論文 / 著書情報
Article / Book Information

題目(和文)	非負値行列分解の統計的学習理論と複合データ分析
Title(English)	Statistical Learning Theory of Nonnegative Matrix Factorization and Multiple Data Analysis
著者(和文)	幸島匡宏
Author(English)	Masahiro Kohjima
出典(和文)	学位:博士(理学), 学位授与機関:東京工業大学, 報告番号:甲第11064号, 授与年月日:2019年3月26日, 学位の種別:課程博士, 審査員:渡邊 澄夫,樺島 祥介,金森 敬文,山下 真,中野 張
Citation(English)	Degree:Doctor (Science), Conferring organization: Tokyo Institute of Technology, Report number:甲第11064号, Conferred date:2019/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

STATISTICAL LEARNING THEORY OF NONNEGATIVE MATRIX FACTORIZATION AND MULTIPLE DATA ANALYSIS

MASAHIRO KOHJIMA

SCHOOL OF COMPUTING
DEPARTMENT OF MATHEMATICAL AND COMPUTING SCIENCE
TOKYO INSTITUTE OF TECHNOLOGY

2018

Abstract

A method of decomposing a nonnegative matrix into a product of low rank nonnegative matrices, nonnegative matrix factorization (NMF), is widely applied to pattern recognition and data analysis. However, mathematical theory of statistical learning and application to multiple data have not been established. In this thesis, we clarify the accuracy of statistical inference of NMF using variational Bayes method and construct a method applicable to multiple data with different granularity. The effectiveness will be clarified by numerical experiment.

Contents

Chapter 1

Introduction

Due to the wide-spread of e-commerce and social networking service and the development of sensor devices, various types of a large amount of data about people and things have been collected and accumulated. Organizations and companies analyze such data in order to, for example, improve their services and customer satisfaction. Data analysis is now widely recognized as an important technology directly linked to the company's strengths [Davenport and Harris, 2007, Davenport et al., 2010]. It is also estimated that a lot of work is automated by data analysis related technologies (AI, robots, etc.), and nearly half of the work force will be substituted [Frey and Osborne, 2017, Nomura Research Institute, 2015]. The importance of data analysis will further increase in the future.

It is known that most of data to be analyzed in recent data analysis can be expressed as a matrix with nonnegative elements, i.e., nonnegative matrix. For example, a set of documents is represented by a matrix representing the number of occurrences of words in each document, where rows and columns correspond to documents and words, respectively. See Figure 1.1. Similarly, purchase logs are expressed as a matrix representing the number of purchases of items by users, where the rows and columns correspond to users and items.

Nonnegative Matrix Factorization (NMF) [Lee and Seung, 1999, Lee and Seung, 2001] is a method that can automatically extract a latent pattern underlying data which are represented by nonnegative matrix and that can complete missing values. More specifically, by applying NMF, the input matrix is decomposed into a product of low rank nonnegative matrices as shown in Fig. 1.2. By using the obtained decomposition result, patterns in data are extracted and missing values are complemented. By applying it to purchase logs, for instance, it extracts potential purchasing patterns such as chocolate and coffee lover. It is also possible to understand which purchasing pattern each user follows.

NMF are applied not only to one input matrix but also to multiple matrices, fox example, each matrix represents the purchase logs of each month, as shown in Fig. 1.3. This setting can be seen as the setting of statistical inference where data which consist a set of samples are available. By applying NMF to such multiple

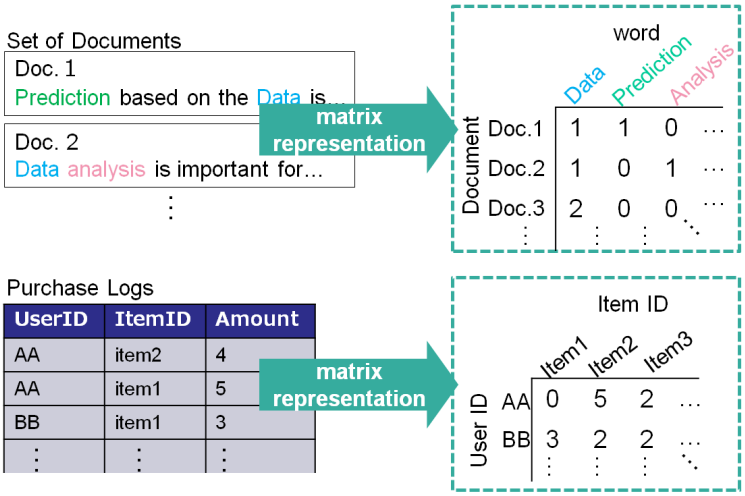


Figure 1.1: Matrix representation of data.

matrices, we could extract a common pattern, e.g., which doesn't depends on the month when data are collected.

In this thesis, we report on two studies on NMF. The first study is to propose an extended NMF that can handle multiple matrices whose granularity of the rows or columns are different [Kohjima et al., 2015, Kohjima et al., 2017]. Due to e.g., the difficulty of comprehensive data collection and protection of personal information, it is required to analyze the data with different granularity, for example, user individual's data representing such as a visit count by user and user group's data representing such as purchase count by gender/age. Figure 1.4 shows a example of matrices whose granularity of the rows are different. Since standard NMF shown in Fig. 1.3 cannot be applied in this setting, it is necessary to construct new method and algorithm.

The second study is to provide a theoretical analysis of variational Bayesian NMF (VBNMF) [Kohjima and Watanabe, 2017], a representative algorithm for NMF. The factorization result output by the VBNMF is determined by the contribution of the hyperparameter to the variational free energy (VFE), which is the objective function of VBNMF. However, theoretical property of VFE has not been clarified. This study investigates the property by asymptotic analysis and clarifies phase transition diagram (Fig. 1.5), which describes the relation between hyperparameters and factorization result.

The rest of this thesis is organized as follows:

- Chapter 2 introduces the model and algorithms of NMF.
- Chapter 3 is devoted to the proposed NMF models for analyzing multiple dataset with inconsistent granularity.
- Chapter 4 provides our theoretical result of NMF.

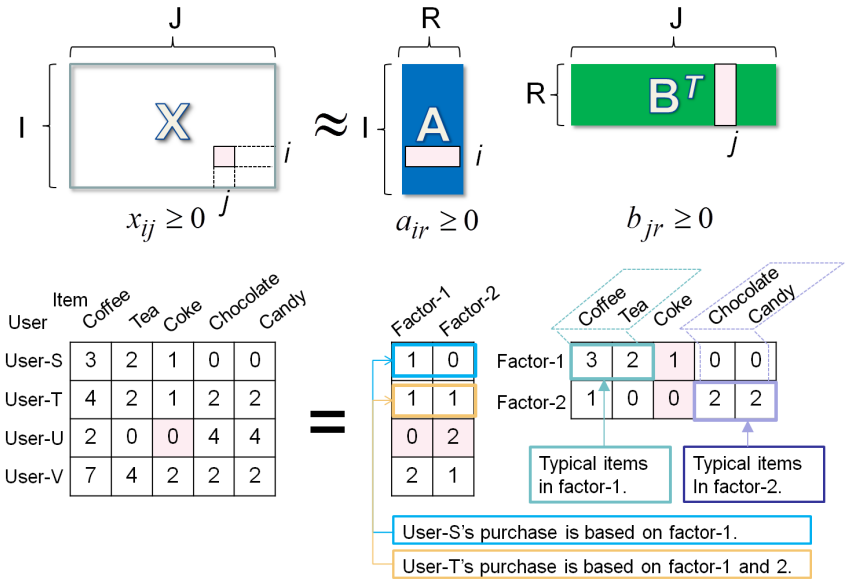


Figure 1.2: Nonnegative matrix factorization.

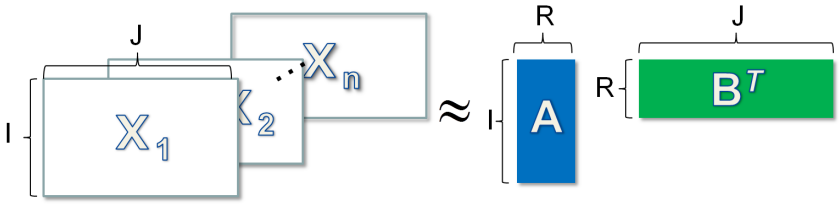


Figure 1.3: NMF handling multiple input matrices.

- Chapter 5 summarizes the thesis.

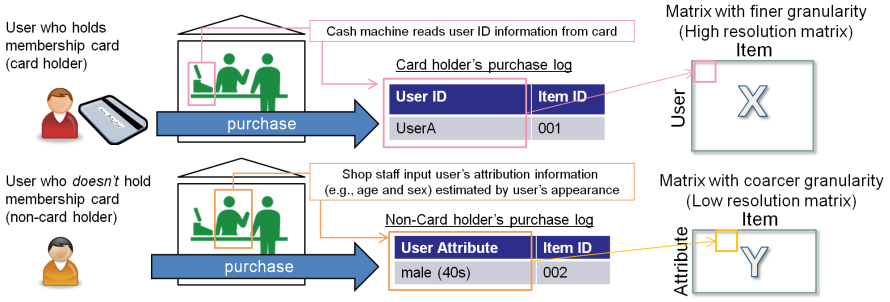


Figure 1.4: Matrices with different granularities.

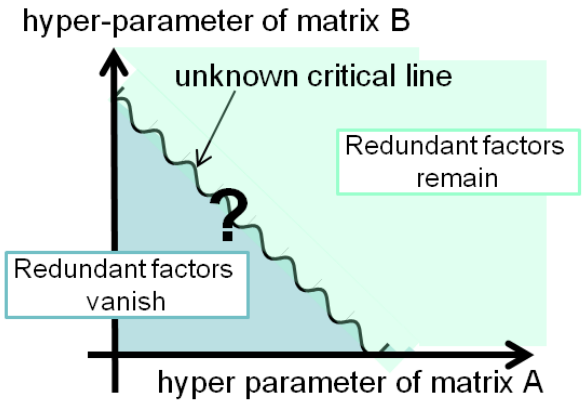


Figure 1.5: (unknown) phase transition diagram.

Chapter 2

Statistical Learning of NMF

In this section, we show the statistical model of nonnegative matrix factorization and the algorithms.

2.1 Nonnegative Matrix Factorization Model

Let $\mathbb{Z}_+^{I\times J}$ and $\mathbb{R}_+^{I\times J}$ be the sets of $I\times J$ matrices whose elements are all nonnegative integers and all nonnegrative real values, respectively. We study a statistical model of NMF which is represented by a probability distribution of $\boldsymbol{X}=(x_{ij})\in\mathbb{Z}_+^{I\times J}$ for a given set of $\boldsymbol{A}\in\mathbb{R}_+^{I\times R}$ and $\boldsymbol{B}\in\mathbb{R}_+^{J\times R}$,

$$P(\boldsymbol{X}|\boldsymbol{A},\boldsymbol{B})=\prod\nolimits_{i,j=1}^{I,J}\mathcal{PO}\big(x_{ij}\Big|\sum\nolimits_{r=1}^{R}a_{ir}b_{jr}\big), \tag{2.1}$$

where a_{ir} and b_{jr} represent the (i,r)-th element of $\boldsymbol{A}$ and the (j,r)-th elements of $\boldsymbol{B}$, respectively. $\mathcal{PO}(z|c)$ is the Poisson distribution of $z\geq 0$ for $c\geq 0$,

$$\mathcal{PO}(z|c)=\frac{c^z\exp(-c)}{z!}.$$

Note that, if independent random variables Z_1 and Z_2 are subject to $\mathcal{PO}(z_1|c_1)$ and $\mathcal{PO}(z_2|c_2)$ respectively, then Z_1+Z_2 is subject to $\mathcal{PO}(z|c_1+z_2)$.

Let $\boldsymbol{X}_1,\boldsymbol{X}_2,\cdots,\boldsymbol{X}_n\in\mathbb{Z}_+^{I\times J}$ be independent random variables and $\boldsymbol{X}^n$ be the set of them, where n is the number of training data. The (i,j) th element of the m th matrix $\boldsymbol{X}_m$ is denoted by x_{ij}^m. The likelihood of the NMF is defined by

$$P(\boldsymbol{X}^n|\boldsymbol{A},\boldsymbol{B})=\prod\nolimits_{m=1}^{n}\prod\nolimits_{i,j=1}^{I,J}\mathcal{PO}\big(x_{ij}^m\Big|\sum\nolimits_{r=1}^{R}a_{ir}b_{jr}\big).$$

We introduce a hidden variable $\boldsymbol{S}^n$ whose element $s_{ijr}^m\in\mathbb{Z}_+^{I\times J}$ represents the contribution of the r-th factor to x_{ij}^m. The joint distribution is given by

$$P(\boldsymbol{X}^n,\boldsymbol{S}^n|\boldsymbol{A},\boldsymbol{B})=\prod\nolimits_{m=1}^{n}\prod\nolimits_{i,j=1}^{I,J}\delta\big(x_{ij}^m-s_{ij\cdot}^m\big)\prod\nolimits_{r=1}^{R}\mathcal{PO}\big(s_{ijr}^m|a_{ir}b_{jr}\big),$$

where $\delta(x) = 1$ if $x = 0$, or $\delta(x) = 0$ otherwise. Note that a dot index means the corresponding one is summed out:

$$s_{\cdot jr} = \sum_{i=1}^{I} s_{ijr}, \quad s_{i\cdot r} = \sum_{j=1}^{J} s_{ijr}, \quad s_{ij\cdot} = \sum_{r=1}^{R} s_{ijr}.$$

It follows that

$$P(\boldsymbol{X}^n|\boldsymbol{A},\boldsymbol{B}) = \sum_{\boldsymbol{S}^n} P(\boldsymbol{X}^n,\boldsymbol{S}^n|\boldsymbol{A},\boldsymbol{B}).$$

For variational Bayesian estimation, we employ the conjugate gamma priors[1] on $\boldsymbol{A}$ and $\boldsymbol{B}$.

$$P(\boldsymbol{A}) = \prod_{i,r=1}^{I,R} \mathcal{G}(a_{ir}|\phi_A,\eta_A/\phi_A),$$
$$P(\boldsymbol{B}) = \prod_{j,r=1}^{J,R} \mathcal{G}(b_{jr}|\phi_B,\eta_B/\phi_B),$$

where ϕ_A, η_A, ϕ_B, and η_B are hyperparameters and $\mathcal{G}$ denotes Gamma distribution,

$$\mathcal{G}(x|\phi,\eta) = \exp\big\{(\phi-1)\log x - x/\eta - \log\Gamma(\phi) - \phi\log\eta\big\}.$$

As shown in Fig. 2.1, a_{ir} and b_{jr} tend to be smaller as ϕ_A and ϕ_B decrease. Using them together, the joint distribution of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n$, and $\boldsymbol{X}^n$ is

$$P(\boldsymbol{X}^n,\boldsymbol{S}^n,\boldsymbol{A},\boldsymbol{B}) = P(\boldsymbol{X}^n,\boldsymbol{S}^n|\boldsymbol{A},\boldsymbol{B})P(\boldsymbol{A})P(\boldsymbol{B}).$$

Figure 2.2 shows a graphical model representation.

2.2 Algorithm

This section provides two representative algorithms for NMF: Majorization Minimization (MM) and Variational Bayes (VB).

2.2.1 Majorization Minimization (MM)

The Majorization Minimization algorithm for NMF was derived by Lee and Seung [Lee and Seung, 1999, Lee and Seung, 2001]. They derived the case where the number of observed matrix $n = 1$. Here we provide the general algorithm with arbitral n.

[1] For more details of conjugate prior, see Appendix A.2.

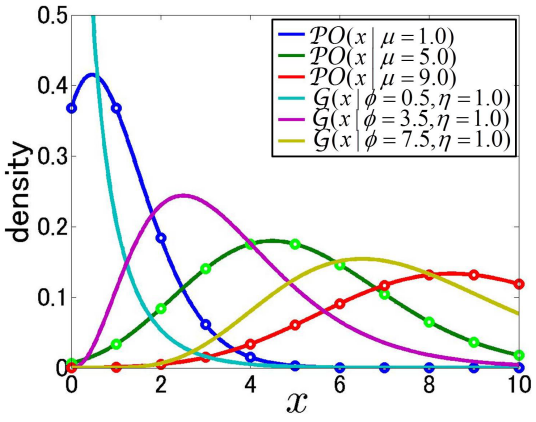


Figure 2.1: Poisson and Gamma distributions.

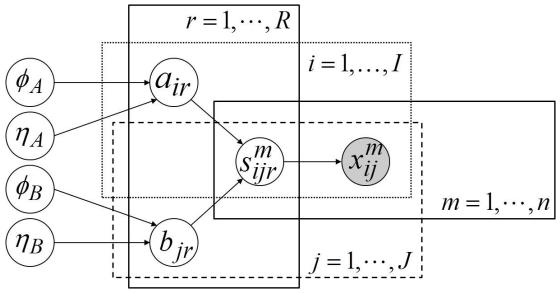


Figure 2.2: Graphical model of NMF.

MM is the algorithm that minimizes negative logarithm of likelihood function (2.1) [2] :

$$\begin{aligned} & -\log P(\boldsymbol{X}^n | \boldsymbol{A}, \boldsymbol{B}) \\ & = -\sum_{m=1}^{n} \sum_{i,j=1}^{I,J} \log \mathcal{PO}\Big(x_{ij}^m \Big| \sum_{r=1}^{R} a_{ir} b_{jr}\Big), \\ & = n \sum_{i,j=1}^{I,J} \Big\{ \hat{x}_{ij} - \bar{x}_{ij} \log(\hat{x}_{ij}) \Big\} + \sum_{m=1}^{n} \sum_{i,j=1}^{I,J} \log(x_{ij}^m!) \end{aligned} \tag{2.2}$$

where

$$\hat{x}_{ij} = \sum_{r=1}^{R} a_{ir} b_{jr}, \quad \bar{x}_{ij} = \frac{1}{n} \sum_{m=1}^{n} x_{ij}^m.$$

We define the function $\mathcal{L}$ by removing constant terms of the negative log-

[2] This is a MM for maximum likelihood estimation. For the definition of maximum likelihood estimation, see Appendix A.3. MM can be used for maximum a posterior estimation. See Appendix C.

likelihood function as follows:

$$\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = n \sum_{i,j=1}^{I,J} \left\{ \hat{x}_{ij} - \bar{x}_{ij} \log \hat{x}_{ij} \right\}. \tag{2.3}$$

Let us also define the auxiliary (majorizing) function $\mathcal{L}^+$ as

$$\mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T}) = n \sum_{i,j=1}^{I,J} \left\{ \hat{x}_{ij} - \bar{x}_{ij} \sum_r t_{ijr} \log \left(\frac{a_{ir} c_{jr}}{t_{ijr}} \right) \right\},$$

where $\boldsymbol{T} = \{t_{ijr}\}$ is auxiliary variables satisfying $\sum_r t_{ijr} = 1$ $(\forall (i,j))$. It can be verified that the auxiliary function $\mathcal{L}^+$ has following two properties:

1. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) \leq \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$
2. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \min_{\boldsymbol{T}} \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$.

Note that the equality holds if and only if

$$t_{ijr} = \frac{a_{ir} b_{jr}}{\sum_{r'} a_{ir'} b_{jr'}}. \tag{2.4}$$

In the scheme of MM [Hunter and Lange, 2004, De Leeuw, 1994], minimization of the function $\mathcal{L}$ is indirectly conducted by minimizing the auxiliary function $\mathcal{L}^+$ as follows:

1. Minimize $\mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$ w.r.t. $\boldsymbol{A}$ or $\boldsymbol{B}$.
2. Minimize $\mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$ w.r.t. $\boldsymbol{T}$ which makes $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$.

For the first step, we compute the partial derivative of $\mathcal{L}^+$ w.r.t. $\boldsymbol{A}$. The necessary condition of the local minima, which satisfies the partial derivative $\frac{\partial \mathcal{F}^+}{\partial a_{ir}} = 0$, is simplified into

$$n \sum_j b_{jr} - n \sum_j \frac{\bar{x}_{ij} b_{jr}}{a_{ir}} = 0 \Leftrightarrow a_{ir} = \frac{\sum_j \bar{x}_{ij} t_{ijr}}{\sum_j b_{jr}}. \tag{2.5}$$

For the second step, by substituting Eq. (2.4) into Eq. (2.5), we obtain the update rules of $\boldsymbol{A}$ given by Eq. (2.6). We omit the derivation of the update rules of $\boldsymbol{B}$ since the derivation are analogous to that of $\boldsymbol{A}$.

$$a_{ir} \leftarrow a_{ir} \frac{\sum_{j=1}^{J} \frac{\bar{x}_{ij}}{\hat{x}_{ij}} b_{jr}}{\sum_{j=1}^{J} b_{jr}}, \tag{2.6}$$

$$b_{jr} \leftarrow b_{jr} \frac{\sum_{i=1}^{I} \frac{\bar{x}_{ij}}{\hat{x}_{ij}} a_{ir}}{\sum_{i=1}^{I} a_{ir}}. \tag{2.7}$$

The above update rules are given by multiplicative form and thus called *multiplicative update rules*. We can confirm that the right hand side of the update

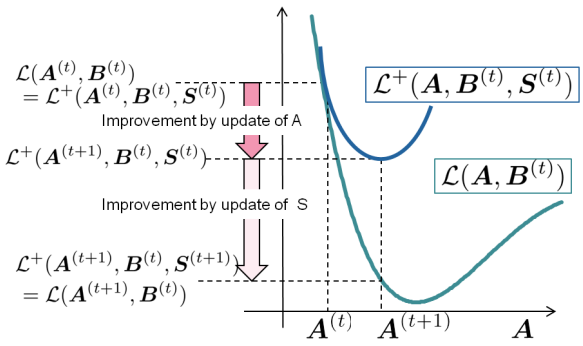


Figure 2.3: Scheme of majorization minimization (MM).

Algorithm 1 Majorization Minimization (MM) for NMF

Input: $\boldsymbol{X}^n$: input matrices, R: the number of factors

Output: $\boldsymbol{A}, \boldsymbol{B}$ such that objective function in Eq. (2.3) is minimized under the non-negative constraint.

1: initialization for $\boldsymbol{A}$ and $\boldsymbol{B}$
2: **repeat**
3: Update $\boldsymbol{A}$ by Eq. (2.6)
4: Update $\boldsymbol{B}$ by Eq. (2.7)
5: **until** Converge

equation of $\boldsymbol{A}$ is (I) always non-negative and (II) equals a_{ir} when $\bar{x}_{ij} = \hat{x}_{ij}$. Randomly setting (non-negative) initial values of factor matrices and iteratively updating the matrices following Eq. (2.6)(2.7), a factorization result is obtained. Algorithm 1 shows a pseudo code. A tuning parameter such as the learning rate doesn't exist. Note that this algorithm can deal with input matrices with missing elements by slight modification. For more details, see Appendix B.

2.2.2 Variational Bayes (VB)

The variational Bayesian (VB) algorithm is used to estimate the variational distribution, which approximates a posterior distribution of parameters and hidden variables [Attias, 1999, Attias, 2000, Jordan et al., 1999]. The VB algorithm for NMF was derived by Cemgil [Cemgil, 2009], where Cemgil derived the case when $n = 1$. For the asymptotic analysis provided in chapter 4, we generalize the algorithm for arbitrary n. The variational distribution $q(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n)$ is optimized by minimizing the functional $\bar{\mathcal{F}}[q]$, which is defined by

$$\bar{\mathcal{F}}[q] = \mathbb{E}_{q(\boldsymbol{A},\boldsymbol{B},\boldsymbol{S}^n)}\left[\log \frac{q(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n)}{P(\boldsymbol{X}^n, \boldsymbol{S}^n, \boldsymbol{A}, \boldsymbol{B})}\right], \tag{2.8}$$

under the constraint that variational distribution is independent:

$$q(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n) = q(\boldsymbol{A})q(\boldsymbol{B})q(\boldsymbol{S}^n).$$

Note that $\mathbb{E}_{q(\boldsymbol{A})q(\boldsymbol{B})q(\boldsymbol{S}^n)}$ denotes the expectation w.r.t. $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{S}^n$ which is subject to the variational distribution. Minimizing the functional $\bar{\mathcal{F}}[q]$ is equivalent to minimizing the Kullback-Leibler (KL) divergence between posterior and variational distributions because $\bar{\mathcal{F}}[q]$ can be represented as follows:

$$\bar{\mathcal{F}}[q] = -\log P(\boldsymbol{X}^n) + \mathrm{KL}(q\|p), \tag{2.9}$$

where $\mathrm{KL}(q\|p)$ is following KL divergence.

$$\mathrm{KL}(q\|p) = \mathbb{E}_{q(\boldsymbol{A},\boldsymbol{B},\boldsymbol{S}^n)}\Big[\log \frac{q(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n)}{P(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}^n|\boldsymbol{X}^n)}\Big].$$

From the optimality condition derived from the variational method, the variational distributions of $\boldsymbol{A}$ and $\boldsymbol{B}$ are gamma distributions and that of $\boldsymbol{S}$ is a multinomial distribution:

$$q(\boldsymbol{A}) = \prod_{i,r} \mathcal{G}(a_{ir}|\alpha^A_{ir}, \beta^A_{ir}), \tag{2.10}$$

$$q(\boldsymbol{B}) = \prod_{j,r} \mathcal{G}(b_{jr}|\alpha^B_{jr}, \beta^B_{jr}), \tag{2.11}$$

$$q(\boldsymbol{S}^n) = \prod_{m,i,j} \mathcal{M}\big(\boldsymbol{s}_{ij}|x^m_{ij}, \{p^S_{ijr}\}\big). \tag{2.12}$$

The consistency condition of the variational Bayesian estimation gives the following recursive formula,

$$\begin{aligned}
\alpha^A_{ir} &= \phi_A + n\bar{s}_{i\cdot r}, \\
\beta^A_{ir} &= \left(\phi_A/\eta_A + n\bar{b}_{\cdot r}\right)^{-1}, \\
\alpha^B_{jr} &= \phi_B + n\bar{s}_{\cdot jr}, \\
\beta^B_{jr} &= \left(\phi_B/\eta_B + n\bar{a}_{\cdot r}\right)^{-1}, \\
p^S_{ijr} &\propto \rho_{ijr} = \exp\left(\mathbb{E}_{q(\boldsymbol{A})q(\boldsymbol{B})}\left[\log a_{ir} + \log b_{jr}\right]\right),
\end{aligned}$$

where the statistics in above equations are computed by

$$\begin{aligned}
\bar{a}_{ir} &= \alpha^A_{ir}\beta^A_{ir}, \\
\bar{b}_{jr} &= \alpha^B_{jr}\beta^B_{jr}, \\
\bar{s}_{ijr} &= \bar{x}_{ij}p^s_{ijr}, \\
\bar{x}_{ij} &= \frac{1}{n}\sum_{m=1}^{n} x^m_{ij}, \\
\mathbb{E}_{q(\boldsymbol{A})}\left[\log a_{ir}\right] &= \Psi(\alpha^A_{ir}) + \log(\beta^A_{ir}), \\
\mathbb{E}_{q(\boldsymbol{B})}\left[\log b_{jr}\right] &= \Psi(\alpha^B_{jr}) + \log(\beta^B_{jr}),
\end{aligned}$$

where $\Psi(\cdot)$ denotes the digamma function. The VB algorithm is the recursive iteration of Eqs. (2.10), (2.11), and (2.12). Algorithm 2 shows a pseudo code.

Algorithm 2 Variational Bayes for NMF (VBNMF)

Input: $\boldsymbol{X}^n$: input matrices, R: the number of factors,
$\phi_A, \eta_A, \phi_B, \eta_B$: hyperparameters
Output: $\alpha^A_{ir}, \beta^A_{ir}, \alpha^B_{jr}, \beta^B_{jr}, p^S_{ijr}$: variational parameters of $\boldsymbol{A}$,$\boldsymbol{B}$, $\boldsymbol{S}^n$
1: Initialize $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{S}^n$.
2: **repeat**
3: Update α^A_{ir}, β^A_{ir} following Eq. (2.10).
4: Update α^B_{jr}, β^B_{jr} following Eq. (2.11).
5: Update p^S_{ijr}, $\bar{s}_{ijr}$ following Eq. (2.12).
6: **until** Converge

2.3 Viewpoint from Divergence Minimization

§ 2.1 and § 2.2.1 shows the NMF model and MM algorithm which minimize negative log-likelihood. This approach can be defined as optimization problem without a description as a statistical model.

Let us define the divergence between matrices as

$$\mathcal{D}_{KL}(\boldsymbol{X}|\hat{\boldsymbol{X}}) = \sum_{i,j=1}^{I,J} d_{KL}(x_{ij}||\hat{x}_{ij}),$$

where $d_{KL}(x_{ij}||\hat{x}_{ij})$ is the generalized Kullback Leibler (KL) divergence:

$$d_{KL}(x_{ij}||\hat{x}_{ij}) = x_{ij} \log \frac{x_{ij}}{\hat{x}_{ij}} - x_{ij} + \hat{x}_{ij}.$$

Figure 2.4 shows the shape of the generalized KL. Since $\mathcal{D}_{KL}$ is equivalent to the negative log-likelihood (Eq. (2.2)) when $n = 1$ by ignoring constant terms, Algorithms which minimize negative log-likelihood can be seen as the algorithm for solving following optimization problem:

$$\begin{aligned} &\arg\min_{\boldsymbol{A},\boldsymbol{B}} \mathcal{D}_{KL}(\boldsymbol{X}|\hat{\boldsymbol{X}}), \\ &\text{s.t. } \boldsymbol{A} \geq 0, \boldsymbol{B} \geq 0. \end{aligned} \tag{2.13}$$

where $\boldsymbol{A} \geq 0$ means that all elements of $\boldsymbol{A}$ are nonnegative. The algorithm for arbitral n corresponds to

$$\begin{aligned} &\arg\min_{\boldsymbol{A},\boldsymbol{B}} \mathcal{D}_{KL}(\bar{\boldsymbol{X}}|\hat{\boldsymbol{X}}), \\ &\text{s.t. } \boldsymbol{A} \geq 0, \boldsymbol{B} \geq 0, \end{aligned} \tag{2.14}$$

where $\bar{\boldsymbol{X}} = \{\bar{x}_{ij}\}_{i,j=1}^{I,J}$ is the average of the input matrices, because the argument of the minimum of a sum of the divergences to each matrix equals to that of

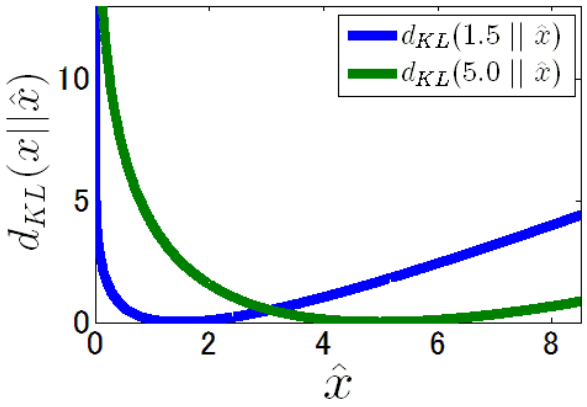


Figure 2.4: Generalized KL divergence.

the minimum of the divergence to the average matrix:

$$\arg\min_{\boldsymbol{A},\boldsymbol{B}} \sum_{m=1}^{n} \mathcal{D}_{KL}(\boldsymbol{X}_m|\hat{\boldsymbol{X}}) = \arg\min_{\boldsymbol{A},\boldsymbol{B}} \mathcal{D}_{KL}(\bar{\boldsymbol{X}}|\hat{\boldsymbol{X}}).$$

When considering the minimization of the negative log likelihood, we can estimate the factor matrices by considering the optimization problem using only the average matrix of the input matrices. However, when we consider maximum a posterior estimation whose objective function is defined with both log-likelihood term and some regularization terms, the number of observed data n affects the degree of the contributions of each terms. Please see Appendix C for more details.

2.4 Variational Free Energy (VFE)

§ 2.2.2 shows the variational Bayesian algorithm for NMF. The minimum value of the objective functional $\bar{\mathcal{F}}[q]$, $\bar{\mathcal{F}}_{vb}$, which referred to as the variational free energy (VFE) (e.g. [MacKay, 2003]) is important quantity for VBNMF.

$$\bar{\mathcal{F}}_{vb} = \min_{q(\boldsymbol{A})q(\boldsymbol{B})q(\boldsymbol{S}^n)} \bar{\mathcal{F}}[q].$$

We can interpret $\bar{\mathcal{F}}_{vb}$ as the objective value at the (optimal) result of the VB algorithm. Note that the reason the term "free energy" is used is that the VFE is an upper bound of free energy, $\mathcal{F}$:

$$\mathcal{F} = -\log P(\boldsymbol{X}^n) = -\log \int P(\boldsymbol{X}^n, \boldsymbol{S}^n, \boldsymbol{A}, \boldsymbol{B}) d\boldsymbol{A} d\boldsymbol{B} d\boldsymbol{S}^n.$$

We have already shown that VFE is the upper bound by Equation (2.9) since KL divergence $KL(q\|p) \geq 0$. Figure 2.5 shows the relation between free energy and

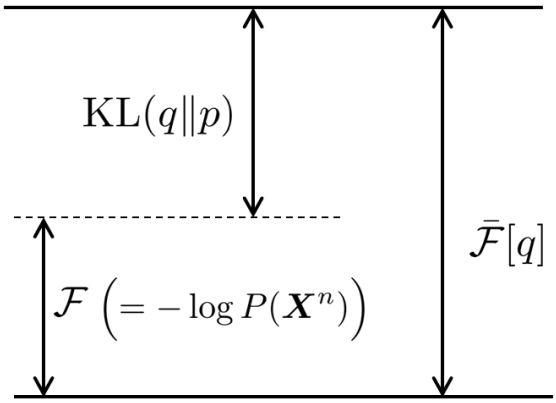


Figure 2.5: Free energy and variational free energy (VFE).

variational free energy. The smaller VFE indicates that a pair of a statistical model and a prior is more appropriate for a given training data, according to the free energy.

The probability density functions $q(\boldsymbol{A})$ and $q(\boldsymbol{B})$ that minimize the objective functional are called the variational posterior distributions. For a_{ir} and b_{jr} which are subject to $q(\boldsymbol{A})$ and $q(\boldsymbol{B})$, the number of elements of the set

$$\Big\{0 \leq r \leq R\,;\ \text{ for all } (i,j)\quad a_{ir} \to 0, b_{jr} \to 0\quad (n \to \infty)\Big\}$$

is referred to as the number of the asymptotic redundant factors R_{red}, and $\hat{R} = R - R_{red}$ is called the effective number of factors. In general, $\hat{R}$ depends on both the true distribution and hyperparameters.

By applying eqs.(2.10), (2.11), and (2.12) to eq.(2.8), it follows that

$$\bar{\mathcal{F}}_{vb} = F_A + F_B + F_X, \tag{2.15}$$

where

$$\begin{aligned}
F_A &= \sum_{i,r}\Big\{\big(\alpha^A_{ir} - \phi_A\big)\Psi(\alpha^A_{ir}) - \phi_A \log(\beta^A_{ir}) + (\frac{\phi_A}{\eta_A})\bar{a}_{ir} \\
&\quad + \log\frac{\Gamma(\phi_A)}{\Gamma(\alpha^A_{ir})} + \phi_A \log(\frac{\eta_A}{\phi_A}) - \alpha^A_{ir}\Big\}, \\
F_B &= \sum_{j,r}\Big\{\big(\alpha^B_{jr} - \phi_B\big)\Psi(\alpha^B_{jr}) - \phi_B \log(\beta^B_{jr}) + (\frac{\phi_B}{\eta_B})\bar{b}_{jr} \\
&\quad + \log\frac{\Gamma(\phi_B)}{\Gamma(\alpha^B_{jr})} + \phi_B \log(\frac{\eta_B}{\phi_B}) - \alpha^B_{jr}\Big\}, \\
F_X &= \sum_{i,j}\Big\{\sum_r n\bar{a}_{ir}\bar{b}_{jr} + \sum_m \log\Gamma(x^m_{ij}+1) - n\bar{x}_{ij}\log\big(\sum_r \rho_{ijr}\big)\Big\}.
\end{aligned}$$

Hence $\bar{\mathcal{F}}_{vb}$ can be numerically calculated for a given training data X^n. In chapter 4, we show its theoretical behaviors and give the hyperparameter design method.

2.5 Literature of NMF

At the end of this chapter, we provide literature of NMF.

NMF first gained attention by Lee and Seung's work [Lee and Seung, 1999]. Lee and Seung shows that NMF can extract the parts-based representation from facial image, each parts of which corresponds to e.g., eye, nose, etc. This result could not be obtained by other standard algorithms such as principle component analysis. Therefore, NMF was recognized as the method that can extract interpretable, parts-based representation from data. Since then, NMF was applied to feature extraction [Li et al., 2001, Hoyer, 2004], speech signal processing [Smaragdis and Brown, 2003, Févotte et al., 2009], and text mining [Xu et al., 2003, Pauca et al., 2004, Shahnaz et al., 2006]. The number of applications to real world problems is still increasing and we can find, e.g., email analysis [Berry and Browne, 2005], recommendation [Zhang et al., 2006], blog's network analysis [Chi et al., 2007], community discovery [Wang et al., 2011], hot topic extraction from social media [Saha and Sindhwani, 2012, Endo et al., 2015] social curation service analysis [Takeuchi et al., 2013] and analysis of ideological stance in social network service [Lahoti et al., 2018].

We showed the two representative algorithms for NMF, majorization minimization (MM) [Hunter and Lange, 2004, De Leeuw, 1994] and variational Bayes (VB) [Attias, 1999, Attias, 2000, Jordan et al., 1999]. It is experimentally shown that VB is robust to noise and sparsity [Brouwer et al., 2017]. The other algorithms such as a variant of gradient descent [Cichocki et al., 2009] and fully Bayesian method using markov chain monte carlo [Schmidt et al., 2009] has been developed, although they are not scope of this thesis.

Several relations between NMF and other machine learning algorithms are also clarified. For example, Ding et al. [Ding et al., 2008] proved that NMF is equivalent to probabilistic latent semantic indexing [Hofmann, 1999], which is the basis of latent dirichlet allocation [Blei et al., 2003], a well-known probabilistic model for document analysis. Thus, it is no exaggeration to say that NMF is a key machine learning algorithm.

Chapter 3

NMF for Inconsistent Resolution Matrices

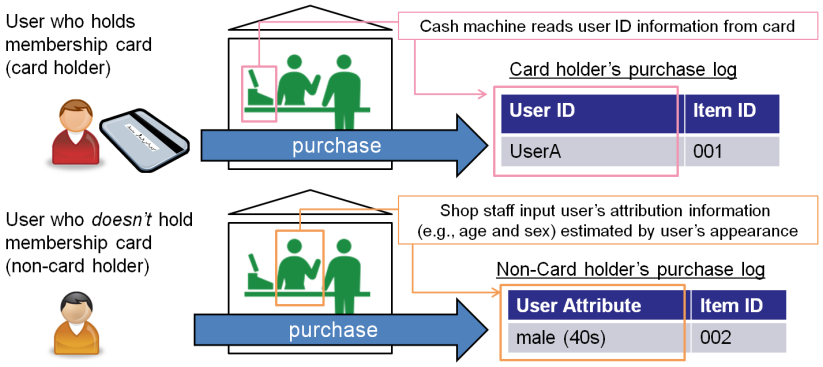
This chapter provides a new NMF model for analyzing a combination of datasets with different granularity. [1]

3.1 Motivation

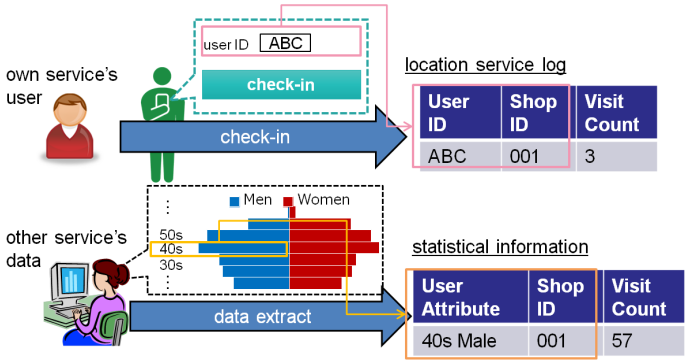
Due to the difficulty of exhaustive data collection and the need to protect personal information, it is becoming more urgent to be able to analyze multiple datasets that have different levels of granularity, for example, user-individual data such as "how many times an item is purchased by a user" and user-group data such as "how many times a shop is visited by users of the same age". Therefore, we consider the problem of inconsistent resolution dataset analysis, which is to analyze a combination of datasets with different granularity. High resolution datasets (such as user-individual data) capture the events that occurred in fine detail such as individual visits and purchases and said to have fine grain size. Low resolution datasets (user group data) offer less detail, i.e. coarser granularity.

We provide two examples that require inconsistent resolution dataset analysis. The first example is an analysis of data collected in the retail industry (Fig. 3.1(a)). Currently, many retail shops collect information about users by issuing them with membership cards. However, since not all shoppers will have a membership card, exhaustively data collection, some purchase log entries do not have the identification information (ID) of the membership card; Instead they contain only information on the sex and age of the user as input by the shop staff from assessments of the appearance of the user at the sales point. Therefore, the collected data consists of user-individual data and user-group

[1] The material in this chapter was presented in part at the IEICE Transactions on Information and Systems (Japanese Edition) [Kohjima et al., 2017], and all the figures of this chapter are reused from [Kohjima et al., 2017] under the permission of the IEICE.



(a) members' and non-members' logs



(b) own/other companies' logs

Figure 3.1: Example of datasets requiring inconsistent resolution analysis.

data. When the purchase log contains little member user data, inconsistent resolution dataset analysis can be useful by allowing use of the purchase data of non-member users. The second example is analysis of the combined datasets of different companies (Fig. 3.1(b)). The social data provided by location information services e.g., Foursquare [2] and Yelp [3], omits the data of individual users to protect personal information; only visit logs of user groups are disclosed, for example, how many "women" have visited a certain shop. Therefore, inconsistent resolution dataset analysis is required if we are to analyze a dataset created by combining user-individual data and the above social data.

In this study, we propose a new method for inconsistent resolution dataset analysis. The proposed method is a probabilistic model based on nonnegative matrix factorization (NMF) [Lee and Seung, 1999, Lee and Seung, 2001, Cichocki et al., 2009]. First of all, to introduce the basic setting of inconsistent resolution dataset analysis, we focus on the situation where two assumptions are

[2] http://gnip.com/sources/foursquare/
[3] http://www.yelp.com/dataset_challenge

satisfied: (A1) common user set exists, (A2) data are independent and identically distributed. We use assumptions (A1) and (A2) and the NMF formulation to propose probabilistic nonnegative inconsistent resolution matrix factorization (*pNimf*) that can jointly analyze high and low resolution data. *pNimf* makes it possible to analyze data more accurately than the methods that use a single set of data. For example, applying *pNimf* to the purchase logs of the members/non-members mentioned above, improves the accuracy of missing value complementation in the matrix, making it possible to more accurately predict the quantities purchased by members / non-members. In addition, it is possible to extract purchasing patterns that reflect the purchasing tendencies of both members and non-members.

pNimf is derived by considering the data generative process that covers the latent high resolution data that underlies the low resolution matrix. Latent high resolution data can be defined from assumption (A1) and relation between high resolution data and low resolution data can be deduced from assumption (A2). While it is not possible to assume that assumptions (A1) and (A2) hold for all problems, approaches that use the relationship described in this paper can be the basis for solving a lot of general problems. We also show the situation that diverges from the above two assumptions, and an extended version of the proposed method is provided for cases that demand different assumptions.

3.2 Proposed Method

3.2.1 Formulation

In this section we focus on the problem of inconsistent resolution dataset analysis in situations where two assumptions are satisfied: (A1) - common user assumption, (A2) independently and identically distributed assumption. Before providing a mathematical representation of these assumptions, we give an intuitive explanation. A certain supermarket issued a members card in December to all users of the store. Clearly then the shop's sales records contain no personal details prior to December. As shown in Figure 3.2, the purchase history for November consists of low resolution data, while that for December contain high resolution data. Note that user attribute information such as sex and age (est.) is recorded in the purchase history for November, In this example, assumption (A1) is that the set of all shop users in November and December are equal (whether or not they purchased any item) Assumption (A2) states that each user will make the same product purchases in November and December. We will explain using this example of purchase log analysis, and the symbol definitions follow this example. However, our research is not limited to this example, and more general circumstances are explained in §3.4.

Definition of Symbols: Let I, J and K represent the number of users, items, and attributes, respectively. We define the element of $\boldsymbol{X}$, x_{ij}, as the number of purchases of item j by user i in Dec. and the element of $\boldsymbol{Y}$, y_{kj}, is the number of times item j was purchased by users with attribute k in Nov. Each $\boldsymbol{X}$

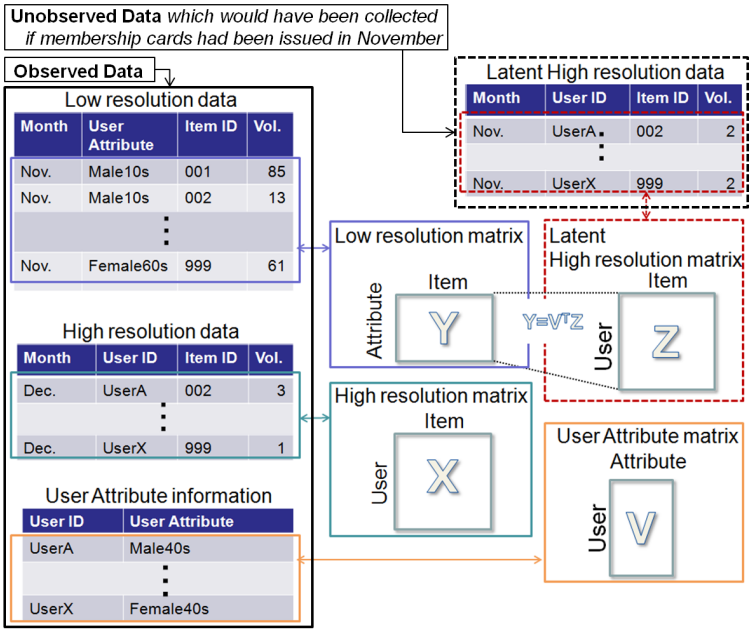


Figure 3.2: Example of observed and unobserved data.

and $\boldsymbol{Y}$ are taken to be the high-resolution matrix and the low-resolution matrix, respectively. We also assume that user's attribute information is available. This assumption is natural because such data is required, for example, when the user creates the membership card. $\boldsymbol{V} = \{v_{ik}\}_{i,k=1}^{I,K}$, whose element $v_{ik} \in \{0, 1\}$ is set to 1 if the attribute of user i is k, otherwise 0.

Latent High Resolution Matrix: Next, we define the *latent high resolution matrix*, $\boldsymbol{Z}$. This matrix plays an important role in our model. We define $\boldsymbol{Z}$ as the matrix that corresponds to the high-resolution data in Nov., i.e. data which would have collected if membership cards had been issued in Nov. Since only low-resolution data is collected in Nov., $\boldsymbol{Z}$ is the unobserved latent high resolution data which lies under the low resolution data. We are usually unable to know the set of users that exist behind $\boldsymbol{Z}$ and the number of rows of $\boldsymbol{Z}$ cannot be defined. To resolve this, we use (A1), which we formally define as follows: *user population of high-resolution data* $\boldsymbol{X}$ *and that of latent high-resolution data* $\boldsymbol{Z}$ *are identical*. (A1) allows us to define the number of the rows of $\boldsymbol{Z}$ as being identical to $\boldsymbol{X}$, I. Then, we define element z_{ij} as the number of purchases of item j by user i in Nov. Importantly, this definition yields a relation between $\boldsymbol{Y}$ and $\boldsymbol{Z}$, $\boldsymbol{Y} = \boldsymbol{V}^T\boldsymbol{Z}$. This comes from the fact that y_{kj} is equal to the summation of z_{ij} over user i with attribute k, i.e. $y_{kj} = \sum_i v_{ik} z_{ij}$.

3.2.2 Model

This subsection presents the proposed model. Let $\boldsymbol{A} := \{a_{ir}\}_{i,r=1}^{I,R}$ and $\boldsymbol{B} := \{b_{jr}\}_{j,r=1}^{J,R}$ be the user factor matrix and item factor matrix, respectively. R is the number of factors. Each vector of factor matrices $(a_{i1}, \cdots, a_{iR}), (b_{j1}, \cdots, b_{jR})$ is interpreted as the latent feature of user i and item j. We also define $\hat{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{B}^T$; its element is written as $\hat{x}_{ij} = \sum_r a_{ir} b_{jr}$. Since the Poisson distribution is frequently used to model count data such as purchase log and visit count, we adopt it for our model. NMF models the probability of generating matrix $\boldsymbol{X}$ as

$$P(\boldsymbol{X}|\boldsymbol{A}, \boldsymbol{B}) = \prod_{i,j=1}^{I,J} \mathcal{PO}(x_{ij}|\hat{x}_{ij}), \tag{3.1}$$

where $\mathcal{PO}$ is the Poisson distribution:

$$\mathcal{PO}(x_{ij}|\hat{x}_{ij}) = \exp\{-\hat{x}_{ij} + x_{ij}\log(\hat{x}_{ij}) - \log\Gamma(x_{ij}+1)\}.$$

Note that our model can be extended, in an analogous manner, to the case that other probability distributions such as Gaussian are adopted.

We derive the proposed method based on the data generative process summarized as follows: (i) define the probability distribution that generates both $\boldsymbol{X}$ and $\boldsymbol{Z}$. (ii) use (A2) *iid assumption* , which we formally define as follows: elements of $\boldsymbol{X}$ and $\boldsymbol{Z}$ that have the same indices, x_{ij} and z_{ij}, follow the identical probability distribution (in this case, Poisson distribution with parameter $\hat{x}_{ij}$ as in Eq. (3.1)) and they are mutually independent. (A2) helps to extract factors which are independent of month. (iii) use the relation between $\boldsymbol{Z}$ and $\boldsymbol{Y}$ ($y_{kj} = \sum_i v_{ik} z_{ij}$) explained in the previous section. Combining these parts, the joint distribution of $\boldsymbol{X}$, $\boldsymbol{Z}$, $\boldsymbol{Y}$ is written as

$$\begin{aligned} &P(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{V}) \\ &= \prod_{i,j} \mathcal{PO}(x_{ij}|\hat{x}_{ij}) \mathcal{PO}(z_{ij}|\hat{x}_{ij}) \prod_{k,j} \delta(y_{kj} - \sum_i v_{ik} z_{ij}), \end{aligned} \tag{3.2}$$

where $\delta(\cdot)$ is the delta function. Figure 3.3(a) shows a graphical model representation. By explicitly modeling the generation of *latent high-resolution matrix* $\boldsymbol{Z}$, we can naturally define the probability distribution of all matrices. However, since the size of $\boldsymbol{Z}$ is $I \times J$, which is considerable, it is desirable to work with more convenient probabilistic models.

The key to practical implementation lies in a characteristic of Poisson distributions: the sum of Poisson-distributed random variables is also a Poisson-distributed random variable, i.e., closed under addition. In our model, z_{ij} represents Poisson-distributed random variables and y_{kj} is their summation. Thus, we can marginalize out $\boldsymbol{Z}$ from Eq. (3.2) which yields the following equation:

$$\begin{aligned} P(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{V}) &= \int P(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{V}) d\boldsymbol{Z} \\ &= \prod_{i,j} \mathcal{PO}(x_{ij}|\hat{x}_{ij}) \prod_{k,j} \mathcal{PO}(y_{kj}|\hat{y}_{kj}), \end{aligned} \tag{3.3}$$

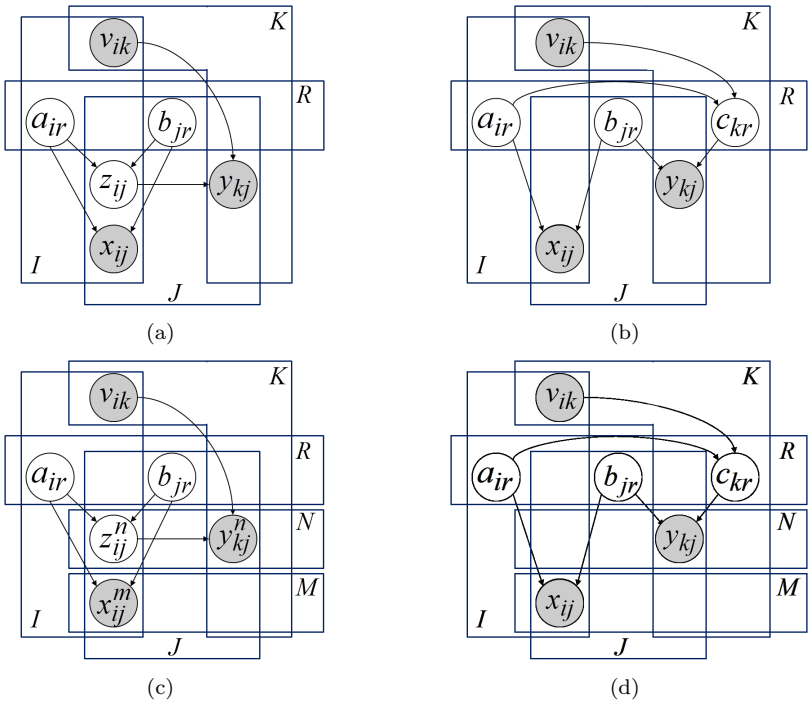


Figure 3.3: Graphical models. Shaded nodes indicate observed variables. Figure (a) presents the original definition of the proposed model described in Eq. (3.2). By marginalizing out $\boldsymbol{Z}$, Fig. (b), which is given by Eq. (3.3), is obtained. Figure (c)(d) represents the generalized model stated in §3.4.

$$\text{where} \quad \hat{y}_{kj} = \sum_{r=1}^{R} c_{kr} b_{jr} \quad \text{and} \quad c_{kr} = \sum_{i=1}^{I} v_{ik} a_{ir}. \tag{3.4}$$

Figure 3.3(b) shows a graphical model representation. Considering that $\boldsymbol{C} := \{c_{kr}\}_{k,r=1}^{K,R}$ is the attribute latent factor matrix, Eq. (3.3) can be interpreted as factorizing the high-resolution matrix and low-resolution matrix simultaneously, while retaining the relation between factor matrices $\boldsymbol{A}$ and $\boldsymbol{C}$ using $\boldsymbol{V}$ ($\boldsymbol{C} = \boldsymbol{V}^T\boldsymbol{A}$ as in Eq. (3.4)). Thus, we call this proposal probabilistic non-negative inconsistent-resolution matrix factorization (*pNimf*). Figure 3.4 shows the factorization form. Note that removing the linear equality relation between factor matrices, $\boldsymbol{C} = \boldsymbol{V}^T\boldsymbol{A}$, *pNimf* is reduced a CMF method (NMMF) [Takeuchi et al., 2013]. Thus, *pNimf* can be seen as subsuming NMMF.

The optimization problem for estimating factor matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ is

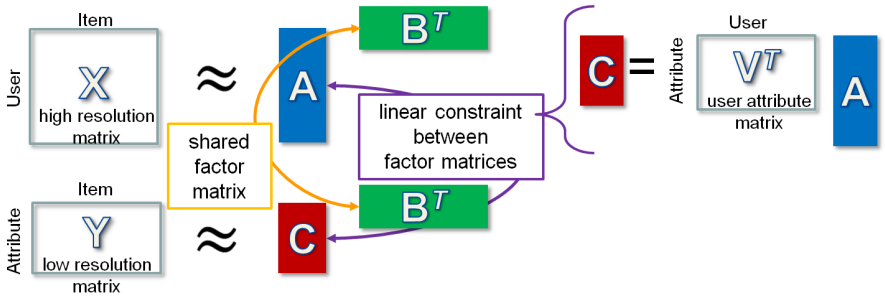


Figure 3.4: Factorization form that corresponds to Fig. 3.3(b).

summarized as follows:

$$\begin{aligned} &\arg\max_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} \log P(\boldsymbol{X},\boldsymbol{Y}|\boldsymbol{A},\boldsymbol{B},\boldsymbol{V}), \\ &\text{s.t. } \boldsymbol{A}\geq 0,\boldsymbol{B}\geq 0,\boldsymbol{C}\geq 0,\boldsymbol{C}=\boldsymbol{V}^T\boldsymbol{A} \end{aligned} \tag{3.5}$$

where $\boldsymbol{A} \geq 0$ means that all elements of $\boldsymbol{A}$ are nonnegative. Note that for the above optimization problem, Eq. (3.5) is equivalent to

$$\begin{aligned} &\arg\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} \{\mathcal{D}_{KL}(\boldsymbol{X}|\hat{\boldsymbol{X}})+\mathcal{D}_{KL}(\boldsymbol{Y}|\hat{\boldsymbol{Y}})\}, \\ &\text{s.t. } \boldsymbol{A}\geq 0,\boldsymbol{B}\geq 0,\boldsymbol{C}\geq 0,\boldsymbol{C}=\boldsymbol{V}^T\boldsymbol{A}. \end{aligned} \tag{3.6}$$

Note that the derivation shown in this subsection is valid if a probability distribution which is closed under summation is adopted.

3.2.3 Algorithm

As shown in the next subsection, the following algorithm can be used to solve the optimization problem posed by Eq. (3.5).

$$a_{ir}^{\text{new}} \leftarrow a_{ir}\frac{\left(\sum_j \frac{x_{ij}}{\hat{x}_{ij}}b_{jr}+\sum_k\sum_j v_{ik}\frac{y_{kj}}{\hat{y}_{kj}}b_{jr}\right)}{\sum_j b_{jr}+\sum_k\sum_j v_{ik}b_{jr}}, \tag{3.7}$$

$$b_{jr}^{\text{new}} \leftarrow b_{jr}\frac{\left(\sum_i \frac{x_{ij}}{\hat{x}_{ij}}a_{ir}+\sum_k \frac{y_{kj}}{\hat{y}_{kj}}c_{kr}\right)}{\sum_i a_{ir}+\sum_k c_{kr}}, \tag{3.8}$$

$$c_{kr}^{\text{new}} \leftarrow \sum_i v_{ik}a_{ir}. \tag{3.9}$$

Update rules for $\boldsymbol{A}$, $\boldsymbol{B}$ are given in "multiplicative form". The right hand side of the update for $\boldsymbol{A}$ is (I) always nonnegative and (II) equals a_{ir} when $x_{ij}=\hat{x}_{ij}$ and $y_{kj}=\hat{y}_{kj}$. By iteratively updating the parameters following Eq. (3.7)-(3.9) from their initial values, the algorithm converges to (local) minima; proof is

Algorithm 3 probabilistic nonnegative inconsistent resolution matrix factorization (*pNimf*)

Input: $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{V}$: input data, R: rank of approximation
Output: $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$: factor matrices
1: initialization for $\boldsymbol{A}, \boldsymbol{B}$ and set $\boldsymbol{C} = \boldsymbol{V}^T\boldsymbol{A}$.
2: **repeat**
3: Update $\boldsymbol{A}$ and $\boldsymbol{C}$ by Eq. (3.7)(3.9)
4: Update $\boldsymbol{B}$ by Eq. (3.8)
5: **until** a stopping condition is met

provided in §3.3. Pseudo code of the method is shown in Algorithm 3. Note that an almost analogous algorithm is derived when matrix $\boldsymbol{X}$ and/or $\boldsymbol{Y}$ has missing values.

3.2.4 Algorithm Derivation

In this subsection, we derive the multiplicative update rules given by Eq. (3.7)(3.8)(3.9). We define the function $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B})$, where constant terms of the objective function in Eq. (3.6) are removed and matrix $\boldsymbol{C}$ is replaced by $\boldsymbol{V}^T\boldsymbol{A}$ as follows:

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = &\sum_{i,j}\Big\{\Big(\hat{x}_{ij} - x_{ij}\log(\hat{x}_{ij})\Big\} \\ &+ \sum_{k,j}\Big\{\hat{y}_{kj} - y_{kj}\log(\hat{y}_{kj})\Big\}.\end{aligned} \tag{3.10}$$

We minimize $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B})$ following the optimization scheme of majorization minimization (MM) [Hunter and Lange, 2004, De Leeuw, 1994], similar to [Lee and Seung, 2001]. Let us define the auxiliary (majorizing) function $\mathcal{L}^+$ as

$$\begin{aligned}&\mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T}) \\ &= \sum_{i,j}\Big\{\Big(\hat{x}_{ij} - x_{ij}\sum_{r=1}^{R} s_{ijr}\log\Big(\frac{a_{ir}b_{jr}}{s_{ijr}}\Big)\Big\} \\ &\quad + \sum_{k,j}\Big\{\hat{y}_{kj} - y_{kj}\sum_{r=1}^{R} t_{kjr}\log\Big(\frac{(\sum_i v_{ik}a_{ir})b_{jr}}{t_{kjr}}\Big)\Big\},\end{aligned} \tag{3.11}$$

where $\boldsymbol{S} = \{s_{ijr}\}$ and $\boldsymbol{T} = \{t_{ijr}\}$ are auxiliary variables satisfying $\sum_r s_{ijr} = 1$ $(\forall(i,j))$, $\sum_r t_{kjr} = 1$ $(\forall(k,j))$. It can be verified that auxiliary function $\mathcal{L}^+$ has the following two properties:

1. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) \leq \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T})$
2. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \min_{\boldsymbol{S},\boldsymbol{T}} \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T})$.

Note that the equality holds if and only if

$$s_{ijr} = \frac{a_{ir}b_{jr}}{\sum_{r'=1}^{R} a_{ir'}b_{jr'}}, \quad t_{kjr} = \frac{\left(\sum_i v_{ik}a_{ir}\right)b_{jr}}{\sum_{r'=1}^{R}\left(\sum_i v_{ik}a_{ir'}\right)b_{jr'}}. \tag{3.12}$$

Since the partial derivative of $\mathcal{L}^+$ w.r.t. $\boldsymbol{A}$ is given by

$$\frac{\partial \mathcal{L}^+}{\partial a_{ir}} = \sum_j b_{jr} + \sum_{k,j} v_{ik}b_{jr} - \sum_j \frac{x_{ij}s_{ijr}}{a_{ir}} - \sum_{j,k} \frac{v_{ik}y_{kj}t_{kjr}}{\sum_{i'} v_{i'k}a_{i'r}},$$

the necessary condition of the local minima, $\frac{\partial \mathcal{L}^+}{\partial a_{ir}} = 0$, can be simplified to

$$a_{ir} = \frac{\sum_j x_{ij}s_{ijr} + \sum_{j,k} \frac{v_{ik}a_{ir}y_{kj}t_{kjr}}{\sum_i v_{ik}a_{ir}}}{\sum_j b_{jr} + \sum_{k,j} v_{ik}b_{jr}}. \tag{3.13}$$

By substituting Eq. (3.12) into Eq. (3.13), we obtain the multiplicative update rules for $\boldsymbol{A}$ given by Eq. (3.7). We omit the derivation of the update rules for $\boldsymbol{B}$ since the derivation is exactly same as that of standard NMF. The update for $\boldsymbol{C}$ is given by the linear constraint.

3.3 Theoretical Analysis

Here we confirm the convergence property of the algorithm.

Theorem 1 *Objective function $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B})$ is monotonically decreasing under the update by Eq.* (3.7)(3.8)(3.9)*. The divergence is invariant if and only if $\boldsymbol{A}, \boldsymbol{B}$ are at a stationary point.*

This theorem indicates that the algorithm reaches a local minimum by update iteration. The theorem is proven by showing that $\mathcal{L}^+$ decreases with each optimization step. We need to prove the following two lemmas to prove the theorem.

Lemma 1 *$\mathcal{L}^+$ is a convex function w.r.t. $\boldsymbol{A}$ and thus $\boldsymbol{A}$ satisfying Eq.* (3.13) *is the global minimum if the other parameters are fixed.*

Proof *Since $-\log(a_{ir})$ is convex and the sum of convex functions is convex, we need to show $-\log(\sum_i v_{ik}a_{ir})$ is convex. Since its Hessian is given by*

$$-\frac{\partial^2 \log(\sum_i v_{ik}a_{ir})}{\partial a_{ir}\partial a_{i'r'}} = \delta_{rr'}\frac{v_{ik}v_{i'k}}{\left(\sum_i a_{ir}\right)^2},$$

where $\delta_{rr'} = 1$ if $r = r'$ and 0 otherwise, it can be expressed by, using a non-degenerate matrix W,W^TW. Therefore, Hessian is positive definite, and thus convex. □

Lemma 2 *The objective $\mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T})$ is minimized w.r.t. $\boldsymbol{S}$ and $\boldsymbol{T}$ when $\boldsymbol{S}$ and $\boldsymbol{T}$ equals Eq.* (3.12) *and $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \min_{\boldsymbol{S},\boldsymbol{T}} \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T})$ holds.*

Proof *By applying Jensen's inequality to the term in Eq.* (3.10)*,*

$$-\log(\hat{x}_{ij}) \leq -\sum_r s_{ijr} \log(\frac{a_{ir}b_{jr}}{s_{ijr}}),$$
$$-\log(\hat{y}_{kj}) \leq -\sum_r t_{kjr} \log(\frac{c_{kr}b_{jr}}{t_{kjr}})$$

holds, and since Eq. (3.12) *is the equality condition, this concludes the proof.* □

The theorem follows from the application of the above lemmas.

Proof *Let us denote the parameter and the auxiliary variables that satisfy* $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{S}, \boldsymbol{T})$ *as* $\boldsymbol{A}^{old}$*,* $\boldsymbol{B}^{old}$*,* $\boldsymbol{S}^{old}$*,* $\boldsymbol{T}^{old}$*. We also denote* $\boldsymbol{A}$ *after the first step of the MM given by Eq.* (3.13) *as* $\boldsymbol{A}^{new}$ *and* $\boldsymbol{S}, \boldsymbol{T}$ *after the second step given by Eq.* (3.12) *as* $\boldsymbol{S}^{new}, \boldsymbol{T}^{new}$*. From lemma 1 and lemma 2,*

$$\mathcal{L}^+(\boldsymbol{A}^{new}, \boldsymbol{S}^{old}, \boldsymbol{T}^{old}) \leq \mathcal{L}^+(\boldsymbol{A}, \boldsymbol{S}^{old}, \boldsymbol{S}^{old}) \ (\forall \boldsymbol{A}),$$
$$\mathcal{L}^+(\boldsymbol{A}^{new}, \boldsymbol{S}^{new}, \boldsymbol{T}^{new}) \leq \mathcal{L}^+(\boldsymbol{A}^{new}, \boldsymbol{S}, \boldsymbol{T}) \ (\forall \boldsymbol{S}, \boldsymbol{T}).$$

Note that we omit the notation of $\boldsymbol{B}$*. Since* $\mathcal{L}(\boldsymbol{A}^{old}) = \mathcal{L}^+(\boldsymbol{A}^{old}, \boldsymbol{S}^{old}, \boldsymbol{T}^{old})$ *and* $\mathcal{L}(\boldsymbol{A}^{new}) = \mathcal{L}^+(\boldsymbol{A}^{new}, \boldsymbol{S}^{new}, \boldsymbol{T}^{new})$*,* $\mathcal{L}(\boldsymbol{A}^{new}) \leq \mathcal{L}(\boldsymbol{A}^{old})$ *holds. Since proof for the update of* $\boldsymbol{B}$ *is analogous, this completes the proof.* □

3.4 Generalization of Algorithms

We now explain the more general scenario that *pNimf* can be applied to. In §3.2.1, we gave the example in which both high-resolution data and low-resolution data are one month purchase logs. However, as long as assumptions (A1) and (A2) are satisfied, *pNimf* could be applied to any problem with theoretical support. Moreover, *pNimf* can deal with multiple high-resolution and low-resolution data by generalizing the data generative process. Let M be the number of high-resolution data entries and $\boldsymbol{X}^m = \{x_{ij}^m\}$ is the m-th high-resolution matrix. Similarly, let N be the number of low-resolution data entries and $\boldsymbol{Y}^n = \{y_{kj}^n\}$ is the n-th low-resolution matrix. Each m(or n) need not correspond to a period of time, e.g. day, week and month (unlike the previous example) and it may instead be an indicator of location such as prefecture and country in which the data was collected. By extending the data generative process represented by Fig. 3.3(a) to Fig. 3.3(c), the estimation procedure is obtained by slight modification of the update rules given by Eq. (3.7)(3.8) as follows:

$$a_{ir}^{\text{new}} \leftarrow a_{ir} \frac{\left(M \sum_j \frac{\bar{x}_{ij}}{\hat{x}_{ij}} b_{jr} + N \sum_k \sum_j v_{ik} \frac{\bar{y}_{kj}}{\hat{y}_{kj}} b_{jr}\right)}{M \sum_j b_{jr} + N \sum_k \sum_j v_{ik} b_{jr}}, \tag{3.14}$$

$$b_{jr}^{\text{new}} \leftarrow b_{jr} \frac{\left(M \sum_i \frac{\bar{x}_{ij}}{\hat{x}_{ij}} a_{ir} + N \sum_k \frac{\bar{y}_{kj}}{\hat{y}_{kj}} c_{kr}\right)}{M \sum_i a_{ir} + N \sum_k c_{kr}}, \tag{3.15}$$

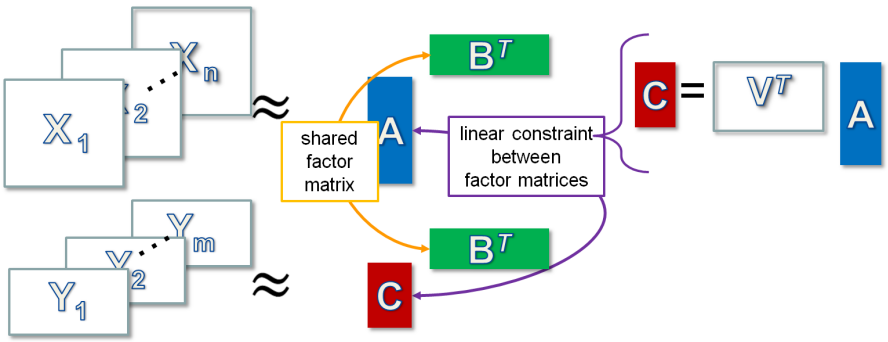


Figure 3.5: Factorization form that corresponds to Fig. 3.3(d).

where, $\bar{x}_{ij} = \frac{1}{M}\sum_{m=1}^{M} x_{ij}^{m}$, $\bar{y}_{kj} = \frac{1}{N}\sum_{n=1}^{N} y_{kj}^{n}$. Coresponding factorization form is shown in Fig. 3.5.

3.5 Experiment

3.5.1 Setting

We evaluate the performance of our method using synthetic data.

We constructed matrices with sizes of $I = 100$, $J = 100$, $K = 10$ using the probabilistic model given by Eq. (3.2). We prepared $\boldsymbol{V}$ whose elements $v_{ik} = 1$ if k is equal to the quotient of i/K and $v_{ik} = 0$ otherwise. Matrices $\boldsymbol{A}$, $\boldsymbol{B}$ are generated by Gamma distribution and high/low resolution matrixes $\boldsymbol{X}$ and $\boldsymbol{Y}$ were prepared with different levels of sparsity.

In our experiments, we used a test set log likelihood to evaluate performance. We split the elements of matrix $\boldsymbol{X}$ into a training dataset and a test dataset and computed the log likelihood of the elements in the test. Test data were treated as missing values in the training phase. Log likelihood of the test data set is defined as $\frac{1}{|\mathcal{T}|}\sum_{(i,j)\in\mathcal{T}} \log \mathcal{PO}\left(x_{ij}\middle|\hat{x}_{ij}\right)$, where $\mathcal{T}$ is the set of element indexes in the test data and $|\cdot|$ indicates the number of elements in the set. We prepared 10 pairs of training and test datasets by randomly extracting 5% of non-zero elements as the test data.

For comparison, we considered the following methods. (1)NMF [Lee and Seung, 1999], traditional method which uses only high-resolution matrix $\boldsymbol{X}$. (2)NMMF [Takeuchi et al., 2013], an NMF-based state-of-the-art CMF method that uses both $\boldsymbol{X}$ and $\boldsymbol{Y}$. The weight parameter of NMMF is chosen from the candidates $\alpha = 0.1, 0.5, 1.0$. We report the result for $\alpha = 1.0$ since it yielded the best result among the candidates.

Table 3.1: Results from synthetic data: test log likelihood for $\boldsymbol{X}$ determined with different sparseness values. Average and standard deviation are shown. Larger values are better. Scores and standard deviation are divided by ten in the 99% sparseness setting.

Sparseness	R	NMF	NMMF	*pNimf*
$\boldsymbol{X}$: 50% $\boldsymbol{Y}$: 10%	5	-2.77(±0.17)	-2.71(±0.10)	**-2.66**(±0.09)
	10	-2.72(±0.18)	-2.54(±0.09)	**-2.42**(±0.04)
	20	-16.7(±20.1)	-3.11(±0.18)	**-3.09**(±0.21)
$\boldsymbol{X}$: 90% $\boldsymbol{Y}$: 40%	5	-6.71(±2.13)	-4.11(±0.97)	**-3.48**(±0.69)
	10	-5.26(±0.92)	-3.28(±0.45)	**-2.96**(±0.23)
	20	-21.5(±4.38)	-7.57(±1.28)	**-5.72**(±0.48)
$\boldsymbol{X}$: 99% $\boldsymbol{Y}$: 80%	5	-5.44(±1.82)	-3.62(±1.73)	**-2.04**(±1.12)
	10	-6.64(±3.52)	-7.05(±2.62)	**-4.59**(±1.67)
	20	-17.0(±32.7)	-7.43(±3.37)	**-6.64**(±3.13)

3.5.2 Results

Table 3.1 shows the results for the synthetic data. Although the three methods have comparable performance when the sparseness of $\boldsymbol{X}$ is 50% and $R = 5, 10$, NMMF and *pNimf* outperform NMF when the sparseness is 90%, and *pNimf* is superior to NMMF when the sparseness is 99%. This indicates that proposed method has better performance when the input matrix is very sparse. It seems that the linear relation between factor matrices using user's attribute information supports *pNimf* in handling the difference in resolution and thus achieving better factorization results.

3.6 Discussion

3.6.1 Application to Real Purchase Log Data

We also evaluated both quantitative and qualitative performance of *pNimf* using real purchase log data [Kohjima et al., 2017, Kohjima et al., 2016]. The result (Table. 2 in [Kohjima et al., 2017]) shows that *pNimf* outperforms NMF and NMMF using test set log likelihood as a performance metric. This indicates the effectiveness of the proposed method for real data. Moreover, the purchase patterns extracted by the proposed method is shown in Fig. 9 in [Kohjima et al., 2016]. It is found that e.g., the users whose attributes are "Male 35 to 49 generations" and "full-time employees" purchase lots of "coffee" and "tobacco" and the the users whose attributes are "FeMale 35 to 49 generations" and "housewife" purchase more "yogurt" and "milk". This indicates the proposed method can extract interpretable patterns from real purchase log data.

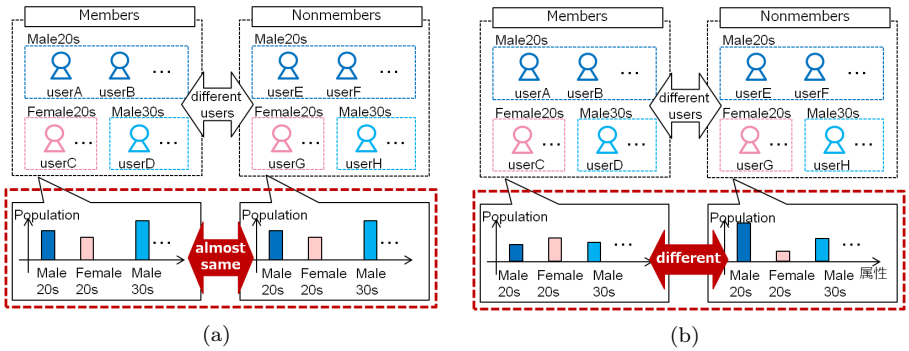


Figure 3.6: An example where (a) members and non-members have almost the same population and (b) members and non-members have different populations. Proposed method shown in § 3.2 can be applied to the case of (a). However, it is not appropriate for the case of (b) since the members and non-members have greatly different purchase volumes.

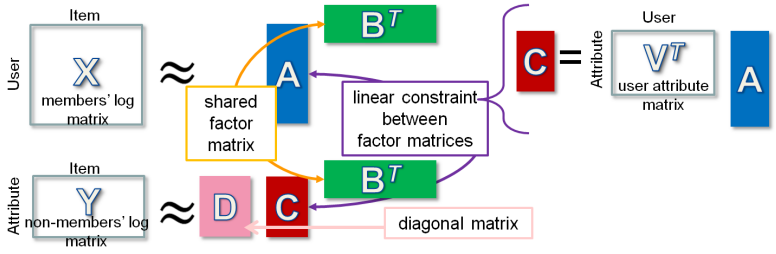


Figure 3.7: Factorization form of the extended method.

3.6.2 Further Extension

In the previous section, we focused on inconsistent resolution dataset analysis with two assumptions (A1) and (A2), proposed a new stochastic model, and verified its effectiveness. The point to note here is that since the relationship that can be introduced between the high resolution matrix and the low resolution matrix can change depending on the problem setting, the proposed probabilistic model may need some modification for some problems in inconsistent resolution dataset analysis. Accordingly, we show examples of how new methods can be developed by extending the proposed model.

Here we consider inconsistent resolution dataset analysis for the case where the member/non-member purchase logs are created by processes different from those indicated by § 3.2.1. In the example of § 3.2.1, since we assumed that membership cards were issued from December, we were able to satisfy (A1) common user set assumption that users in November and users in December were the

same. However, for data collected at shops that can be used by non-member users, such as convenience stores, assumption (A1) no longer holds since the members and the non-members always exist together and represent different user groups. An explanation for this is made below using Figure 3.6.

Figure 3.6 shows examples of possible situations when members and non-member users are different. The difference between Fig. 3.6 (a) and (b) is whether the population for each attribute in members and non-members is almost the same or very different. The proposed method indicated by § 3.2 has some validity in the case of (a), it loses validity in the case of (b). This is because assumptions (A1) and (A2) imply that "the total purchase amount of each item for each attribute is the almost same for members and non-members" and it is generally appropriate for (a), whereas it is clearly inappropriate in the setting of (b).

As a new assumption, we consider the approach that introduces a new assumption (A3), the attribute purchase quantity proportionality assumption, that is, "member purchase history is roughly proportional to non-member purchase history". Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be a high resolution matrix representing members' log, and a low resolution matrix representing non-members' log, respectively. Since the members' purchase log of attribute k is $\sum_{i=1}^{I} v_{ik}\boldsymbol{x}_i$, and the non-members' purchase log is $\boldsymbol{y}_k$, the proportional relation of assumption (A3) is represented by the equation $\boldsymbol{y}_k \propto \sum_{i=1}^{I} v_{ik}\boldsymbol{x}_i$.

If there are a certain number of members with attribute k, it is considered quite natural to make this assumption. Therefore, we consider a factorization form that holds this proportional relation on between $\hat{\boldsymbol{X}}$, $\hat{\boldsymbol{Y}}$. By defining the diagonal matrix $\boldsymbol{D} := diag(\{d_k\}_{k=1}^{K})$ whose elements d_k represent the proportionality constant of attribute k, the following equation holds given the proportional relationship:

$$\hat{\boldsymbol{Y}} = \boldsymbol{D}\boldsymbol{V}^T\hat{\boldsymbol{X}}. \tag{3.16}$$

Using factorization form $\hat{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{B}^T$ for $\hat{\boldsymbol{X}}$ and substituting it into Eq. (3.16), the factorization form for $\boldsymbol{Y}$ becomes

$$\hat{\boldsymbol{Y}} = \boldsymbol{D}\boldsymbol{C}\boldsymbol{B}^T, \quad \boldsymbol{C} = \boldsymbol{V}^T\boldsymbol{A}. \tag{3.17}$$

Summarizing the above yields the following probabilistic model:

$$\begin{aligned} &p(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}, \boldsymbol{V}) \\ &= \prod_{i,j} \mathcal{PO}(x_{ij}|\hat{x}_{ij}) \prod_{k,j} \mathcal{PO}(\beta y_{kj}|\beta\hat{y}_{kj}), \end{aligned}$$

$$\text{where} \quad \hat{y}_{kj} = \sum_{r=1}^{R} d_k c_{kr} b_{jr} \quad \text{and} \quad c_{kr} = \sum_{i=1}^{I} v_{ik} a_{ir},$$

where β is a weight parameter that controls the contribution of non-member data. Figure 3.7 shows the factorization form of this model. The difference from the factorization form shown in Fig. 3.4 of §3.2 is the existence of diagonal

Algorithm 4 extended model of *pNimf*

Input: $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{V}$: input data, R: rank of approximation
Output: $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$: factor matrices
1: initialization for $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{D}$ and set $\boldsymbol{C} = \boldsymbol{V}^T\boldsymbol{A}$.
2: **repeat**
3: Update $\boldsymbol{A}$ and $\boldsymbol{C}$ by Eq. (3.18)(3.20)
4: Update $\boldsymbol{B}$ by Eq. (3.19)
5: Update $\boldsymbol{D}$ by Eq. (3.21)
6: **until** a stopping condition is met

matrix $\boldsymbol{D}$ and thus the former is an extended factorization form. Parameter update rules for $a_{ir}, b_{jr}, c_{kr}, d_k$ are derived as follows:

$$a_{ir}^{\text{new}} \leftarrow a_{ir} \frac{\left(\sum_j \frac{x_{ij}}{\hat{x}_{ij}} b_{jr} + \beta \sum_k d_k v_{ik} \sum_j \frac{y_{kj}}{\hat{y}_{kj}} b_{jr}\right)}{\sum_j b_{jr} + \beta \sum_k d_k v_{ik} \sum_j b_{jr}}, \tag{3.18}$$

$$b_{jr}^{\text{new}} \leftarrow b_{jr} \frac{\left(\sum_i \frac{x_{ij}}{\hat{x}_{ij}} a_{ir} + \beta \sum_k \frac{y_{kj}}{\hat{y}_{kj}} d_k c_{kr}\right)}{\sum_i a_{ir} + \beta \sum_k d_k c_{kr}}, \tag{3.19}$$

$$c_{kr}^{\text{new}} \leftarrow \sum_i v_{ik} a_{ir}, \tag{3.20}$$

$$d_k^{\text{new}} \leftarrow \frac{\sum_j y_{kj}}{\sum_r c_{kr} (\sum_j b_{jr})}. \tag{3.21}$$

Pseudo code is shown in Algorithm 4.

3.6.3 Related Works

In recent years, collective matrix factorization (CMF) or multiple matrix factorization (MMF) techniques have been proposed for multiple dataset analysis [Singh and Gordon, 2008]. A CMF/MMF extension of NMF called Nonnegative Multiple Matrix Factorization (NMMF) [Takeuchi et al., 2013] has been described [Lee and Choi, 2009, Takeuchi et al., 2013]. Factorization form of NMMF is shown in Fig. 3.8. These techniques combine multiple matrices and have been reported to offer better performance than is possible when using only a single matrix. However, these methods are not designed to handle datasets that have different resolutions. For a context different from CMF/MMF, Aimoto et al. proposed a method for combining information of aggregated data (corresponding to low resolution matrix in this paper) in matrix factorization [Aimoto and Kashima, 2013]. However, this method is specialized for situations where the datasets with different granularity represent exactly the same data, making it unsuitable as a basic method for general inconsistent resolution dataset analysis.

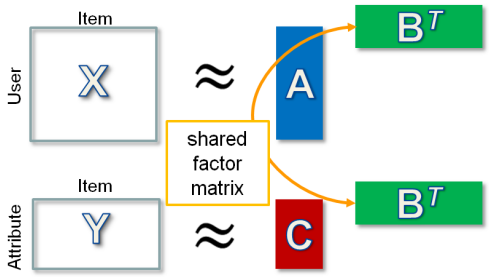


Figure 3.8: Factorization form of NMMF [Takeuchi et al., 2013].

Chapter 4

Theoretical Analysis

This chapter provides a theoretical analysis of variational Bayesian NMF (VBNMF) [Kohjima and Watanabe, 2017] [1].

4.1 Motivation

The standard algorithms for NMF such as majorization minimization [Lee and Seung, 2001] and variational Bayes (VB) [Cemgil, 2009], require the setting of the number of factors. Since the *true* number of factors of the input matrix is unknown, the chosen number of factors may be larger than the true one. This setting frequently appears in practical model selection scenarios. In this case, the factorization result cannot be uniquely determined, as shown in Fig. 4.1. Because both Result case 1, in which redundant factors vanish, and Result case 2, in which redundant factors remain, can exactly reconstruct the input matrix, we cannot distinguish which result is better from the difference from the input matrix. In order to compare the results, the factorization results should be evaluated by the value of hyperparameters.

In this paper, we theoretically prove the following two results. (i) The factorization results of the variational Bayesian NMF algorithm (VBNMF) are changed according to hyperparameters. (ii) Its critical line is explicitly given by the size of the input matrix. Figure 4.2 shows our theoretical results. Depending on whether the hyperparameters are in the area above or below the critical line $\phi_A I + \phi_B J = (I + J)/2$, the factorization results drastically change, where I and J are sizes of the input matrix and ϕ_A and ϕ_B are hyperparameters. We call this phenomenon *phase transition* of the VBNMF. Clarification of the phase transition structure provides useful insight in the hyperparameter design.

These results are derived by analyzing the minimum value of the objective

[1] Reprinted by permission from Springer Nature: Springer Nature, Artificial Neural Networks and Machine Learning -ICANN 2017, 26th International Conference on Artificial Neural Networks, Phase Transition Structure of Variational Bayesian Nonnegative Matrix Factorization, Masahiro Kohjima and Sumio Watanabe, Springer (2017)

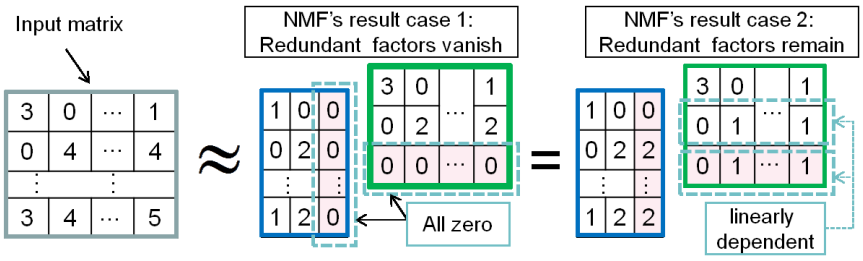


Figure 4.1: An example of NMF's factorization results when redundant factor exists.

function of the VBNMF, which is called the variational free energy (VFE). In this paper, we consider the setting that the amount of data (the number of observed matrices) is sufficiently large and identify the optimal number of factors through an asymptotic analysis. Note that the setting where multiple matrices are observed arises in recent application, e.g., purchase data analysis [Kohjima et al., 2015] and traffic data analysis [Deng et al., 2016].

4.2 Theoretical Result

This section provides our main theoretical result. In the proof of the theorem, we assume that the following assumption is satisfied.

Assumption 1 *The set of training data* $\boldsymbol{X}^n$ *is independently generated by*

$$P(\boldsymbol{X}|\boldsymbol{X}^*) = \prod_{i,j=1}^{I,J} \mathcal{PO}\left(x_{ij}\middle|\left(\boldsymbol{X}^*\right)_{ij}\right),$$

where $\boldsymbol{X}^*$ *is a true nonnegative matrix. The nonnegative rank [Cohen and Rothblum, 1993] of* $\boldsymbol{X}^*$ *is denoted by* R^**. Nonnegative rank is defined as the smallest rank of the nonnegative matrix factorization, which it is not smaller than the standard rank. For more details, see [Vavasis, 2009].*

Our main theorem clarifies the effect of hyperparameters on the result of the VBNMF.

Main Theorem *Suppose assumption 1 is satisfied and* $R \geq R^*$*. Then, as the number of observed matrices* $n \to \infty$*, the asymptotic form of the VFE is given by*[2]

$$\bar{\mathcal{F}}_{vb} = \mathcal{E} + \lambda_{vb}\log(n) + \mathcal{O}_p(1), \tag{4.1}$$

[2] $\mathcal{O}_p$ is the order notation of random variables. A sequence of random variables X_n is said to be $\mathcal{O}_p(1)$ if it is bounded in probability [Van der Vaart, 2000].

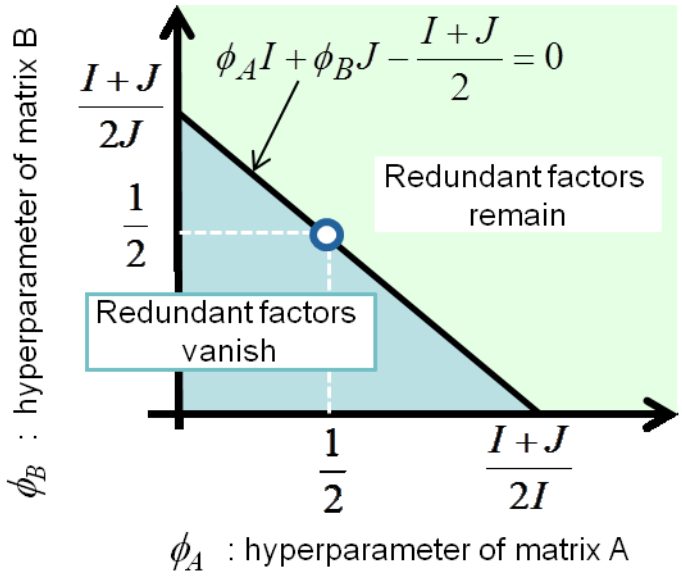


Figure 4.2: Phase transition diagram obtained by our analysis

where $\mathcal{E}$ *is the empirical entropy defined by*

$$\mathcal{E} = -\sum_{m=1}^{n} \log p(\boldsymbol{X}_m | \boldsymbol{X}^*), \tag{4.2}$$

and

$$\lambda_{vb} = \begin{cases} (\phi_A I + \phi_B J)(R - R^*) + \frac{I+J}{2} R^* & \left(\textit{if } \phi_A I + \phi_B J < \frac{(I+J)}{2} \right) \\ \frac{I+J}{2} R & \left(\textit{othewise} \right). \end{cases} \tag{4.3}$$

The effective number of factors $\hat{R}$ *satisfies*

$$\hat{R} = \begin{cases} R^* & \left(\text{ if } \phi_A I + \phi_B J < (I+J)/2 \right), \\ R & \left(\text{ otherwise } \right). \end{cases} \tag{4.4}$$

The proof is shown in the following sections. Here, we provide an interpretation of the theorem. Equation (4.4) shows the mathematical law that the optimal number of factors $\hat{R}$ is determined by the hyperparameters. Figure 4.2 is the diagram that describes the relation between hyperparameters and $\hat{R}$, which we call *phase transition*. In the area under the critical line $\phi_A I + \phi_B J = (I+J)/2$, $\hat{R}$ equals the *true* number of true factors, R^*. On the other hand, above the critical line, the optimal number of factors equals the number of factors of the statistical model, R. Since ϕ_A and ϕ_B are the parameters of the gamma prior, our result coincides with the fact smaller values make $\boldsymbol{A}$ and $\boldsymbol{B}$ sparse. Analogous to the optimal number of factors, Equation (4.1) and (4.3) show that

the behavior of VFE is also changed whether the hyperparameters are above or under the critical line.

Figure 4.3 visualizes the optimal number of factors, $\hat{R}$, and the VFE when $I = J = 5$. The critical line is the straight line connecting $(\phi_A, \phi_B) = (0.0, 1.0)$ and $(1.0, 0.0)$. Figure 4.4 also shows the VFE varying the size of input matrices. As the number of columns, J, increases, the angle of the critical line changes and then the region under the line changes. For example, $(\phi_A, \phi_B) = (0.2, 0.6)$ is under the line when $I = J = 5$ but is above the line when $I = 5, J = 20$. Therefore, our theorem shows that the hyperparemters should be carefully chosen considering the size of input matrices. We discuss the application of the theorem for hyperaparameter design in § 4.5.1

4.3 Experiment

In this section, we confirm the validity of the main theorem through numerical experiment. We prepared the true matrix $\boldsymbol{X}^* = \{x^*_{ij}\} \in \mathbb{R}_+^{5\times J}$ as $x^*_{ij} = \max(4 - (j\%5), 1)$ if $i = 0, 1, 2$ and otherwise, $x^*_{ij} = \max((j\%5) - 1, 1)$. Note that $c\%d$ denotes the remainder when c is divided by d. Obviously, nonnegative rank of $\boldsymbol{X}^*$, $R^* = 2$. Using this matrix, we generated matrices $\boldsymbol{X}^n$ following Eq. (1) and applied the VBNMF. Using the matrices and the result of the VBNMF, we computed the empirical entropy, $\mathcal{E}$, and the experimental value of the VFE, $\bar{\mathcal{F}}_{vb}$. To obtain the experimental values, we ran the VBNMF 2000 times with a random initialization and set the maximum number of iterations to 1000. We checked whether the asymptotic value of VFE in the main theorem was satisfied since it is the key of our theoretical results. We conducted an experiment involving varying the size of input matrices and the number of factors.

Figure 4.5 shows the results when the hyperparameters were set to $\phi_A = \eta_A = \phi_B = \eta_B = 1.0$. The horizontal axis represents the number of observed matrices with log scale. The solid line represents the theoretical value $\lambda_{vb}\log(n)$, and the angle corresponds to λ_{vb}. The marked point represents the experimental value $\bar{\mathcal{F}}_{vb} - \mathcal{E}$. The dashed line represents the linear regression line to the experimental values. Since Eq. (4.1) contains the $\mathcal{O}_p(1)$ constant term, there exists a small difference between the solid and dashed lines. Therefore, we need to focus on the angle of the solid and dashed lines since it indicates the coefficient with respect to $\log(n)$. We can easily confirm that the angles of the lines are almost the same. This means our theory effectively explains the experimental results.

4.4 Proof of Main Theorem

Finally, this section provides the proof of main theorem.

Theorem 2 *As the number of matrices $n \to \infty$, the asymptotic form of the VFE $\bar{\mathcal{F}}_{vb}$ is given by $\bar{\mathcal{F}}_{vb} = \mathcal{E} + \{\min_{R^*\le\hat{R}\le R}\Lambda(R, \hat{R})\}\log(n) + \mathcal{O}_p(1)$, where $\Lambda(R, \hat{R}) = (\phi_A I + \phi_B J)R - \left(\phi_A I + \phi_B J - \frac{I+J}{2}\right)\hat{R}$.*

Note that $\hat{R}$ is *the effective number of factors*, which does *not* vanish. The main theorem is immediately obtained from theorem 2.

Proof of Main Theorem *From the definition of* $\Lambda(R, \hat{R})$*, in the case of* $\phi_A I + \phi_B J < (I + J)/2$*, the smaller* $\hat{R}$ *is, the smaller* $\bar{\mathcal{F}}_{vb}$*. Therefore,* $R^*_{vb} = R^*$*. In the another case, a larger* $\hat{R}$ *is better, and* $R^*_{vb} = R$*. Substituting* R^*_{vb} *into* $\hat{R}$*, Eq.* (4.3) *is obtained.* □

Thus, we need to prove theorem 2. It requires following three lemmas, Lemmas 3, 4, and 5. The proofs of these lemmas are provided in the end of this chapter.

Lemma 3 *As the number of matrices* $n \to \infty$*, the first and second terms of Eq.* (2.15)*,* F_A *and* F_B*, are given by* $F_A = \left\{\phi_A IR - \left(\phi_A - \frac{1}{2}\right) I\hat{R}\right\} \log(n) + \mathcal{O}_p(1)$ *and* $F_B = \left\{\phi_B JR - \left(\phi_B - \frac{1}{2}\right) J\hat{R}\right\} \log(n) + \mathcal{O}_p(1)$*.*

Lemma 4 F_X *in Eq.* (2.15) *is given by* $F_X = -\log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) + \mathcal{O}_p(1)$*.*

Lemma 5 *Suppose assumption 1 is satisfied and* R *is not less than* R^**. Then,* $\mathcal{F}[q]$ *is minimized if and only if* $\hat{R}$ *satisfies* $R^* \leq \hat{R} \leq R$*. Moreover, as the number of matrices* $n \to \infty$*, the asymptotic form of* F_X *is given by* $F_X = \mathcal{E} + \mathcal{O}_p(1)$*.*

By applying Lemmas 3, 4, 5, theorem 2 is proven.

Proof (Theorem 2) *From Eq.* (2.15)*,* $\bar{\mathcal{F}}_{vb} = F_A + F_B + F_X$ *holds. Using the lemma 3, 4 and 5, we can obtain the asymptotic form with* $\hat{R}$*. Since the VFE with* $\hat{R}$ *is minimized when* $\hat{R}$ *minimizes the* $\Lambda(R, \hat{R})$*, we complete the proof.* □

4.5 Discussion

4.5.1 Hyperparameter Design

We introduce the hyperparameter design method based on the main theorem. Here we discuss three examples and provide corresponding recommended settings.

Design Method 1: It is required that the redundant factors vanish.

From our theorem, if $\phi_A I + \phi_B J < \frac{(I+J)}{2}$ is satisfied, i.e., (ϕ_A, ϕ_B) is within the region under the critical line, the redundant factors vanish. However, as we discussed in the previous section, the region depends on the size of the input matrices, I, J. Therefore, we recommend that both hyperparameters are set to be smaller than 0.5, $\phi_A, \phi_B < 0.5$. Because it is always under the critical line regardless of the size of the input matrix. Figure. 4.6(a) shows the area of recommended setting.

Design Method 2: It is required that (i) the redundant factors vanish and (ii) matrix $\boldsymbol{A}$ is more sparse than matrix $\boldsymbol{B}$.

The second condition (ii) is sometimes necessary in practical data analysis, for example, when NMF is applied to the clustering problem [Xu et al., 2003,

Shahnaz et al., 2006]. In the clustering, either matrix $\boldsymbol{A}$ or $\boldsymbol{B}$ is regarded as a cluster assignment indicator matrix and it is preferable that the indicator matrix is more sparse than the other. In this case, $\phi_A I + \phi_B J < (I+J)/2$ and $\phi_A < \phi_B$ should be satisfied. As similar to the example 1, if $\phi_A, \phi_B < 0.5$ then $\phi_A I + \phi_B J < (I+J)/2$, we recommend the hyperparameters are set to a value within the red triangle region in Fig. 4.6(b).

From above examples, it is confirmed that the result of our theorem is useful for hyperparameter design.

4.5.2 Related Works

A statistical model is called a *singular model* if the mapping from parameter to distribution is not one to one and Fisher information matrix is not positive definite [Watanabe, 2009]. It is known that not only NMF but also a lot of modern statistical models such as Gaussian mixture models, Bernoulli mixture model, hidden Markov models, Bayesian networks are singular models [Watanabe, 2009].

In the field of learning theory, free energy has been regarded as an important quantity to be clarified. Since it was shown that their asymptotic behaviors depend on a model and differ in Bayesian estimation of singular models [Watanabe, 2001a, Watanabe, 2001b], unlike the behavior of regular statistical model [Akaike, 1974, Schwarz, 1978], the theoretical analysis has been actively conducted [Yamazaki and Watanabe, 2003, Yamazaki and Watanabe, 2002, Yamazaki and Watanabe, 2005, Aoyagi and Watanabe, 2005, Aoyagi, 2010].

Variational free energy, which we focus on this chapter, have also been the targets of theoretical analysis in learning theory [Watanabe and Watanabe, 2006, Watanabe and Watanabe, 2007, Kaji et al., 2010, Hosino et al., 2005, Watanabe et al., 2009, Nakajima et al., 2014]. The existence of the phase transition structure is also confirmed in aforementioned modern statistical models. However, theoretical analysis of NMF has not been conducted. Our study will contribute to the field of learning theory by proving the existence of phase transition structure in the case of the NMF.

4.6 Proof of Lemmas

In the proof of lemmas, we use the following two inequalities of the digamma and log-digamma functions [Alzer, 1997], for $x > 0$,

$$\frac{1}{2x} < \log(x) - \Psi(x) < \frac{1}{x}., \tag{4.5}$$

$$0 \leq \log\Gamma(x) - \left\{(x - \frac{1}{2})\log(x) - x + \frac{1}{2}\log 2\pi\right\} \leq \frac{1}{12x}. \tag{4.6}$$

The inequality (4.5) indicates that the difference by substituting $\log(x)$ for $\Psi(x)$ can be bounded. The substitution for $\log\Gamma(x)$ is analogous.

Proof (Lemma 3) *Using the inequality* (4.5)(4.6),

$$\begin{aligned} F_A = \sum_{(i,r)} \Big\{ -\big(\phi_A - \frac{1}{2}\big) \log(\alpha_{ir}^A) - \phi_A \log(\beta_{ir}^A) + (\phi_A/\eta_A)\bar{a}_{ir} \\ + \log\Gamma(\phi_A) + \phi_A \log(\eta_A/\phi_A) \Big\} + \mathcal{O}_p(1). \end{aligned} \tag{4.7}$$

We denote the effective number of factors as $\hat{R}$*. Without loss of generality, we can assume that* $\bar{s}_{i\cdot r} = \bar{s}_{\cdot jr} = 0$ *is satisfied for all* r, $\hat{R} < r \leq R$*. By substituting Eq.* (2.10) *into Eq.* (4.7)*, we obtain*

$$\begin{aligned} F_A &= \sum_{i}^{I} \sum_{r=1}^{\hat{R}} \Big\{ -\big(\phi_A - \frac{1}{2}\big) \big\{ \log(n) + \log(\frac{\phi_A}{n} + \bar{s}_{i\cdot r}) \big\} \Big\} \\ &\quad + \sum_{i}^{I} \sum_{r=\hat{R}}^{R} \Big\{ -\big(\phi_A - \frac{1}{2}\big) \log\big(\phi_A\big) \Big\} \\ &\quad + \sum_{(i,r)} \Big\{ \phi_A \log(n) + \phi_A \log(\bar{b}_{\cdot r}) + (\phi_A/\eta_A)\bar{a}_{ir} \Big\} \\ &\quad + IR \log\Gamma(\phi_A) + IR\phi_A \log(\eta_A/\phi_A) + \mathcal{O}_p(1) \\ &= \Big\{ \phi_A IR - \big(\phi_A - \frac{1}{2}\big) I\hat{R} \Big\} \log(n) + \mathcal{O}_p(1). \end{aligned}$$

Derivation for F_B *is analogous.* □

Proof (Lemma 4) *It is sufficient to show*

$$\sum_{i,j} \bar{x}_{ij} \log\big(\sum_r \rho_{ijr}\big) = \sum_{i,j} \bar{x}_{ij} \log\big(\sum_r \bar{a}_{ir}\bar{b}_{jr}\big) + \mathcal{O}_p(1). \tag{4.8}$$

Since ρ_{ijr} *is given by Eq.* (2.12)*, we complete the proof by constructing the upper and lower bounds using inequality* (4.5)*. The upper bound can be derived as follows:*

$$\begin{aligned} &\sum\nolimits_{(i,j)} \bar{x}_{ij} \log\big(\sum_r \rho_{ijr}\big) \\ &< \sum_{(i,j)} \bar{x}_{ij} \log\Big\{ \sum_r \alpha_{ir}^A \exp\Big(-\frac{1}{2\alpha_{ir}^A}\Big) \cdot \beta_{ir}^A \cdot \alpha_{jr}^B \exp\Big(-\frac{1}{2\alpha_{jr}^B}\Big) \cdot \beta_{jr}^B \Big\} \\ &= \sum_{(i,j)} \bar{x}_{ij} \log\left\{ \sum_r \bar{a}_{ir}\bar{b}_{jr} \exp\Big(-\frac{1}{2\alpha_{ir}^A}\Big) \exp\Big(-\frac{1}{2\alpha_{jr}^B}\Big) \right\} \\ &\leq \sum_{(i,j)} \bar{x}_{ij} \log\Big(\sum_r \bar{a}_{ir}\bar{b}_{jr}\Big) + \mathcal{O}_p(1). \end{aligned}$$

Similarly, we can also derive the lower bound,

$$\sum_{(i,j)} \bar{x}_{ij} \log\big(\sum_r \rho_{ijr}\big) \geq \sum_{(i,j)} \bar{x}_{ij} \log\Big(\sum_r \bar{a}_{ir}\bar{b}_{jr}\Big) + \mathcal{O}_p(1).$$

This completes the proof. □

Proof (Lemma 5) *The proof is completed by showing the upper and lower bounds. We first show the upper bound. Let us denote the optimal number of factors that minimizes the VFE as R^*_{vb}. From the definition of the VFE and the results of Lemmas 3 and 4, the VFE can be written as*

$$\bar{\mathcal{F}}_{vb} = \Lambda(R, R^*_{vb}) \log(n) - \log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) + \mathcal{O}_p(1).$$

Let us consider that R^*_{vb} is larger than R^*. In this case, $\bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}$ can reconstruct the true matrix $\boldsymbol{X}^*$. Since the VFE is defined as the minimum value of functional $\bar{\mathcal{F}}[q]$, the VFE satisfies

$$\begin{aligned}
\bar{\mathcal{F}}_{vb} &= \Lambda(R, R^*_{vb}) - \log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) + \mathcal{O}_p(1) \\
&\leq \Lambda(R, R^*_{vb}) - \log P(\boldsymbol{X}^n | \boldsymbol{X}^*) + \mathcal{O}_p(1) \\
&= \Lambda(R, R^*_{vb}) + \mathcal{E} + \mathcal{O}_p(1).
\end{aligned}$$

Then, $-\log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) \leq \mathcal{E}$ is shown. When R^*_{vb} is less than R^*,

$$\log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) - \log P(\boldsymbol{X}^n | \boldsymbol{X}^*) = \mathcal{O}_p(n)$$

holds then the VFE in this case is larger than that in the previous case. Thus, we can consider that R^*_{vb} is larger than R^*. Next, we show the lower bound by using the classical statistical learning theory. Let us define the probabilistic model that has a parameter of Poisson distribution for all the elements of matrix (i, j), $\tilde{\boldsymbol{\mu}} = \{\tilde{\mu}_{ij}\}_{i,j=1}^{I,J}$. By using this model, the probability of generating matrix $\boldsymbol{X}^n$ can be written as $\tilde{P}(\boldsymbol{X}_m | \tilde{\boldsymbol{\mu}}) = \prod_{(i,j)} \mathcal{PO}(x^m_{ij} | \tilde{\mu}_{ij})$. From the statistical learning theory, the maximum likelihood estimator (MLE) $\tilde{\boldsymbol{\mu}}_{ML}$ satisfies

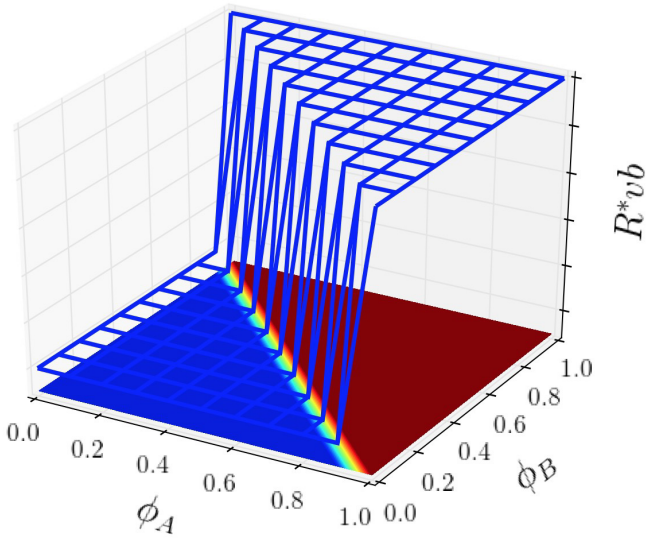
$$\frac{1}{n} \sum_{m=1}^{n} \log \frac{P(\boldsymbol{X}_m | \boldsymbol{X}^*)}{\tilde{P}(\boldsymbol{X}_m | \tilde{\boldsymbol{\mu}}_{ML})} = \frac{C}{n} + \mathcal{O}_p(\frac{1}{n}),$$

where C is a constant term. From the definition of the MLE,

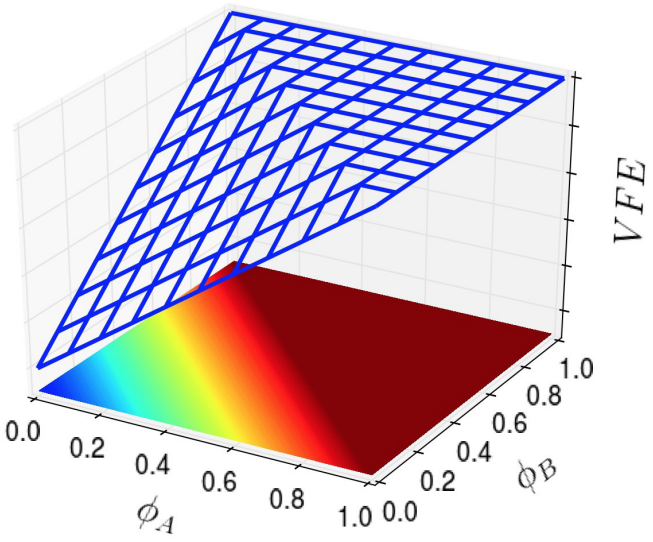
$$\frac{1}{n} \sum_{m=1}^{n} \log \frac{P(\boldsymbol{X}_m | \boldsymbol{X}^*)}{P(\boldsymbol{X}_m | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}})} \geq \frac{1}{n} \sum_{m=1}^{n} \log \frac{P(\boldsymbol{X}_m | \boldsymbol{X}^*)}{\tilde{P}(\boldsymbol{X}_m | \tilde{\boldsymbol{\mu}}_{ML})}$$

holds, then we can obtain the lower bound as follows.

$$\begin{aligned}
-\log P(\boldsymbol{X}^n | \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}) &\geq -\log P(\boldsymbol{X}^n | \boldsymbol{X}^*) + C + \mathcal{O}_p(1) \\
&= \mathcal{E} + \mathcal{O}_p(1).
\end{aligned}$$

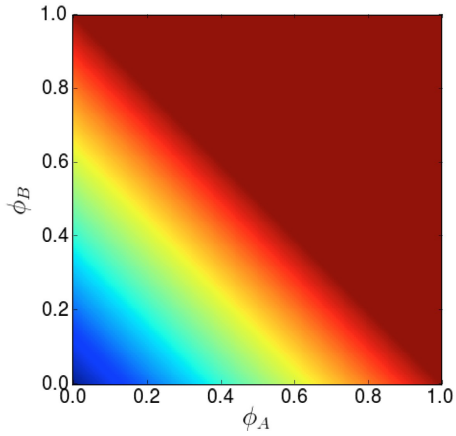


(a) $\hat{R}$

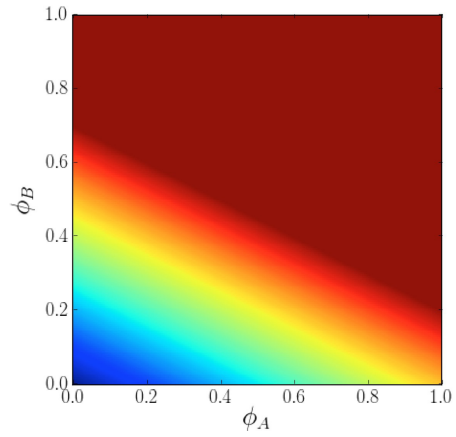


(b) VFE (λ_{vb})

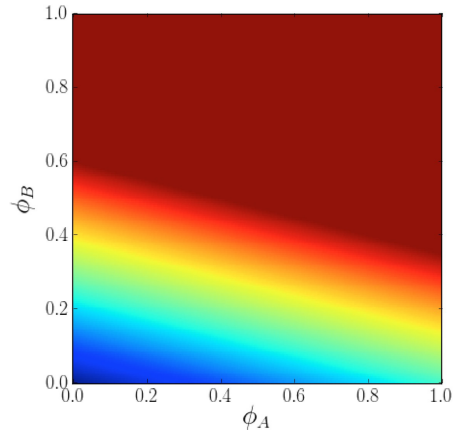
Figure 4.3: Visualization of the optimal number of factors, $\hat{R}$, and VFE (or equivalently, the main term of VFE, λ_{vb}) when $I = J = 5$.



(a) $I = 5, J = 5$

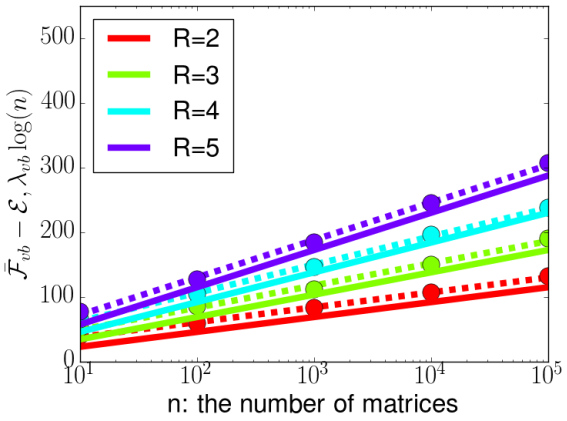


(b) $I = 5, J = 10$

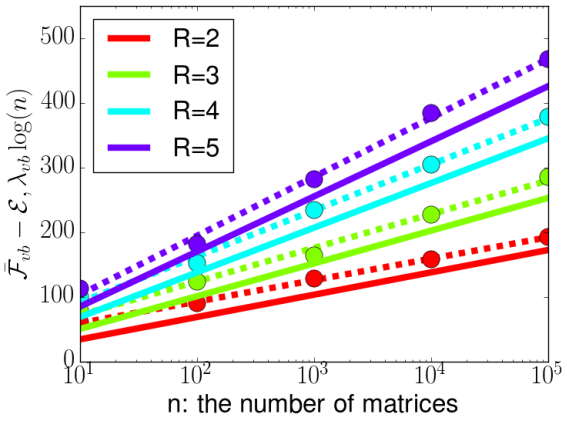


(c) $I = 5, J = 20$

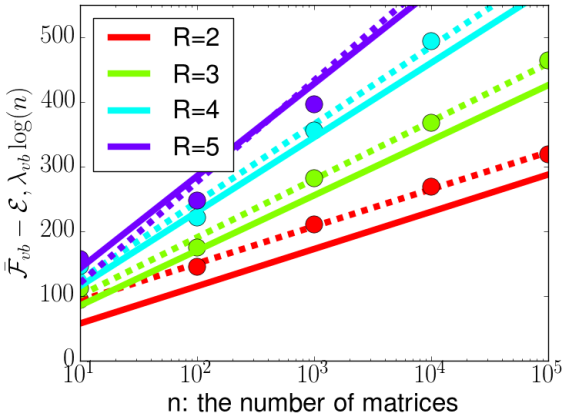
Figure 4.4: Visualization of VFE varying the size of input matrices.



(a) $I = 5, J = 5$

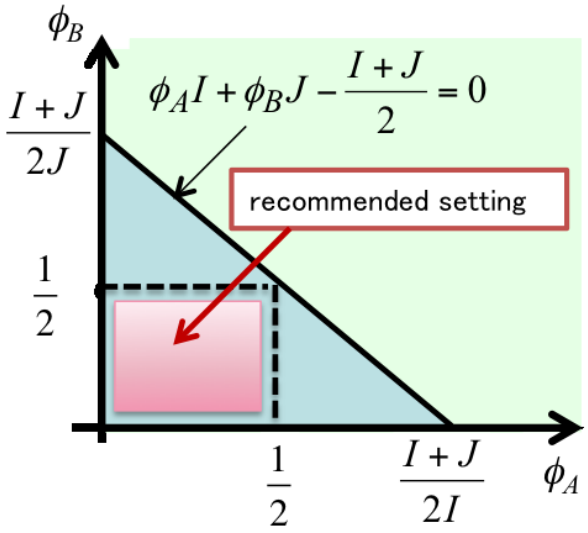


(b) $I = 5, J = 10$

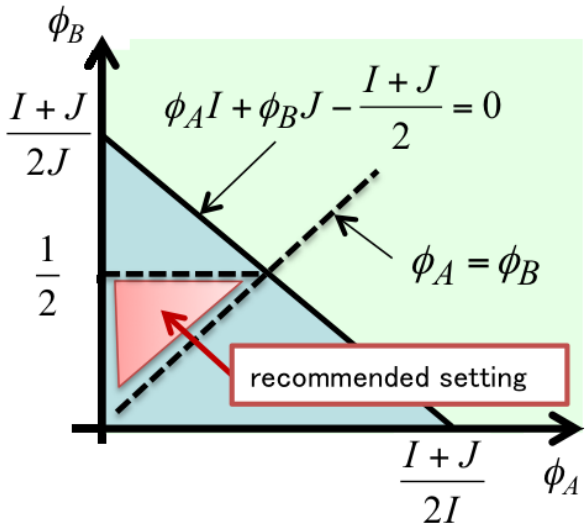


(c) $I = 5, J = 20$

Figure 4.5: Comparison of experimental and theoretical values of VFE.



(a) Example 1



(b) Example 2

Figure 4.6: Recommended hyperparameters for Example 1 and 2.

Chapter 5

Summary

This thesis provides our fruits of work on statistical learning theory of nonnegative matrix factorization (NMF) and multiple data analysis. The contributions of our works are summarized as follows:

- We proposed a new method for inconsistent resolution dataset analysis. By considering the data generative process using the latent high resolution matrix, we constructed a new probabilistic model based on NMF. We also derive majorization minimization (MM) algorithm for the model and provides the proof that the algorithm converges to (local) minima. Experimental results show that the effectiveness of the proposed method.
- We theoretically clarified the phase transition structure of Variational Bayesian NMF (VBNMF) through the asymptotic analysis of variational free enerygy (VFE). The numerical experiments support the validity of our analysis.

For the inconsistent resolution dataset analysis, the remaining research topics include further expansion of the model by, for example, introducing seasonality. It is also important to analyze how the difference of the grain sizes between the high and low resolution matrices affects the degree of performance improvement.

For the theoretical analysis, future work of this research includes an extension of our analysis to the case where some different factorization form such as tri-factorization [Cichocki et al., 2009] is adopted. Analysis of nonparametric (variational) Bayesian NMF [Hoffman et al., 2010] is also promising research direction.

Appendix

A Terminology of Statistical Learning

Let us define a (*statistical*) *model* and *a prior* distribution as $f(x|\theta)$ and $g(\theta;\phi)$, respectively. We denote *data* as $\boldsymbol{X}^n = \{x_i\}_{i=1}^n$, where n is the number of data and $x_i \in \mathbb{R}^{d_x}$. We assume that the each data $x_1, \cdots, x_n$ is independently identically distributed (i.i.d).

Based on the above definitions, the probability that data $\boldsymbol{X}^n$ is distributed from the model given a parameter θ, $P(X^n|\theta)$, can be written as

$$P(X^n|\theta) = \prod_{i=1}^{n} f(x_i|\theta). \tag{5.1}$$

This is called *likelihood*. The hyperparameter ϕ is usually set by manually. Then, we omit the notion of ϕ and denote *a prior* probability of a parameter as

$$P(\theta) = g(\theta;\phi). \tag{5.2}$$

The *a posterior* probability of a parameter given data, $P(\theta|X^n)$, is derived from Bayes rule:

$$P(\theta|X^n) = \frac{P(X^n|\theta)P(\theta)}{\int P(X^n|\theta)P(\theta)d\theta}. \tag{5.3}$$

Minus logarithm of the denominator of this equation is the *free energy*:

$$\mathcal{F} = -\log P(X^n) = -\log\Big(\int P(X^n|\theta)P(\theta)d\theta\Big). \tag{5.4}$$

A.1 Model

Here we show some examples of model f.

Example 1.1 (Gaussian): The parameter of (1-dimensional) Gaussian distribution consists of mean parameter $\mu \in \mathbb{R}$ and variance patemeter $\sigma^2 \in \mathbb{R}_+$, i.e., $\theta = (\mu, \sigma)$. The probability density function is defined as follows:

$$f_{\mathcal{N}}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big).$$

For ease of calculation, following precision parameter representation is often used.

$$f_{\mathcal{N}}(x|\mu,\tau^{-1}) = \sqrt{\frac{\tau}{2\pi}}\exp\Big(-\frac{\tau(x-\mu)^2}{2}\Big).$$

Example 2.1 (Poisson): Poisson distribution has the rate parameter $\lambda \in \mathbb{R}_+$ and is defined as

$$f_{\mathcal{PO}}(x|\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}.$$

It is known that the above Gaussian distribution and Poisson distribution belongs to *exponential family*, whose probability distribution is defined as

$$f_E(x|\eta) = h(x)\exp\Big(\eta \cdot T(x) - A(\eta)\Big), \tag{5.5}$$

where η is the natural parameter, $T(x)$ is *sufficient statistics* and $A(\eta)$ is the log-normalizer.

Let us confirm that Gaussian distribution belongs to exponential family. Since

$$f_{\mathcal{N}}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}}\exp\Big(\frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}x^2 - \frac{\mu^2}{2\sigma^2} - \log\sigma\Big),$$

setting $h(x) = 1/\sqrt{2\pi}, \eta = (\mu/\sigma^2, -1/2\sigma^2), T(x) = (x, x^2), A(\eta) = \mu^2/2\sigma^2 + \log\sigma = -\eta_1^2/4\eta_2 - 1/2\log(-2\eta_2)$ leads the form of Eq. (5.5). Similarly, Poisson distribution can be represented by the form by setting $h(x) = 1/x!, \eta = \log\lambda, T(x) = x, A(\eta) = \lambda = \exp(\eta)$ because

$$f_{\mathcal{PO}}(x|\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!} = \frac{1}{x!}\cdot\exp\big(\log\lambda \cdot x - \lambda).$$

Although exponential family can express broader family of probability distributions, it doesn't include many modern statistical models which has hierarchical structure such as mixture models. A representative mixture model is Gaussian mixture model (GMM).

Example 3.1 (Gaussian Mixture Model): The parameter of (1-dimensional) GMM consists of mixing ratio $\pi_k \in [0,1]$ ($\sum_k \pi_k = 1.0$), mean parameter $\mu_k \in \mathbb{R}$ and precision patemeter $\tau_k \in \mathbb{R}_+$, i.e., $\theta = (\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau}), \boldsymbol{\pi} = \{\pi_k\}_{k=1}^K, \boldsymbol{\mu} = \{\mu_k\}_{k=1}^K, \boldsymbol{\tau} = \{\tau_k\}_{k=1}^K)$ where K is the number of component. The probability density function is defined below:

$$f_{\mathrm{GMM}}(x|\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\tau}) = \sum\nolimits_{k=1}^K \pi_k f_{\mathcal{N}}(x|\mu_k,\tau_k^{-1}).$$

Note that NMF shown in this thesis is also one of the model by considering a input matrix $\boldsymbol{X} \in \mathbb{Z}_+^{I\times J}$ is IJ-dimensional vector, $x \in \mathbb{R}^{IJ}$. and factor matrices $\boldsymbol{A}, \boldsymbol{B}$ is the parameter.

A.2 Conjugate Prior

A prior distribution g is called *conjugate prior* of model f if the prior and the posterior probability (Eq.(5.3)) is the same family of distribution. This property enables us to obtain e.g., analytic form of predictive distribution explained later. We show two examples: conjugate prior of Poisson and Gaussian.

Example 2.2 (Prior and posterior of Poisson distribution model): As we stated in chapter 2, Gamma distribution is conjugate prior of Poisson distribution. We rewrite a definition of Gamma distribution below:

$$g_{\mathcal{G}}(\lambda|a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)}\lambda^{a_0-1}\exp\bigl(-b_0\lambda\bigr),$$

where a_0 and b_0 is the hyperparameters. We can easily check that a posterior distribution is also Gamma distribution. From Eq.(5.3),

$$P(\lambda|X^n) \propto f_{\mathcal{PO}}(X^n|\lambda)g_{\mathcal{G}}(\lambda|a_0, b_0) \propto \lambda^{a_0+\sum_i x_i-1}\exp\bigl\{-(b_0+n)\lambda\bigr\},$$

and then a posterior distribution is

$$P(\lambda|X^n) = g_{\mathcal{G}}(\lambda|a, b), \quad a = a_0 + \sum\nolimits_i x_i,\ b = b_0 + n. \tag{5.6}$$

Example 1.2 (Prior and posterior of Gaussian distribution model): For Gaussian distribution (with precision parameter representation), Gaussian-gamma distribution is the conjugate prior. Gaussian-gamma distribution is defined as the product of Gaussian and gamma distribution with hyperparameter μ_0, τ_0, a_0, b_0:

$$\begin{aligned}
&g_{\mathcal{NG}}(\mu, \tau|\mu_0, \tau_0, a_0, b_0)\\
&= \underbrace{\sqrt{\frac{\tau_0\tau}{2\pi}}\exp\Bigl(-\frac{\tau_0\tau}{2}(\mu_k-\mu_0)^2\Bigr)}_{\mathcal{N}(\mu_k|\mu_0,(\tau_0\tau)^{-1})}\underbrace{\frac{1}{\Gamma(a_0)}b_0^{a_0}\tau^{a_0-1}\exp(-b_0\tau)}_{\mathcal{G}(\tau|a_0,b_0)}\\
&\propto \exp\Bigl(-\tau\Bigl\{b_0+\frac{\tau_0}{2}(\mu-\mu_0)^2\Bigr\}\Bigr)\cdot\tau^{\frac{1}{2}+a_0-1}.
\end{aligned}$$

From Eq.(5.3),

$$\begin{aligned}
&P(\mu, \tau|X^n) \propto f_{\mathcal{G}}(X^n|\mu, \tau^{-1})g_{\mathcal{NG}}(\mu, \tau|\mu_0, \tau_0, a_0, b_0)\\
&\propto \tau^{\frac{n}{2}}\exp\Bigl(-\sum_i\frac{\tau(x_i-\mu)^2}{2}\Bigr)\exp\Bigl(-\tau\Bigl\{b_0+\frac{\tau_0}{2}(\mu-\mu_0)^2\Bigr\}\Bigr)\cdot\tau^{\frac{1}{2}+a_0-1}\\
&\propto \exp\Bigl(-\tau\Bigl\{b_0+\frac{1}{2}\sum_i x_i^2+\frac{\tau_0+n}{2}\mu^2-(\tau_0\mu_0+\sum_i x_i)\mu+\frac{\tau_0}{2}\mu_0^2\Bigr\}\Bigr)\cdot\tau^{\frac{1}{2}+a_0+\frac{n}{2}-1}\\
&\propto \exp\Bigl(-\tau\Bigl\{b_0+\frac{1}{2}\sum_i x_i^2-\frac{\tau_0+n}{2}\bar{\mu}^2+\frac{\tau_0}{2}\mu_0^2+\frac{\tau_0+n}{2}\bigl(\mu-\bar{\mu}\bigr)^2\Bigr\}\Bigr)\cdot\tau^{\frac{1}{2}+a_0+\frac{n}{2}-1}
\end{aligned}$$

and then a posterior distribution is also Gaussian-gamma distribution.

$$P(\mu,\tau|X^n) = g_{\mathcal{NG}}(\mu,\tau|\bar{\mu},\bar{\tau},a,b), \quad \bar{\mu} = \frac{\tau_0\mu_0 + \sum_i x_i}{\tau_0 + n}, \tag{5.7}$$
$$\bar{\tau} = \tau_0 + n,\ a = a_0 + \frac{n}{2},\ b = b_0 + \frac{1}{2}\Big(\sum_i x_i^2 - n\bar{\mu}^2\Big) + \frac{\tau_0}{2}(\mu_0^2 - \bar{\mu}^2).$$

More generally, if a model f belongs to exponential family, its conjugate prior exists and is given by

$$g_E(\eta|\xi_0,\nu_0) = \mathcal{Z}(\xi_0,\nu_0)\exp\Big(\eta\cdot\xi_0 - \nu_0 A(\eta)\Big),$$

where ξ_0 and ν_0 are hyperparameters and $\mathcal{Z}(\xi_0,\nu_0)$ is the normalizer. Since

$$P(\eta|X^n) \propto f_E(X^n|\eta)g_E(\eta) \propto \exp\Big\{\eta\cdot\big(\xi_0 + \sum\nolimits_{i=1}^n T(x_i)\big) - (\nu_0+n)A(\eta)\Big\},$$

and

$$P(\eta|X^n) = g_E\Big(\eta\Big|\xi_0 + \sum\nolimits_{i=1}^n T(x_i), \nu_0 + n\Big),$$

it is confirmed that g_E is the conjugate prior of f_E.

A.3 Point Estimation

This subsection provides a approach called point estimation. Purpose of point estimation is to find an optimum point of parameter. There are two representative methods, maximum likelihood estimation and maximum a posterior estimation. In maximum likelihood estimation, a target point of parameter is the one which maximizes likelihood. In maximum a posterior estimation, a target point of parameter is the one which maximizes posterior probability. Estimated parameters by the methods are called maximum likelihood estimator (MLE) or maximum a posterior estimator (MAP). By substituting an MLE/MAP for a parameter of a model, *predictive distribution*, which is used for predicting the distribution of a new data, is constructed.

Given a model, a prior and data, the maximum likelihood estimator (MLE), θ_{MLE}, the maximum a posterior estimator (MAP), θ_{MAP}, are formally defined as follows:

$$\theta_{MLE} = \arg\max_\theta \mathcal{L}_{MLE}(\theta),$$
$$\theta_{MAP} = \arg\max_\theta \mathcal{L}_{MAP}(\theta),$$

where $\mathcal{L}_{MLE}(\theta)$ and $\mathcal{L}_{MAP}(\theta)$ are the following log-likelihood and log-posterior probability,

$$\mathcal{L}_{MLE}(\theta) = \log P(X^n|\theta) = \sum\nolimits_{i=1}^n \log f(x_i|\theta),$$
$$\mathcal{L}_{MAP}(\theta) = \log P(\theta|X^n) = \sum\nolimits_{i=1}^n \log f(x_i|\theta) + \log g(\theta) + Const.$$

We show two examples: MLE and MAP of Gaussian and that of Poisson.

Example 1.3 (MLE and MAP of Gaussian distribution model): Using Eq. (5.7),

$$\begin{aligned}\mathcal{L}_{MAP}(\mu,\tau) =&\Big(\frac{1}{2}+a_0+\frac{n}{2}-1\Big)\log\tau \\ &-\tau\Big\{b_0+\frac{1}{2}\sum_i x_i^2-\frac{\tau_0+n}{2}\bar{\mu}^2+\frac{\tau_0}{2}\mu_0^2+\frac{\tau_0+n}{2}\big(\mu-\bar{\mu}\big)^2\Big\}\end{aligned}$$

Solving $\frac{\partial\mathcal{L}_{MAP}(\mu,\tau)}{\partial\mu}=0$ and $\frac{\partial\mathcal{L}_{MAP}(\mu,\tau)}{\partial\tau}=0$, MAP is given by

$$\mu_{MAP}=\bar{\mu},\ \tau_{MAP}=\frac{\frac{1}{2}+a_0+\frac{n}{2}-1}{b_0+\frac{1}{2}\sum_i x_i^2-\frac{\tau_0+n}{2}\bar{\mu}^2+\frac{\tau_0}{2}\mu_0^2}.$$

Similarly, MLE can be derived as follows:

$$\mu_{MLE}=\frac{1}{n}\sum_i x_i,\ \sigma^2_{MLE}=\frac{1}{n}\sum_i(x_i-\mu_{MLE})^2.$$

Example 2.3 (MLE and MAP of Poisson distribution model): Using Eq. (5.6),

$$\mathcal{L}_{MAP}(\lambda)=\log P(\lambda|\boldsymbol{X}^n)=\Big(a_0+\sum_i x_i-1\Big)\log\lambda-(b_0+n)\lambda.$$

Solving $\frac{\partial\mathcal{L}_{MAP}(\lambda)}{\partial\lambda}=0$, MAP is

$$\lambda_{MAP}=\frac{a_0+\sum_i x_i-1}{b_0+n}.$$

Similarly, MLE is given by

$$\lambda_{MLE}=\frac{\sum_i x_i}{n}.$$

We confirmed that MLE and MAP for Gaussian and Poisson distribution has analytic form. In general, MLE and MAP of a model which belongs to exponential family is computed using sufficient statistics. This can be confirmed that the log-posterior of exponential family is given by

$$\mathcal{L}^{MAP}(\eta)=\eta\cdot\Big(\xi_0+\sum\nolimits_i T(x_i)\Big)-(\nu_0+n)A(\eta),$$

and that solving $\frac{\partial\mathcal{L}^{MAP}(\eta)}{\partial\eta}=0$ leads

$$\tilde{\mu}_{MAP}=\frac{\xi_0+\sum_i T(x_i)}{\nu_0+n},$$

where $\tilde{\mu}$ is the mean value parameter defined as $\tilde{\mu}=\mathbb{E}_{f_E(x|\eta)}[T(x)](=\frac{\partial A(\eta)}{\partial\eta})$. Similarly, MLE is given by

$$\tilde{\mu}_{MAP}=\frac{\sum_i T(x_i)}{n}.$$

As we stated, many modern statistical models such as GMM are not exponential family and don't have analytic form of MLE/MAP. For such models, an iterative algorithms such as Expectation Maximization (EM) algorithm [Dempster et al., 1977] is frequently used. The key of EM algorithm is introducing latent variable and *complete* likelihood, which indicates the joint probability of data and latent variable. In the case of GMM, latent variable $Z^n = \{z_{ik}\}_{i,k=1}^{n,K}$ is used,where $z_{ik} = \{0, 1\}$ represents whether i-th data belongs to k-th component or not. A complete likelihood is defined as follows:

$$P(X^n, Z^n|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \prod_{i=1}^{n}\prod_{k=1}^{K}\left\{\pi_k f_{\mathcal{N}}(x_i|\mu_k, \tau_k^{-1})\right\}^{z_{ik}}.$$

Note that $\sum_{Z^n} P(X^n, Z^n|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \prod_{i=1}^{n} f_{\mathrm{GMM}}(x_i|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau})$. For more details, see e.g., [Bishop, 2006].

Predictive distribution of point estimation is constructed by substituting MLE or MAP, i.e.,

$$\begin{aligned} P_{MLE}(x|X^n) &= f(x|\theta_{MLE}), \\ P_{MAP}(x|X^n) &= f(x|\theta_{MAP}). \end{aligned} \tag{5.8}$$

A.4 Bayesian Estimation

This subsection provides a approach called Bayesian estimation. In Bayesian estimation, a posterior probability of parameter (which is derived by Bayes rule! See Eq. (5.3)) has a central role. For example, (Bayesian) predictive distribution is constructed by taking the expectation of a model w.r.t. a posterior probability of parameter:

$$P_{Bayes}(x|X^n) = \int f(x|\theta)P(\theta|X^n)d\theta. \tag{5.9}$$

Unlike the predictive distribution in point estimation (Eq. (5.8)), this predictive distribution is not same family of model and that of prior in general. Let us show examples.

Example 2.4 (Bayesian predictive distribution of Poisson distribution model): Bayesian predictive distribution of Poisson model is given by negative binomial

distribution. Using Eq. (5.6)(5.9)

$$\begin{aligned} P_{Bayes}(x|X^n) &= \int f_{\mathcal{PO}}(x|\lambda) g_{\mathcal{G}}(\lambda|a,b) d\lambda \\ &= \frac{1}{x!}\frac{b^a}{\Gamma(a)} \int \lambda^{a+x-1} \exp\bigl\{-(b+1)\lambda\bigr\} d\lambda \\ &= \frac{1}{x!}\frac{b^a}{\Gamma(a)} \frac{\Gamma(a+x)}{(b+1)^{a+x}} \\ &= \frac{\Gamma(a+x)}{x!\Gamma(a))} \Big(\frac{1}{b+1}\Big)^x \Big(\frac{b}{b+1}\Big)^a \\ &= \mathrm{NB}\left(x\Big|a, \frac{1}{b+1}\right), \end{aligned}$$

where NB is the negative binomial distribution:

$$\mathrm{NB}(k|r,p) = \frac{\Gamma(k+r)}{k!\Gamma(r)} p^k (1-p)^r.$$

Example 1.4 (Bayesian predictive distribution of Gaussian distribution model): Following the derivation by Murphy [Murphy, 2007],

$$\begin{aligned} P_{Bayes}(x|X^n) &= \int f_{\mathcal{N}}(x|\mu,\tau) g_{\mathcal{NG}}(\mu,\tau|\bar{\mu},\bar{\tau},a,b) d\lambda \\ &= \mathrm{St}\left(x\Big|\bar{\mu}_k, \big(\frac{a\bar{\tau}}{b_k(\bar{\tau}+1)}\big)^{-1}, 2a\right), \end{aligned}$$

where St is the Student-t's distribution:

$$\mathrm{St}(x|\mu,\sigma^2,\nu) = \frac{\Gamma(\nu/2+1/2)}{\Gamma(\nu/2)} \frac{1}{\sigma\sqrt{\pi\nu}} \left[1 + \frac{1}{\nu}\Big(\frac{x-\mu}{\sigma}\Big)^2\right]^{-\nu/2-1/2}.$$

We discuss a bit more detail on student-t's distribution. As shown in Fig. 5.1, student-t has a long tail in comparison to Gaussian distribution and converges to Gaussian in the limit $\nu \to \infty$. Therefore, when the number of data n is small and $2a$ is small, the distribution has a long tail reflecting parameter uncertainty, and converges to Gaussian as n goes to infinity.

More generally, the predictive distribution of exponential family is represented by following form:

$$P_{Bayes}(x|X^n) = \int f_E(x|\eta) g_E(\eta|\xi,\nu) d\eta = \frac{\mathcal{Z}(\xi,\nu)}{\mathcal{Z}(\xi+T(x),\nu+1)} h(x). \quad (5.10)$$

We confirmed that the Bayesian prediction of Gaussian and Poisson is represented by known, analytic distributions. The other examples of models whose a posterior and predictive distribution have analytic form can be found in e.g. [Bernardo and Smith, 2009]. However, for many modern statistical model, both a posterior distribution and predictive distribution don't have such analytical form and it is necessary to use some sampling method such as markov chain

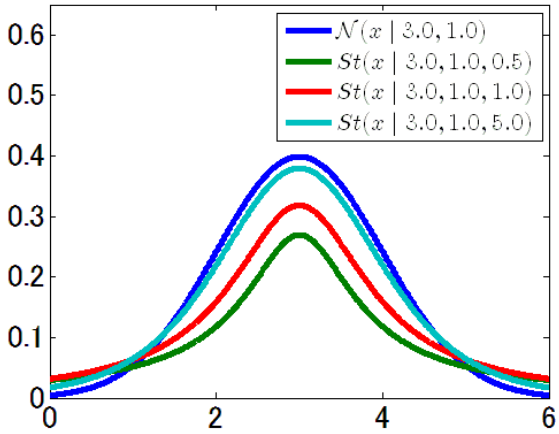


Figure 5.1: Comparison of Gaussian and Student-t.

monte carlo (MCMC) [Andrieu et al., 2003]. By using sampled parameters from a posterior distribution, predictive distribution is numerically constructed. It is known that MCMC usually requires high computational cost and long running time. Then, the use of extended variant of MCMC such as exchange monte carlo [Hukushima and Nemoto, 1996] is also investigated for Bayesian estimation [Nagata and Watanabe, 2008].

A.5 Variational Bayesian Estimation

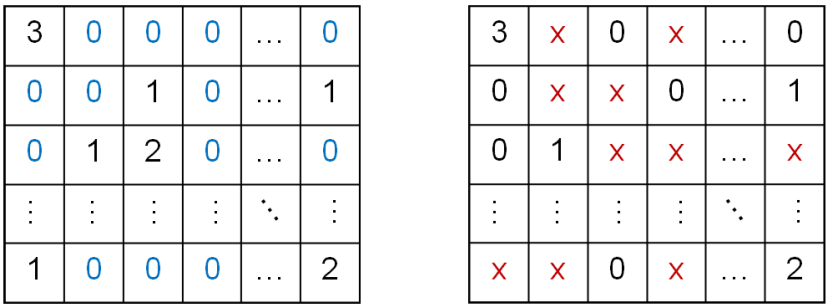
This subsection provides a approach called variational Bayesian (VB) estimation. As we stated in previous subsection, for many modern statistical model, a posterior distribution is not represented by analytic distribution. Then, VB is used to estimate the variational distribution, which approximates a posterior distribution of parameters and hidden variables [Attias, 1999, Attias, 2000, Jordan et al., 1999]. For deriving tractable algorithm, it is needed to adopt some restricted family of distribution. Then, as we have done in Eq. (2.2.2) for NMF, a factorized distribution, where some parameters and latent variables are independent, is adopted.

Here we show the example of GMM. following factorized distribution often used:

$$q(Z^n, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = q(Z^n)q(\boldsymbol{\pi})q(\boldsymbol{\mu}, \boldsymbol{\tau}).$$

Estimation is done by minimizing following functional,

$$\bar{\mathcal{F}}[q] = \mathbb{E}_{q(Z^n)q(\boldsymbol{\pi})q(\boldsymbol{\mu},\boldsymbol{\tau})}\Big[\log \frac{q(Z^n)q(\boldsymbol{\pi})q(\boldsymbol{\mu}, \boldsymbol{\tau})}{P(X^n, Z^n|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau})P(\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\tau})}\Big],$$



(a) (b)

Figure 5.2: Examples of matrix with (a) many zero elements (blue) and with (b) many missing elements (red).

and predictive distribution is constructed as follows.

$$P_{VB}(x|X^n) = \int f(x|\theta)q(\theta)d\theta.$$

For more details, see e.g., [Bishop, 2006].

B Practical Implementation of NMF

This section describes a practical implementation of NMF for large size matrix.

Data which are analyzed by recent data analysis are often represented by a matrix whose size (the number of rows and columns) are very large, and thus naive implementation may take a long time for estimation. Therefore, it is necessary to use practical implementations where some property of such large matrix are taken into account.

There are two representative properties that a large matrix holds: (i) existence of many zero elements and (ii) existence of many missing elements, as shown in Fig. 5.2. The matrix which has either or both property is often called *sparse* matrix. A example of (i) is purchase logs which are represented by a matrix whose size corresponds to the number of users and items. Each element indicates purchase amount of an item by a user. Although the size may be large like more than ten thousand, the variety of items which are bought by a single users is not so many like less than 1% of the total number of items. Therefore, there are many zero elements in the matrix. A example of (ii) is movie rating logs which are represented by a matrix whose size corresponds to the number of users and movies. Each element indicates a rating of a movie by a user. Since the variety of movies watched and rated by a single user is very limited, there are many missing elements, which indicate the corresponding ratings are not given. These two properties can be incorporated for practical implementation.

B.1 Handling Zero Elements

Here we show how the "many zero" property can be incorporated into the algorithm implementation. For the simplicity, we consider the case that number of input matrices $n = 1$ and then input matrix is $\boldsymbol{X} = \{x_{ij}\}_{i,j=1}^{I,J}$.

Let us define the set of indexes of which elements are non-zero values as Ω^{nz}, i.e., $x_{ij} \neq 0$ if $(i,j) \in \Omega^{nz}$. Moreover, we denote Ω_i^{nz} and Ω_j^{nz} the set of non-zero columns of i-th row and that of non-zero rows of j-th column. From this definition, objective function of NMF (Eq. (2.3)) can be written as follows.

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) &= \sum\nolimits_{i,j=1}^{I,J} \left\{ \hat{x}_{ij} - x_{ij} \log \hat{x}_{ij} \right\} \\ &= \sum\nolimits_{i,j=1}^{I,J} \hat{x}_{ij} - \sum\nolimits_{(i,j)\in\Omega_{nz}} x_{ij} \log \hat{x}_{ij}.\end{aligned}$$

The second term of the above final equation contains only summation over non-zero elements. Following the derivation of MM algorithm, the update equations are given as follows:

$$\begin{aligned}a_{ir} &\leftarrow a_{ir} \frac{\sum_{j\in\Omega_i^{nz}} \frac{x_{ij}}{\hat{x}_{ij}} b_{jr}}{\sum_{j=1}^{J} b_{jr}}, \\ b_{jr} &\leftarrow b_{jr} \frac{\sum_{i\in\Omega_j^{nz}} \frac{x_{ij}}{\hat{x}_{ij}} a_{ir}}{\sum_{i=1}^{I} a_{ir}}.\end{aligned}$$

Since we can skip the zero elements in the computation, computational cost of the each step of the algorithm is $O(LR)$, where L is the total number of non-zero elements in $\boldsymbol{X}$.

B.2 Handling Missing Elements

Here we show how the "many missing" property can be incorporated into the algorithm implementation. Similar to the previous subsection, we consider the case that number of input matrices $n = 1$ and then input matrix is $\boldsymbol{X} = \{x_{ij}\}_{i,j=1}^{I,J}$ for simplicity.

Let us define the set of indexes of which elements are observed (non-missing) as Ω^o. Moreover, we denote Ω_i^o and Ω_j^o the set of non-missing columns of i-th row and that of non-zero rows of j-th column. When the missing elements exist, objective function of NMF can be designed by ignoring missing elements:

$$\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \sum\nolimits_{(i,j)\in\Omega^o} \left\{ \hat{x}_{ij} - x_{ij} \log \hat{x}_{ij} \right\}$$

Note that the summation is taken only over non-missing elements. Following

the derivation of MM algorithm,

$$a_{ir} \leftarrow a_{ir} \frac{\sum_{j \in \Omega_i^o} \frac{x_{ij}}{\bar{x}_{ij}} b_{jr}}{\sum_{j \in \Omega_i^o} b_{jr}},$$
$$b_{jr} \leftarrow b_{jr} \frac{\sum_{i \in \Omega_j^o} \frac{x_{ij}}{\bar{x}_{ij}} a_{ir}}{\sum_{i \in \Omega_j^o} a_{ir}}.$$

Since we can skip the missing elements in the computation, computational cost of the each step of the algorithm is $O(LR)$, where L is the total number of observed elements in $\boldsymbol{X}$.

We showed that the computation cost depends on the number of non-zero/missing elements. If it is still large, use of distributed processing may be promising way for faster estimation [Liu et al., 2010, Yin et al., 2014]. Online algorithm [Cao et al., 2007, Guan et al., 2012] is also good choice for estimation with limited memory.

C MAP Estimation of NMF

Majorization Minimization (MM) algorithm provided in § 2.2.1 is designed for MLE. The MM algorithm can be used for MAP.

We consider the setting the conjugate gamma priors are employed, analogous to chapter 2.

$$P(\boldsymbol{A}) = \prod_{i,r=1}^{I,R} \mathcal{G}(a_{ir}|\phi_A, \eta_A/\phi_A),\ P(\boldsymbol{B}) = \prod_{j,r=1}^{J,R} \mathcal{G}(b_{jr}|\phi_B, \eta_B/\phi_B),$$

where ϕ_A, η_A, ϕ_B, and η_B are hyperparameters.

Using the negative log-likelihood (Eq. (2.2)) and the above prior, the objective function $\mathcal{L}_{MAP}$ can be written as follows:

$$\begin{aligned}\mathcal{L}_{MAP}(\boldsymbol{A}, \boldsymbol{B}) =& n \sum_{i,j=1}^{I,J} \left\{ \hat{x}_{ij} - \bar{x}_{ij} \log \hat{x}_{ij} \right\} \\ &+ \sum_{i,r=1}^{I,R} \left\{ (\phi_A - 1) \log a_{ir} - \frac{\phi_A}{\eta_A} a_{ir} \right\} \\ &+ \sum_{j,r=1}^{J,R} \left\{ (\phi_B - 1) \log b_{jr} - \frac{\phi_B}{\eta_B} b_{jr} \right\}.\end{aligned} \tag{5.11}$$

Let us also define the auxiliary (majorizing) function $\mathcal{L}^+_{MAP}$ as

$$\begin{aligned}\mathcal{L}^+_{MAP}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T}) =& n \sum_{i,j=1}^{I,J} \left\{ \hat{x}_{ij} - \bar{x}_{ij} \sum\nolimits_r t_{ijr} \log \left(\frac{a_{ir} b_{jr}}{t_{ijr}} \right) \right\} \\ &- \sum_{i,r=1}^{I,R} \left\{ (\phi_A - 1) \log a_{ir} - \frac{\phi_A}{\eta_A} a_{ir} \right\} \\ &- \sum_{j,r=1}^{J,R} \left\{ (\phi_B - 1) \log b_{jr} - \frac{\phi_B}{\eta_B} b_{jr} \right\}.\end{aligned}$$

where $\boldsymbol{T} = \{t_{ijr}\}$ is auxiliary variables satisfying $\sum_r t_{ijr} = 1$ $(\forall (i,j))$. It can be verified that the auxiliary function $\mathcal{L}^+$ has following two properties:

1. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) \leq \mathcal{L}^{+}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$
2. $\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}) = \min_{\boldsymbol{T}} \mathcal{L}^{+}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T})$.

Note that the equality holds if and only if

$$t_{ijr} = \frac{a_{ir} b_{jr}}{\sum_{r'} a_{ir'} b_{jr'}}. \tag{5.12}$$

Since the partial derivative of $\mathcal{L}^{+}_{MAP}$ is computed as

$$\begin{aligned}
\frac{\partial}{\partial a_{ir}} \mathcal{L}^{+}_{MAP}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T}) =& n \sum_{j=1}^{J} b_{jr} - n \sum_{j=1}^{J} \bar{x}_{ij} t_{ijr} \frac{t_{ijr}}{a_{ir} b_{jr}} \frac{b_{jr}}{t_{ijr}} - \frac{\phi_A - 1}{a_{ir}} + \frac{\phi_A}{\eta_A} \\
=& n \sum_{j=1}^{J} b_{jr} - n \sum_{j=1}^{J} \frac{\bar{x}_{ij} t_{ijr}}{a_{ir}} - \frac{\phi_A - 1}{a_{ir}} + \frac{\phi_A}{\eta_A} \\
=& n \sum_{j=1}^{J} b_{jr} + \frac{\phi_A}{\eta_A} - \frac{n \sum_{j=1}^{J} \bar{x}_{ij} t_{ijr} + \phi_A - 1}{a_{ir}},
\end{aligned}$$

setting $\frac{\partial}{\partial a_{ir}} \mathcal{L}^{+}_{MAP}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{T}) = 0$ leads

$$a_{ir} = \frac{n \sum_{j=1}^{J} \bar{x}_{ij} t_{ijr} + \phi_A - 1}{n \sum_{j=1}^{J} b_{jr} + \frac{\phi_A}{\eta_A}}. \tag{5.13}$$

By substituting Eq. (5.12) into Eq. (5.13), following update rules for MAP is obtained.

$$a_{ir} = \frac{a_{ir} \sum_{j=1}^{J} \frac{\bar{x}_{ij}}{\tilde{x}_{ij}} b_{jr} + \frac{\phi_A - 1}{n}}{\sum_{j=1}^{J} b_{jr} + \frac{1}{n} \frac{\phi_A}{\eta_A}} \tag{5.14}$$

$$b_{jr} = \frac{b_{jr} \sum_{i=1}^{I} \frac{\bar{x}_{ij}}{\tilde{x}_{ij}} a_{ir} + \frac{\phi_B - 1}{n}}{\sum_{i=1}^{I} a_{ir} + \frac{1}{n} \frac{\phi_B}{\eta_B}} \tag{5.15}$$

Algorithm 5 shows a pseudo code. As the number of input matrices $n \to \infty$, the above update rules are equivalent to that for MLE (Eq.(2.6)(2.7)).

Algorithm 5 Majorization Minimization (MM) for MAP of NMF

Input: $\boldsymbol{X}^n$: input matrices, R: the number of factors, $\phi_A, \eta_A, \phi_B, \eta_B$: hyper-parameters

Output: $\boldsymbol{A}, \boldsymbol{B}$ such that objective function in Eq. (5.11) is minimized under the non-negative constraint.

1: initialization for $\boldsymbol{A}$ and $\boldsymbol{B}$
2: **repeat**
3: Update $\boldsymbol{A}$ by Eq. (5.14)
4: Update $\boldsymbol{B}$ by Eq. (5.15)
5: **until** Converge

List of Symbols

Symbol	Description
I	the number of rows
J	the number of columns
K	the number of rows
R	the number of factors
$\boldsymbol{X}$	$I \times J$ input matrix
$\boldsymbol{X}^n$	set of input matrices, $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_n$
$\bar{\boldsymbol{X}}$	average of input matrices, $\frac{1}{n}\sum_m \boldsymbol{X}_m$
$\boldsymbol{Y}$	$K \times J$ input matrix
$\boldsymbol{Y}^m$	set of input matrices, $\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_m$
$\bar{\boldsymbol{Y}}$	average of input matrices, $\frac{1}{m}\sum_\ell \boldsymbol{Y}_\ell$
$\boldsymbol{A}$	factor matrix
$\boldsymbol{B}$	factor matrix
$\boldsymbol{C}$	factor matrix
$\boldsymbol{S}$	latent/auxiliary variable
$\boldsymbol{T}$	latent/auxiliary variable
$\mathcal{L}$	objective function to be minimized using MM
$\bar{\mathcal{F}}$	objective functional to be minimized using VB
$\bar{\mathcal{F}}_{vb}$	Variational free energy (VFE)
$\mathcal{E}$	Empirical entropy.
R^*	Nonnegative rank of true matrix $\boldsymbol{X}^*$
R^*_{vb}	Optimal number of factors that minimize VFE

List of Figures

List of Tables

List of Publications

Journal Articles

1. Kohjima, M., Matsubayashi, T., and Sawada, H. (2017). Probabilistic models based on non-negative matrix factorization for inconsistent resolution dataset analysis. *The IEICE Transactions on Information and Systems (Japanese Edition)*, 100(4):520–529 (This paper was selected for IEICE 2017 Best Paper Award)

Survey Articles

1. Kohjima, M., Matsubayashi, T., and Sawada, H. (2016). Multiple data analysis and non-negative matrix/tensor factorizaiton [I]: Multiple data analysis and its advances. *The Journal of the Institute of Electronics, Information and Communication Engineers*, 99(6)

2. Matsubayashi, T., Kohjima, M., and Sawada, H. (2016). Multiple data analysis and non-negative matrix/tensor factorizaiton [II・finish]: Tensor data analysis and applications. *The Journal of the Institute of Electronics, Information and Communication Engineers*, 99(7)

International Conference Proceedings

1. Kohjima, M., Matsubayashi, T., and Sawada, H. (2015). Probabilistic non-negative inconsistent-resolution matrices factorization. In *24th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1855–1858

2. Kohjima, M. and Watanabe, S. (2017). Phase transition structure of variational bayesian nonnegative matrix factorization. In *International Conference on Artificial Neural Networks (ICANN)*, pages 146–154. Springer (This paper was selected for ICANN 2017 Best Paper Award)

Acknowledgement

This work has been carried out at Watanabe Laboratory, Department of Mathematical and Computing Science, Tokyo Institute of Technology. I would like to express my gratitude to supervisor Professor Sumio Watanabe for his help, support and encouragement. I learned the importance of probing the true nature of a thing from Prof. Watanabe. I would like to thank to Dr. Hiroshi Sawada and Dr. Tatsushi Matubayashi for fruitful comments on the study at NTT Service Evolution Laboratories. I learned the fun of researching at company from Dr. Sawada and Dr. Matubayashi. I am grateful to Prof. Yumiharu Nakano, Prof. Makoto Yamashita, Prof. Takafumi Kanamori, and Prof. Yoshiyuki Kabashima for reviewing this thesis and providing comments on it. I am also grateful to former and present members of Watanabe Lab. and Proactive Navigation Project at NTT Service Evolution Labs. for lively discussion with them. Finally, I appreciate to my family for their warm encouragement.

Masahiro Kohjima

Bibliography

[Aimoto and Kashima, 2013] Aimoto, Y. and Kashima, H. (2013). Matrix factorization with aggregated observations. In *Advances in Knowledge Discovery and Data Mining*, pages 521–532. Springer.

[Akaike, 1974] Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.

[Alzer, 1997] Alzer, H. (1997). On some inequalities for the gamma and psi functions. *Mathematics of Computation of the American Mathematical Society*, 66(217):373–389.

[Andrieu et al., 2003] Andrieu, C., De Freitas, N., Doucet, A., and Jordan, M. I. (2003). An introduction to mcmc for machine learning. *Machine learning*, 50(1-2):5–43.

[Aoyagi, 2010] Aoyagi, M. (2010). Stochastic complexity and generalization error of a restricted boltzmann machine in bayesian estimation. *Journal of Machine Learning Research*, 11(Apr):1243–1272.

[Aoyagi and Watanabe, 2005] Aoyagi, M. and Watanabe, S. (2005). Stochastic complexities of reduced rank regression in bayesian estimation. *Neural Networks*, 18(7):924–933.

[Attias, 1999] Attias, H. (1999). Inferring parameters and structure of latent variable models by variational bayes. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 21–30. Morgan Kaufmann Publishers Inc.

[Attias, 2000] Attias, H. (2000). A variational baysian framework for graphical models. In *Advances in neural information processing systems*, pages 209–215.

[Bernardo and Smith, 2009] Bernardo, J. M. and Smith, A. F. (2009). *Bayesian Theory*, volume 405. John Wiley & Sons.

[Berry and Browne, 2005] Berry, M. W. and Browne, M. (2005). Email surveillance using non-negative matrix factorization. *Computational & Mathematical Organization Theory*, 11(3):249–264.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

[Brouwer et al., 2017] Brouwer, T., Frellsen, J., and Lió, P. (2017). Comparative study of inference methods for bayesian nonnegative matrix factorisation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 513–529. Springer.

[Cao et al., 2007] Cao, B., Shen, D., Sun, J.-T., Wang, X., Yang, Q., and Chen, Z. (2007). Detect and track latent factors with online nonnegative matrix factorization. In *IJCAI*, volume 7, pages 2689–2694.

[Cemgil, 2009] Cemgil, A. T. (2009). Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*.

[Chi et al., 2007] Chi, Y., Zhu, S., Song, X., Tatemura, J., and Tseng, B. L. (2007). Structural and temporal analysis of the blogosphere through community factorization. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 163–172. ACM.

[Cichocki et al., 2009] Cichocki, A., Zdunek, R., Phan, A. H., and Amari, S. (2009). *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons.

[Cohen and Rothblum, 1993] Cohen, J. E. and Rothblum, U. G. (1993). Nonnegative ranks, decompositions, and factorizations of nonnegative matrices. *LINEAR ALGEBRA AND ITS APPLICATIONS*, 190:149–168.

[Davenport and Harris, 2007] Davenport, T. H. and Harris, J. G. (2007). *Competing on analytics: The new science of winning*. Harvard Business Press.

[Davenport et al., 2010] Davenport, T. H., Harris, J. G., and Morison, R. (2010). *Analytics at work: Smarter decisions, better results*. Harvard Business Press.

[De Leeuw, 1994] De Leeuw, J. (1994). Block-relaxation algorithms in statistics. In *Information systems and data analysis*, pages 308–324. Springer.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38.

[Deng et al., 2016] Deng, D., Shahabi, C., Demiryurek, U., Zhu, L., Yu, R., and Liu, Y. (2016). Latent space model for road networks to predict time-varying traffic. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1525–1534.

[Ding et al., 2008] Ding, C., Li, T., and Peng, W. (2008). On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Computational Statistics & Data Analysis*, 52(8):3913–3927.

[Endo et al., 2015] Endo, Y., Toda, H., and Koike, Y. (2015). What's hot in the theme: Query dependent emerging topic extraction from social streams. In *Proceedings of the 24th International Conference on World Wide Web*, pages 31–32. ACM.

[Févotte et al., 2009] Févotte, C., Bertin, N., and Durrieu, J. L. (2009). Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural computation*, 21(3):793–830.

[Frey and Osborne, 2017] Frey, C. B. and Osborne, M. A. (2017). The future of employment: how susceptible are jobs to computerisation? *Technological forecasting and social change*, 114:254–280.

[Guan et al., 2012] Guan, N., Tao, D., Luo, Z., and Yuan, B. (2012). Online nonnegative matrix factorization with robust stochastic approximation. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7):1087–1099.

[Hoffman et al., 2010] Hoffman, M., Blei, D. M., and Cook, P. (2010). Bayesian nonparametric matrix factorization for recorded music. In *Proceedings of the 27th International Conference on Machine Learning*, pages 439–446.

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

[Hosino et al., 2005] Hosino, T., Watanabe, K., and Watanabe, S. (2005). Stochastic complexity of variational bayesian hidden markov models. In *Proceedings of International Joint Conference on Neural Networks*, volume 2, pages 1114–1119. IEEE.

[Hoyer, 2004] Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(Nov):1457–1469.

[Hukushima and Nemoto, 1996] Hukushima, K. and Nemoto, K. (1996). Exchange monte carlo method and application to spin glass simulations. *Journal of the Physical Society of Japan*, 65(6):1604–1608.

[Hunter and Lange, 2004] Hunter, D. R. and Lange, K. (2004). A tutorial on mm algorithms. *The American Statistician*, 58(1):30–37.

[Jordan et al., 1999] Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233.

[Kaji et al., 2010] Kaji, D., Watanabe, K., and Watanabe, S. (2010). Phase transition of variational bayes learning in bernoulli mixture. *Australian Journal of Intelligent Information Processing Systems*, 11(4).

[Kohjima et al., 2015] Kohjima, M., Matsubayashi, T., and Sawada, H. (2015). Probabilistic non-negative inconsistent-resolution matrices factorization. In *24th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1855–1858.

[Kohjima et al., 2016] Kohjima, M., Matsubayashi, T., and Sawada, H. (2016). Multiple data analysis and non-negative matrix/tensor factorizaiton [I]: Multiple data analysis and its advances. *The Journal of the Institute of Electronics, Information and Communication Engineers*, 99(6).

[Kohjima et al., 2017] Kohjima, M., Matsubayashi, T., and Sawada, H. (2017). Probabilistic models based on non-negative matrix factorization for inconsistent resolution dataset analysis. *The IEICE Transactions on Information and Systems (Japanese Edition)*, 100(4):520–529.

[Kohjima and Watanabe, 2017] Kohjima, M. and Watanabe, S. (2017). Phase transition structure of variational bayesian nonnegative matrix factorization. In *International Conference on Artificial Neural Networks (ICANN)*, pages 146–154. Springer.

[Lahoti et al., 2018] Lahoti, P., Garimella, K., and Gionis, A. (2018). Joint non-negative matrix factorization for learning ideological leaning on twitter. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 351–359. ACM.

[Lee and Seung, 1999] Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.

[Lee and Seung, 2001] Lee, D. D. and Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562.

[Lee and Choi, 2009] Lee, H. and Choi, S. (2009). Group nonnegative matrix factorization for EEG classification. In *International Conference on Artificial Intelligence and Statistics*, pages 320–327.

[Li et al., 2001] Li, S. Z., Hou, X. W., Zhang, H. J., and Cheng, Q. S. (2001). Learning spatially localized, parts-based representation. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.

[Liu et al., 2010] Liu, C., Yang, H.-c., Fan, J., He, L.-W., and Wang, Y.-M. (2010). Distributed nonnegative matrix factorization for web-scale dyadic data analysis on mapreduce. In *Proceedings of the 19th international conference on World wide web*, pages 681–690. ACM.

[MacKay, 2003] MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

[Matsubayashi et al., 2016] Matsubayashi, T., Kohjima, M., and Sawada, H. (2016). Multiple data analysis and non-negative matrix/tensor factorizaiton [II・finish]: Tensor data analysis and applications. *The Journal of the Institute of Electronics, Information and Communication Engineers*, 99(7).

[Murphy, 2007] Murphy, K. P. (2007). Conjugate bayesian analysis of the univariate gaussian: a tutorial. `http://www.cs.ubc.ca/~murphyk/Papers/bayesGauss.pdf`.

[Nagata and Watanabe, 2008] Nagata, K. and Watanabe, S. (2008). Exchange monte carlo sampling from bayesian posterior for singular learning machines. *IEEE Transactions on Neural Networks*, 19(7):1253–1266.

[Nakajima et al., 2014] Nakajima, S., Sato, I., Sugiyama, M., Watanabe, K., and Kobayashi, H. (2014). Analysis of variational bayesian latent dirichlet allocation: Weaker sparsity than map. In *Advances in Neural Information Processing Systems*, pages 1224–1232.

[Nomura Research Institute, 2015] Nomura Research Institute (2015). 49 % of the japanese working population can be replaced by artificial intelligence and robots (nihon no roudou jinkou no 49% ga jinkoutinou ya robotto nado de daitai kanou ni). *News Release*. `https://www.nri.com/-/media/Corporate/jp/Files/PDF/news/newsrelease/cc/2015/151202_1.pdf`.

[Pauca et al., 2004] Pauca, V. P., Shahnaz, F., Berry, M. W., and Plemmons, R. J. (2004). Text mining using non-negative matrix factorizations. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 452–456. SIAM.

[Saha and Sindhwani, 2012] Saha, A. and Sindhwani, V. (2012). Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 693–702. ACM.

[Schmidt et al., 2009] Schmidt, M. N., Winther, O., and Hansen, L. K. (2009). Bayesian non-negative matrix factorization. In *Independent Component Analysis and Signal Separation*, pages 540–547.

[Schwarz, 1978] Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

[Shahnaz et al., 2006] Shahnaz, F., Berry, M. W., Pauca, V. P., and Plemmons, R. J. (2006). Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386.

[Singh and Gordon, 2008] Singh, A. P. and Gordon, G. J. (2008). Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658.

[Smaragdis and Brown, 2003] Smaragdis, P. and Brown, J. C. (2003). Non-negative matrix factorization for polyphonic music transcription. In *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180.

[Takeuchi et al., 2013] Takeuchi, K., Ishiguro, K., Kimura, A., and Sawada, H. (2013). Non-negative multiple matrix factorization. In *IJCAI*, volume 13, pages 1713–1720.

[Van der Vaart, 2000] Van der Vaart, A. W. (2000). *Asymptotic statistics*, volume 3. Cambridge university press.

[Vavasis, 2009] Vavasis, S. A. (2009). On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377.

[Wang et al., 2011] Wang, F., Li, T., Wang, X., Zhu, S., and Ding, C. (2011). Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22(3):493–521.

[Watanabe et al., 2009] Watanabe, K., Shiga, M., and Watanabe, S. (2009). Upper bound for variational free energy of bayesian networks. *Machine Learning*, 75(2):199–215.

[Watanabe and Watanabe, 2006] Watanabe, K. and Watanabe, S. (2006). Stochastic complexities of gaussian mixtures in variational bayesian approximation. *The Journal of Machine Learning Research*, 7:625–644.

[Watanabe and Watanabe, 2007] Watanabe, K. and Watanabe, S. (2007). Stochastic complexities of general mixture models in variational bayesian learning. *Neural Networks*, 20(2):210–219.

[Watanabe, 2001a] Watanabe, S. (2001a). Algebraic analysis for nonidentifiable learning machines. *Neural Computation*, 13(4):899–933.

[Watanabe, 2001b] Watanabe, S. (2001b). Algebraic geometrical methods for hierarchical learning machines. *Neural Networks*, 14(8):1049–1060.

[Watanabe, 2009] Watanabe, S. (2009). *Algebraic geometry and statistical learning theory*, volume 25. Cambridge University Press.

[Xu et al., 2003] Xu, W., Liu, X., and Gong, Y. (2003). Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273. ACM.

[Yamazaki and Watanabe, 2002] Yamazaki, K. and Watanabe, S. (2002). Stochastic complexity of bayesian networks. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 592–599. Morgan Kaufmann Publishers Inc.

[Yamazaki and Watanabe, 2003] Yamazaki, K. and Watanabe, S. (2003). Singularities in mixture models and upper bounds of stochastic complexity. *Neural networks*, 16(7):1029–1038.

[Yamazaki and Watanabe, 2005] Yamazaki, K. and Watanabe, S. (2005). Algebraic geometry and stochastic complexity of hidden markov models. *Neurocomputing*, 69(1-3):62–84.

[Yin et al., 2014] Yin, J., Gao, L., and Zhang, Z. M. (2014). Scalable nonnegative matrix factorization with block-wise updates. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 337–352. Springer.

[Zhang et al., 2006] Zhang, S., Wang, W., Ford, J., and Makedon, F. (2006). Learning from incomplete ratings using non-negative matrix factorization. In *Proceedings of the 2006 SIAM international conference on data mining*, pages 549–553. SIAM.