includes some “entertainment” documents, while that of Yoo’s method includes almost all of “sports” documents. From the fact that Yoo’s TC 1 includes some words related to box office, documents related to box office chart can be integrated into DC 1 in Yoo’s estimates. On the other hand, there are no word cluster on sports in ONMTF-N. The “sports” documents are indeed misclassified to DC 1, which includes many entertainment documents, by ONMTF-N. Although this misclassifying of “sports” documents leads to poorer ARI of ONMTF-N than Yoo’s method, ONMTF-N detects the meaningful cluster related to box office chart instead of “sports.” However, ONMTF-N has irrelevantly estimated the document clusters unfamiliar with the real document labels. Moreover, ONMTF-N has some assorted term clusters that are not affected by document clustering. The difference of the two can be explained from the aspect of their estimating algorithms. In fact, ONMTF-N exhibits less freedom than Yoo’s method for estimating factor matrices owing to their orthogonality. Indeed its objective function value at convergence is larger than Yoo’s: ONMTF-N has 241.8; and Yoo’s method has 241.1. This estimating problem can be solved by more random starts thanks to the higher computational speed of ONMTF-N in comparison to Yoo’s method.

Table 4 Terms of 10 clusters in the k/b datasets obtained using ONMTF-N

TC 1	a_{j1}	TC 2	a_{j2}	TC 3	a_{j3}	TC 4	a_{j4}	TC 5	a_{j5}
Film	0.26	Cell	0.25	Box	0.43	Stock	0.29	Week	0.48
Tv	0.17	Cancer	0.24	Million	0.33	Internet	0.26	Bestsell	0.45
Hollywood	0.15	Risk	0.24	Weekend	0.29	Compani	0.21	Weekli	0.35
cb	0.14	Studi	0.22	Offic	0.27	Dow	0.21	Hardcov	0.32
Star	0.14	Research	0.20	Movi	0.24	Microsoft	0.17	Publish	0.28
Diana	0.12	Patient	0.19	Gross	0.24	Percent	0.16	Paperback	0.25
Fox	0.12	Women	0.18	Sept	0.23	Comput	0.16	Fiction	0.14
Game	0.12	Diseas	0.17	Top	0.22	Busi	0.14	Mass	0.11
Festiv	0.12	Heart	0.16	Chart	0.21	Market	0.13	Random	0.10
Season	0.11	Drug	0.14	Exhibitor	0.17	Intel	0.13	Trade	0.08
TC 6	a_{j6}	TC 7	a_{j7}	TC 8	a_{j8}	TC 9	a_{j9}	TC 10	a_{j10}
Emmi	0.74	Report	0.21	Rate	0.27	Deal	0.20	York	0.21
Win	0.31	Accord	0.20	Adult	0.20	Network	0.19	Unit	0.18
Drama	0.25	People	0.17	Includ	0.16	Quote	0.16	Averag	0.17
Comedi	0.21	American	0.16	Time	0.16	Am	0.16	Call	0.17
Actor	0.16	Univers	0.16	Previou	0.14	Set	0.16	Loss	0.17
Award	0.16	Develop	0.15	Home	0.14	Octob	0.16	Offer	0.16
Franz	0.15	Death	0.15	Septemb	0.14	Wednesdai	0.16	System	0.16
Actress	0.14	Author	0.15	Program	0.14	Mondai	0.15	Pm	0.16
Sundai	0.14	Lead	0.14	Nation	0.14	Record	0.15	Gain	0.15
Gillian	0.13	Surgeri	0.13	Fridai	0.13	Tuesdai	0.14	Provid	0.14

The value on the right side of each term is the nonzero value of a_{jq} in each term. All a_{jq} are standardized such that the length of each column vector in A is 1. Only the terms that have top 10 values of nonzero a_{jq} in each cluster are shown

Table 5 Terms of 10 clusters in the k/b datasets by Yoo's method. The value of the right side of each of the term is the value of a_{jq} on each of the term. All a_{jq} are standardized such that the length of each column vector in A is 1. Only terms are shown which have top 10 values of a_{jq} in each cluster

TC 1	a_{j1}	TC 2	a_{j2}	TC 3	a_{j3}	TC 4	a_{j4}	TC 5	a_{j5}
Film	0.24	Risk	0.30	Game	0.33	Stock	0.35	Week	0.49
Tv	0.22	Patient	0.24	Season	0.17	Compani	0.24	Bestsell	0.44
Box	0.17	Studi	0.21	Marlin	0.16	Internet	0.19	Weekli	0.35
Top	0.16	Heart	0.20	Pippen	0.16	Microsoft	0.17	Hardcov	0.32
Star	0.14	Drug	0.20	Blackhawk	0.16	Percent	0.17	Publish	0.27
Festiv	0.12	Women	0.19	Surgeri	0.15	Busi	0.16	Paperback	0.25
Weekend	0.12	Infect	0.17	Indian	0.15	Industri	0.15	Fiction	0.13
Music	0.12	Blood	0.17	Oriol	0.15	Financi	0.14	Mass	0.11
Diana	0.12	Breast	0.16	Coach	0.14	Oet	0.13	Random	0.09
Pictur	0.11	Increas	0.14	Nomo	0.14	Trad	0.13	Trade	0.08
TC 6	a_{j6}	TC 7	a_{j7}	TC 8	a_{j8}	TC 9	a_{j9}	TC 10	a_{j10}
Emmi	0.74	Million	0.28	Dow	0.38	Cancer	0.41	Home	0.14
Win	0.32	Cb	0.27	Internet	0.23	Cell	0.30	Includ	0.14
Drama	0.25	Hollywood	0.23	Quarter	0.22	Research	0.29	People	0.14
Comedi	0.21	Fox	0.19	Intel	0.19	Gene	0.21	York	0.12
Actor	0.16	Debut	0.19	Softwar	0.19	Brain	0.20	Call	0.12
Franz	0.15	Premier	0.19	Chip	0.18	Diseas	0.19	Program	0.12
Award	0.15	Film	0.15	Apple	0.17	Mutat	0.14	Clinton	0.12
Actress	0.14	Deal	0.13	Quote	0.17	Studi	0.14	Lead	0.11
Sundai	0.13	Award	0.13	Oper	0.16	Tumor	0.10	Receiv	0.11
Gillian	0.13	Ticket	0.13	Technologi	0.16	Test	0.10	Previo	0.11

Table 6 Middle factor matrix S of the *kla* dataset

Term	ONMTF-N						Yoo's method					
	DC1	DC2	DC3	DC4	DC5	DC6	DC1	DC2	DC3	DC4	DC5	DC6
TC 1	1.76	0.08	0.17	0.12	0.05	0.11	1.53	0.01	0.04	0.05	0.01	0.00
TC 2	0.09	1.95	0.01	0.05	0.01	0.00	0.00	1.27	0.00	0.00	0.01	0.00
TC 3	0.28	0.04	1.57	0.11	0.04	0.03	0.03	0.00	1.59	0.02	0.00	0.00
TC 4	0.22	0.06	0.04	1.61	0.02	0.01	0.14	0.04	0.03	1.35	0.00	0.00
TC 5	0.16	0.06	0.01	0.05	1.75	0.02	0.02	0.01	0.00	0.01	1.80	0.00
TC 6	0.27	0.01	0.03	0.02	0.00	1.99	0.04	0.00	0.00	0.00	0.00	2.02
TC 7	0.36	0.81	0.04	0.19	0.02	0.03	0.93	0.00	0.00	0.00	0.00	0.03
TC 8	0.77	0.39	0.06	0.17	0.04	0.08	0.00	0.00	0.00	0.69	0.00	0.00
TC 9	0.85	0.11	0.06	0.58	0.02	0.06	0.01	1.17	0.00	0.01	0.00	0.00
TC 10	0.38	0.36	0.06	0.44	0.03	0.02	0.55	0.45	0.15	0.15	0.00	0.00

Table 7 Our interpretation for term clusters obtained by two methods

Cluster	ONMTF-N	Yoo's method
TC 1	Cinema and television	Cinema, television, and box office
TC 2	Clinical	Clinical
TC 3	Box office	Sports
TC 4	Economics and technology	Economics and technology
TC 5	Book sales	Book sales
TC 6	Emmy awards	Emmy awards
TC 7	Words used for a research reference	Cinema and its profit
TC 8	(Assorted terms)	Internet technology
TC 9	(Assorted terms)	Biology
TC 10	(Assorted terms)	(Assorted terms)

7 Conclusion

In this paper, we proposed a new method for ONMTF, namely, ONMTF-N, in which the objective function value is monotonically non-increasing and the orthogonality of the factor matrices is maintained. In addition, we proposed two other ONMTF methods, namely, ONMTF-P and ONMTF-CP. The main contributions of this paper are as follows. First, our simulation study and an application involving some document-term matrices indicated that ONMTF-N shows higher estimation accuracy than previous methods. Second, we derived a new auxiliary function to optimize the middle factor matrix in ONMTF-CP using an inequality of a bivariate concave function. Third, another simulation study indicated that ONMTF-CP may be robust against the effect of extremely large positive values.

NMFs with orthogonality, including our methods, should be used considering the trade-off between its easy-to-understand estimates and its under fitting. An orthogonal

constraint simplifies a factor matrix, thereby facilitating result interpretation. However, a factor matrix with a simplified structure leads to a poor approximation to data matrix (e.g., Y .)

Two issues remain to be addressed in the future. First, cluster degeneration tends to occur in the three proposed methods. Therefore, further investigation is required to develop an approach for rapidly seeking alternative initial parameters, in order to avoid cluster degeneration. Second, it is necessary to develop an approach for estimating the best number of clusters for both row objects and column objects.

Acknowledgements We would like to express our greatest appreciation to the editor and referees for their insightful comments, which have helped us significantly improve the paper.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Ailem M, Role F, Nadif M (2016) Graph modularity maximization as an effective method for co-clustering text data. *Knowl Based Syst* 109:160–173
- Banerjee A, Dhillon I, Ghosh J, Sra S (2003) Generative model-based clustering of directional data. In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp 19–28
- Berry MW, Browne M, Langville AN, Pauca VP, Plemmons RJ (2007) Algorithms and applications for approximate nonnegative matrix factorization. *Comput Stat Data Anal* 52(1):155–173
- Boley D (1998) Hierarchical taxonomies using divisive partitioning. Technical Report TR-98-012, Department of Computer Science, University of Minnesota, Minneapolis
- Boley D, Gini M, Gross R, Han EHS, Hastings K, Karypis G, Kumar V, Mobasher B, Moore J (1999) Document categorization and query generation on the world wide web using webace. *Artif Intell Rev* 13(5–6):365–391
- Carabias-Orti JJ, Rodríguez-Serrano FJ, Vera-Candeas P, Cañadas-Quesada FJ, Ruiz-Reyes N (2013) Constrained non-negative sparse coding using learnt instrument templates for realtime music transcription. *Eng Appl Artif Intell* 26(7):1671–1680
- Chen G, Wang F, Zhang C (2009) Collaborative filtering using orthogonal nonnegative matrix tri-factorization. *Inf Process Manag* 45(3):368–379
- Choi S (2008) Algorithms for orthogonal nonnegative matrix factorization. In: *Neural Networks, 2008. IJCNN 2008. IEEE international joint conference on IEEE world congress on computational intelligence*, IEEE, pp 1828–1832
- Cichocki A, Amari Si (2010) Families of alpha-beta-and gamma-divergences: flexible and robust measures of similarities. *Entropy* 12(6):1532–1568
- Costa G, Ortale R (2014) XML document co-clustering via non-negative matrix tri-factorization. In: *2014 IEEE 26th international conference on tools with artificial intelligence (ICTAI)*, IEEE, pp 607–614
- Ding C, Li T, Peng W, Park H (2006) Orthogonal nonnegative matrix tri-factorizations for clustering. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp 126–135
- Dunn PK, Smyth GK (2001) Tweedie family densities: methods of evaluation. In: *Proceedings of the 16th international workshop on statistical modelling*, Odense, Denmark, pp 2–6
- Févotte C, Idier J (2011) Algorithms for nonnegative matrix factorization with the β -divergence. *Neural Comput* 23(9):2421–2456
- Févotte C, Bertin N, Durrieu JL (2009) Nonnegative matrix factorization with the Itakura–Saito divergence: with application to music analysis. *Neural Comput* 21(3):793–830
- Govaert G, Nadif M (2013) *Co-clustering*. Wiley, Hoboken

- Hubert L, Arabie P (1985) Comparing partitions. *J Classif* 2(1):193–218
- Jørgensen B (1997) The theory of dispersion models. CRC Press, Boca Raton
- Kim Y, Kim TK, Kim Y, Yoo J, You S, Lee I, Carlson G, Hood L, Choi S, Hwang D (2011) Principal network analysis: identification of subnetworks representing major dynamics using gene expression data. *Bioinformatics* 27(3):391–398
- Kimura K, Tanaka Y, Kudo M (2014) A fast hierarchical alternating least squares algorithm for orthogonal nonnegative matrix factorization. In: *ACML*
- Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. *Nature* 401(6755):788–791
- Lee DD, Seung HS (2001) Algorithms for non-negative matrix factorization. In: *Advances in neural information processing systems*, pp 556–562
- Li T, Peng W (2005) A clustering model based on matrix approximation with applications to cluster system log files. In: *European conference on machine learning*, Springer, pp 625–632
- Li Y, Zhang X, Sun M (2017) Robust non-negative matrix factorization with β -divergence for speech separation. *ETRI J* 39(1):21–29
- Li Z, Wu X, Peng H (2010) Nonnegative matrix factorization on orthogonal subspace. *Pattern Recognit Lett* 31(9):905–911
- Mauthner T, Kluckner S, Roth PM, Bischof H (2010) Efficient object detection using orthogonal NMF descriptor hierarchies. In: Goesele M, Roth S, Kuijper A, Schiele B, Schindler K (eds) *Pattern recognition*. Springer, pp 212–221
- Mirzal A (2014) A convergent algorithm for orthogonal nonnegative matrix factorization. *J Comput Appl Math* 260:149–166
- Nakano M, Kameoka H, Le Roux J, Kitano Y, Ono N, Sagayama S (2010) Convergence-guaranteed multiplicative algorithms for nonnegative matrix factorization with β -divergence. In: *2010 IEEE international workshop on machine learning for signal processing (MLSP)*, IEEE, pp 283–288
- Ohnishi T, Dunn PK (2007) Analysis of the rainfall data in queensland using the tweedie glm. In: *Proceedings of the 2007 Japanese joint statistical meeting, Japanese joint statistical meeting*, pp 18–18
- Pompili F, Gillis N, Absil PA, Glineur F (2014) Two algorithms for orthogonal nonnegative matrix factorization with application to clustering. *Neurocomputing* 141:15–25
- Simsekli U, Cemgil A, Yilmaz YK (2013) Learning the beta-divergence in Tweedie compound poisson matrix factorization models. In: *Proceedings of the 30th international conference on machine learning (ICM-13)*, pp 1409–1417
- Smyth GK, Jørgensen B (2002) Fitting tweedie's compound poisson model to insurance claims data: dispersion modelling. *Astin Bull* 32(01):143–157
- Tan VY, Févotte C (2013) Automatic relevance determination in nonnegative matrix factorization with the/spl beta-divergence. *IEEE Trans Pattern Anal Mach Intell* 35(7):1592–1605
- Van Mechelen I, Bock HH, De Boeck P (2004) Two-mode clustering methods: a structured overview. *Stat Methods Med Res* 13(5):363–394
- Vichi M (2001) Double k-means clustering for simultaneous classification of objects and variables. In: Borra S, Rocci R, Vichi M, Schader M (eds) *Advances in classification and data analysis*. Springer, pp 43–52
- Virtanen T (2007) Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Trans Audio Speech Lang Process* 15(3):1066–1074
- Virtanen T, Gemmeke JF, Raj B, Smaragdis P (2015) Compositional models for audio processing: uncovering the structure of sound mixtures. *IEEE Signal Process Mag* 32(2):125–144
- Wang F, Zhu H, Tan S, Shi H (2016) Orthogonal nonnegative matrix factorization based local hidden Markov model for multimode process monitoring. *Chin J Chem Eng* 24:856–860
- Wang H, Nie F, Huang H, Makedon F (2011) Fast nonnegative matrix tri-factorization for large-scale data co-clustering. In: *IJCAI proceedings-international joint conference on artificial intelligence*, vol 22, p 1553
- Wang YX, Zhang YJ (2013) Nonnegative matrix factorization: a comprehensive review. *IEEE Trans on Knowl Data Eng* 25(6):1336–1353
- Weninger F, Schuller B (2012) Optimization and parallelization of monaural source separation algorithms in the openBLISSART toolkit. *J Signal Process Syst* 69(3):267–277
- Xue Y, Tong CS, Chen Y, Chen WS (2008) Clustering-based initialization for non-negative matrix factorization. *Appl Math Comput* 205(2):525–536

- Yoo J, Choi S (2008) Orthogonal nonnegative matrix factorization: Multiplicative updates on Stiefel manifolds. In: Intelligent data engineering and automated learning–IDEAL 2008, Springer, pp 140–147
- Yoo J, Choi S (2009) Probabilistic matrix tri-factorization. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing (ICASSP), IEEE, pp 1553–1556
- Yoo J, Choi S (2010) Nonnegative matrix factorization with orthogonality constraints. *J Comput Sci Eng* 4(2):97–109
- Yoo J, Choi S (2010) Orthogonal nonnegative matrix tri-factorization for co-clustering: multiplicative updates on Stiefel manifolds. *Inf Process Manag* 46(5):559–570