LEARNING MIXED DIVERGENCES IN COUPLED MATRIX AND TENSOR FACTORIZATION MODELS

Umut Şimşekli, Ali Taylan Cemgil, Beyza Ermiş

Dept. of Computer Engineering, Boğaziçi University, 34342, Bebek, İstanbul, Turkey

ABSTRACT

Coupled tensor factorization methods are useful for sensor fusion, combining information from several related datasets by simultaneously approximating them by products of latent tensors. In these methods, the choice of a suitable optimization criteria becomes difficult as observed datasets may have different statistical characteristics and their relative importance for the task at hand can vary. In this paper, we present an algorithmic framework for coupled factorization that, while estimating a latent factorization also estimates a specific β-divergence for each dataset as well as the relative weights in an overall additive cost function. We evaluate the proposed method on both synthetical and real datasets, where we apply our methods on a link prediction problem. The results show that our method outperforms the state-of-the-art by a significant margin.

Index Terms— Divergence learning, Coupled tensor factorizations, Tweedie distribution

1. INTRODUCTION

Coupled tensor factorization methods are useful in various application areas such as audio processing [1], computational psychology [2], bioinformatics [3] or collaborative filtering [4], where information from diverse sources are available and need to be combined for arriving at useful predictions. Examples of such situations are abound: for example for product recommendation, a product-buyer rating matrix can be enhanced with demographic information from the customer and connectivity information from a social network. In musical audio processing, one example is having a large collection of annotated audio data and symbolic music score information. The common theme in all such applications is the data fusion problem.

As a warm up, let us consider an example coupled matrix factorization model where two observed data matrices X_1 and X_2 are collectively decomposed as

$$X_1(i,m) \approx \hat{X}_1(i,m) = \sum\nolimits_k Z_1(i,k)Z_3(k,m)$$
$$X_2(j,m) \approx \hat{X}_2(j,m) = \sum\nolimits_k Z_2(j,k)Z_3(k,m) \qquad (1)$$

The factor Z_3 is the *shared factor* in both decompositions, making the overall model coupled. This coupled model has shown to be useful various fields [2, 3, 5, 6]. The aim in this model is to estimate the latent factors Z_1, Z_2, and Z_3 given X_1 and X_2, where we need to solve the following optimization problem:

$$Z^{\star}_{1:3} = \underset{Z_1,Z_2,Z_3}{\arg\min} \left[\frac{1}{\phi_1} D_1(X_1||Z_1Z_3) + \frac{1}{\phi_2} D_2(X_2||Z_2Z_3) \right] \qquad (2)$$

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) Grant no. 113M492. U. Ş. is funded by a PhD Scholarship from TUBITAK.

where D_1 and D_2 are *divergence functions* measuring the approximation error and the *dispersion parameters* ϕ_1 and ϕ_2 are the relative weights for the error in the approximation to each observed tensor.

Another coupled factorization model that is popular in link-prediction applications [7] is given as follows:

$$X_1(i,j,k) \approx \hat{X}_1(i,j,k) = \sum\nolimits_r Z_1(i,r)Z_2(j,r)Z_3(k,r)$$
$$X_2(i,m) \approx \hat{X}_2(i,m) = \sum\nolimits_r Z_1(i,r)Z_4(m,r)$$
$$X_3(j,n) \approx \hat{X}_3(j,n) = \sum\nolimits_r Z_2(j,r)Z_5(n,r) \qquad (3)$$

where X_1 is decomposed by using a Parafac model and the side informations X_2 and X_3 are decomposed by using different matrix factorization models.

In applications, as we will also demonstrate in our experiments, one often needs to develop custom model topologies, where either the observed objects or the latent factors have multiple entities and cannot be represented without loss of structure using a matrix. To have this modeling flexibility for real world data sets that may consist of several tensors and require custom models, we would like to develop an algorithmic framework that is able to handle a broad variety of model topologies. In this study, we make use of the Generalized Coupled Tensor Factorization (GCTF) framework [8] that aims to cover all possible model topologies and coupled factorization models. In this framework, there are N_x different observed tensors $\{X_\nu\}_{\nu=1}^{N_x}$, each of them approximated by an output tensor $\{\hat{X}_\nu\}_{\nu=1}^{N_x}$, where these output tensors are functions of N_z different latent factors $\{Z_\alpha\}_{\alpha=1}^{N_z}$. In this notation, we refer vectors as tensors with 1 index and matrices as tensors with 2 indices.

Example 1. In Eq.1, we have $N_x = 2$ observed tensors and $N_z = 3$ latent factors. Similarly, in Eq.3, we have $N_x = 3$ observed tensors and $N_z = 5$ latent factors. The output tensors $\hat{X}_\nu$ are model-specific functions of the latent factors Z_α as illustrated in Eqs. 1 and 3.

Given the dispersions and the divergence functions, the optimal latent factors can be found by minimizing the following objective:

$$Z^{\star}_{1:N_z} = \underset{Z_{1:N_z}}{\arg\min} \sum_{\nu=1}^{N_x} \frac{1}{\phi_\nu} D_\nu(X_\nu||\hat{X}_\nu) \qquad (4)$$

However, in practice the dispersion parameters and the divergence functions are not known. In coupled models the success of a method may hinge criticaly on a good setting of these parameters, yet manual selection is not straightforward especially when the number of observed tensors, N_x, is large. Hence, we will be concerned with the following problems:

Estimation of the dispersions: The dispersion parameters ϕ_ν play a key role in coupled factorizations as they form the balance between the approximation error to X_ν for example observations may have

been recorded using different and unknown scales. Typically, such weight parameters are selected manually [5, 9] and data is assumed to be suitably preprocessed. In a statistical setting, these relative weights are directly proportional to the observation noise variances and can be estimated directly from data.

Automatic selection of the divergences: Euclidean divergence is commonly used in tensor models, implicitly related to a conditionally Gaussian noise assumption. However, heavy-tailed noise distributions are often needed for robust estimation and more specific noise models are needed for sparse data, where Gaussian assumptions fall short. Choosing suitable divergence functions D_ν becomes even more critical in coupled models due to the data heterogeneity, where the observed tensors X_ν may have different statistical characteristics. In such cases, it is useful to choose a specific divergence for each observed matrix, where we call total cost functions such as Eq.4 as *mixed divergences*.

In this study, we present a novel algorithmic framework for coupled tensor factorizations where we jointly estimate the dispersions ϕ_ν, divergence functions D_ν, and the factors Z_α with arbitrary model topologies. We formulate the problem from a probabilistic perspective, as making inference in Tweedie models that is an important special case of exponential dispersion models. We apply our method on synthetic and real datasets for a link prediction problem where the aim is to predict the missing parts of an observed tensor. The results show that our method outperforms the state-of-the-art by a significant margin.

In the rest of the paper, we will make use of the following vector notation where we define, $x_\nu \equiv \mathbf{vec}(X_\nu)$ and $\hat{x}_\nu \equiv \mathbf{vec}(\hat{X}_\nu)$. Here, $\mathbf{vec}(\cdot)$ is the vectorization operator (i.e. the colon operator in Matlab). We index the elements of the vectorized tensors as $x_\nu(i)$, where $i \in \{1, \dots, S_\nu\}$ and S_ν is the number of elements in x_ν.

2. THE MODEL

The GCTF framework assumes the following probabilistic model on the observed tensors: [8]

$$x_\nu(i) \sim \mathcal{TW}_{p_\nu}\big(x_\nu(i); \hat{x}_\nu(i), \phi_\nu\big), \qquad \nu = 1, \dots, N_x$$

where $\mathcal{TW}$ denotes the Tweedie distribution that is an important special case of exponential dispersion models, characterized by three parameters: mean, dispersion, and power. Tweedie densities $\mathcal{TW}_p(x; \hat{x}, \phi)$ can be written in the following moment form: $\log \mathbb{P}(x; \hat{x}, \phi, p) = -\log K(x, \phi, p) - \phi^{-1} d_p(x||\hat{x})$, where $K(\cdot)$ is the normalizing constant, $\hat{x}$ is the mean, ϕ is the dispersion, p is the power parameter and $d_p(\cdot)$ denotes the β-divergence defined as:

$$d_p(x||\hat{x}) = \frac{x^{2-p}}{(1-p)(2-p)} - \frac{x\hat{x}^{1-p}}{1-p} + \frac{\hat{x}^{2-p}}{2-p}, \;\; (p = 2 - \beta) \tag{5}$$

This model is also known as the *power variance* model, since the variance of the data has the following form: $\mathrm{var}(x) = \phi \hat{x}^p$.

By taking appropriate limits, it is easy to verify that $d_p(\cdot)$ is the Euclidean distance square, information divergence, and Itakura-Saito divergence for $p = 0, 1, 2$, respectively. In the probabilistic counterpart, different choices of p yield well-known important distributions such as Gaussian ($p = 0$), Poisson ($p = 1$), compound Poisson ($1 < p < 2$), gamma ($p = 2$) and inverse Gaussian ($p = 3$) distributions. Excluding the interval $0 < p < 1$ for which no exponential dispersion model exists, for all other values of p, one obtains Tweedie stable distributions [10].

An important property is that the normalization constant K does not depend on $\hat{x}$; hence it is easy to see that for fixed p and ϕ, solving a maximum likelihood problem for $\hat{x}$ is indeed equivalent to minimization of the β-divergence. For the familiar Gaussian case, we have $\mathcal{TW}_0(x; \hat{x}, \phi) = (2\pi\phi)^{-1/2} \exp(-\phi^{-1} d_0(x; \hat{x}))$ where $d_0(x; \hat{x}) = (x - \hat{x})^2/2$ and the dispersion is simply the variance. As for all admissible p, we have a similar form; the Tweedie models generalize the established theory of least squares linear regression to more general noise models (restricted to identity link functions).

In this probabilistic setting, the power parameter p_ν determines the divergence function (e.g., D_ν in Eq.4), the dispersion ϕ_ν determines the relative weight of the cost, and the mean parameter $\hat{x}_\nu$ is the output of the desired factorization as exemplified in Eqs.1 and 3. The key contribution of this study is to jointly estimate all of the parameters $p_{1:N_x}$, $\phi_{1:N_x}$, and $Z_{1:N_z}$ for the complicated cases of p, which we will describe in the next section.

Even though coupled factorization models have been widely studied in several domains, estimation of the relative weights and divergences (here, corresponding to ϕ and p) has not been extensively explored. In [2], a maximum likelihood estimator for ϕ was presented under Gaussian observation models. We developed a similar approach in [11], where we presented a MAP schema for estimating ϕ for the cases where $p \in \{0, 1, 2, 3\}$. In [12] a score matching approach was proposed for estimating the power parameter p only for the non-negative matrix factorization model that uses a Tweedie distribution with unitary dispersion ($\phi = 1$) as the observation model. Recently, a score matching approach was proposed in [13] for making inference in Tweedie distributions with $p \geq 0$, where the Tweedie distribution was approximated by an 'exponential divergence' distribution. The authors estimated the dispersion and power parameter by running a grid search procedure on this approximate distribution. Besides the $\beta-$divergence, the authors also enabled automatic selection of α, γ, and Rényi divergences through non-linear transformations. In [14], we presented three different methods for estimating the parameters ϕ and p for factorization models with compound Poisson observations. However, these methods are only valid for $p \in (1, 2)$ and cannot be generalized for the other cases. The need for a general and efficient method for coupled factorization models that would handle the whole Tweedie family still prevails.

3. INFERENCE

In this study we propose a novel algorithmic framework for maximum a-posteriori estimation, where the aim is to solve the following optimization problem:

$$\max_{\substack{Z_{1:N_z} \\ \phi_{1:N_x} \\ p_{1:N_x}}} \sum_{\nu=1}^{N_x} \left[\sum_{i=1}^{S_\nu} \Big(K_{\nu i} - \frac{1}{\phi_\nu} d_p(x_\nu(i)||\hat{x}_\nu(i)) \Big) + \log \mathbb{P}(\phi_\nu) \right] \tag{6}$$

where $K_{\nu i} = -\log K(x_\nu(i), \phi_\nu, p_\nu)$ and $\mathbb{P}(\phi_\nu)$ is the conjugate prior distribution of the dispersions, that is an inverse gamma distribution: $\phi_\nu \sim \mathcal{IG}(\phi_\nu; \tau_\phi, \kappa_\phi)$. Note that, when the dispersion and power parameters are known, the normalizing constant $K(\cdot)$ and the conjugate prior become irrelevant and the problem reduces to β-divergence minimization. However, when these parameters are not known, we need to deal with the normalizer $K(\cdot)$, an expression without a simple closed form apart from the cases $p = 0, 1, 2, 3$. Here, we focus on these challenging cases where $p \notin \{0, 1, 2, 3\}$.

In order to estimate the latent factors, dispersions, and power parameters jointly, we propose an iterative schema where we divide the optimization problem of Eq.6 into simpler subproblems. The

ultimate method is a coordinate descent algorithm, where each parameter is updated at each iteration given the up-to-date values of the remaining parameters. Here, each iteration i consists of three estimation steps, stated as follows:

$$Z_\alpha^{(i+1)} = \arg\max_{Z_\alpha} \sum\nolimits_{\nu=1}^{N_x} \log \mathbb{P}(x_\nu | Z_{1:N_\alpha}, \phi_\nu^{(i)}, p_\nu^{(i)}), \quad \forall \alpha \quad (7)$$

$$\phi_\nu^{(i+1)} = \arg\max_{\phi_\nu} \log\big(\mathbb{P}(x_\nu | \phi_\nu, \hat{x}_\nu^{(i+1)}, p^{(i)}) \mathbb{P}(\phi_\nu)\big), \quad \forall \nu \quad (8)$$

$$p_\nu^{(i+1)} = \arg\max_{p_\nu} \log \mathbb{P}(x_\nu | \hat{x}_\nu^{(i+1)}, \phi_\nu^{(i+1)}, p_\nu), \quad \forall \nu \quad (9)$$

Given the dispersions and the power parameters, the first problem (Eq.7) reduces to the well-known problem of minimizing the β-divergence between the observations X_ν and the model outputs $\hat{X}_\nu$ with respect to $Z_{1:N_z}$ (see Eq.4). Therefore, any standard algorithm that minimizes the β-divergence can be used here. In this study, for this task we make use of the multiplicative update rules given in [8].

3.1. Learning Mixed β-Divergences

The maximum likelihood estimate of ϕ has analytical solution only for the Gaussian and the inverse Gaussian distributions. Inferring ϕ is intractable for the other cases. In our previous work [11], we showed that the inference becomes tractable for the Poisson and gamma distributions when the gamma functions in the probability mass and density functions are approximated with Stirling's approximation. The MAP estimate of ϕ for the cases $p \in \{0, 1, 2, 3\}$ is given as follows: [11]

$$\phi_\nu^\star = \frac{\Big(\sum_{i=1}^{N} d_{p_\nu}(x_\nu(i) || \hat{x}_\nu(i))\Big) + \kappa_\phi}{S_\nu/2 + \tau_\phi + 1}, \quad p \in \{0, 1, 2, 3\} \quad (10)$$

where $d_p(\cdot)$ is the β-divergence, defined in Eq.5.

Here, we focus on the remaining cases of p, where the probability density functions cannot be written in closed-form analytical expressions. However, they can be expressed as infinite series that is defined as follows: [10]

$$\mathcal{TW}_p(x; \hat{x}, \phi) = \frac{1}{x \xi_p} \left(\sum_{k=1}^{\infty} V_k(x, p, \phi) \right) a(x, \hat{x}, p) \quad (11)$$

$$\text{where,} \quad a(x, \hat{x}, p) = \exp\left\{ \frac{1}{\phi} \left(\frac{\hat{x}^{1-p} x}{1-p} - \frac{\hat{x}^{2-p}}{2-p} \right) \right\}$$

and $\xi_p = 1$ for $p \in (1, 2)$ and $\xi_p = \pi$ otherwise.

The Tweedie density with $p \in (1, 2)$ coincides with the compound Poisson distribution [10]. The compound Poisson distribution is an interesting distribution as it has a support for continuous positive data and a discrete probability mass at zero. The presence of the discrete mass at zero makes this distribution suitable for such applications where the observations are sparse [14]. For $x = 0$, the density function is defined as $\mathcal{TW}_p(x; \cdot) = \exp(\hat{x}^{2-p}/(\phi(p-2)))$ and for $x > 0$, it follows the form of Eq.11, where the terms V_k for this distribution is defined as follows: (with $\alpha = (2-p)/(1-p)$)

$$V_k(x, p, \phi) = \frac{x^{-k\alpha} (p-1)^{k\alpha} \phi^{k(\alpha-1)}}{(2-p)^k \Gamma(k+1) \Gamma(-k\alpha)} \quad (12)$$

The cases $p < 0$ and $p > 2$ of the Tweedie class correspond to Tweedie stable distributions. Tweedie stable models are heavy-tailed distributions and they are left-skewed for $p < 0$ and right-skewed for $p > 2$. The Tweedie stable models with $p > 2$ can be useful for many applications, including audio signal processing [15] and computer networks [16]. The Tweedie stable models with $p < 0$ can be used for risk modeling [17], however their applications on coupled factorization models are limited. We present the derivations for $p < 0$ for completeness. For the Tweedie models with $p < 0$ and $p > 2$, the terms V_k are defined as follows: [10]

$$V_k(x, p, \phi) = \frac{\Gamma(1 + \frac{k}{\alpha}) \phi^{\frac{k}{p-2}} (-1)^k \sin(\frac{k\pi}{\alpha})}{\Gamma(k+1)(1-p)^k (2-p)^{-\frac{k}{\alpha}} x^{-k}}, \qquad (p < 0)$$

$$V_k(x, p, \phi) = \frac{\Gamma(1 + \alpha k) \phi^{k(1-\alpha)} (-1)^k \sin(-k\pi\alpha)}{\Gamma(k+1)(p-1)^{-\alpha k} (p-2)^k x^{\alpha k}}. \quad (p > 2)$$

In order to estimate the dispersions in the compound Poisson and the Tweedie stable distributions, we use a limited memory quasi-Newton method, namely the L-BFGS-B algorithm [18]. This method requires the gradient of the map objective function that is given as follows:

$$\frac{\partial g(\phi_\nu)}{\partial \phi_\nu} = \frac{1}{\phi_\nu^2} \Big[\sum_{i=1}^{S_\nu} \Big(\frac{\hat{x}_\nu(i)^{2-p_\nu}}{2-p_\nu} + \frac{x_\nu(i) \hat{x}_\nu(i)^{1-p_\nu}}{p_\nu - 1} \Big) + \kappa_\phi \Big] - \frac{1}{\phi_\nu} \Big[c_p \frac{\sum_{i=1}^{S_\nu} \sum_{k=1}^{\infty} k V_k(x_\nu(i), p_\nu, \phi_\nu)}{\sum_{i=1}^{S_\nu} \sum_{k=1}^{\infty} V_k(x_\nu(i), p_\nu, \phi_\nu)} + \tau_\phi + 1 \Big] \quad (13)$$

where $g(\phi_\nu) = -\log \mathbb{P}(x_\nu, \phi_\nu | \hat{x}_\nu, p_\nu)$ and $c_p = 1 - p_\nu$ for $p_\nu < 0$ and $c_p = 1/(p_\nu - 1)$ otherwise.

The gradient requires two infinite summations to be computed, which is intractable. In this study, we utilize efficient numerical methods by following [19] for approximate computation of these summations. This method locates the indices k where the terms V_k make the major contribution to the sum. The infinite sum is then approximated by summing up the terms in the located region. The cases $p \in (1, 2)$ and $p > 2$ is described in [19]; for completeness we explain the method for $p < 0$ in the supp. document [20].

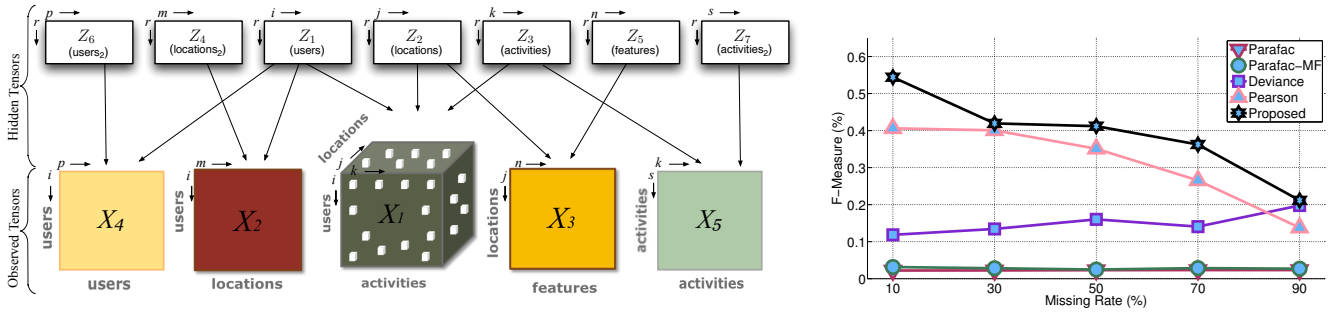
The last step of the proposed method (Eq.9) is to compute the maximum likelihood estimate of the power parameter p. Unfortunately, the optimal p does not have an analytical solution; the state-of-the-art is based on running numerical methods on this problem [19, 21, 14]. In this study, we utilize a grid search procedure in order to estimate the power parameter p given the other parameters. Note that, even though it is not explicitly demonstrated in this study, the proposed methods are scalable; in large-scale settings, they can be implemented in an embarrassingly parallel fashion as the problem is separable over the indices i. The pseudo-code of the proposed method is provided in the supp. document [20].

4. EXPERIMENTS

4.1. Synthetic Data

We illustrate the proposed method on the simple model defined in Eq.1. Here, we randomly generate the latent variables $Z_{1:3}$, $\phi_{1:2}$, power parameters $p_{1:2}$, and the observed tensors $X_{1:2}$. Our aim is to find the MAP estimates of all the latent variables given X_1 and X_2.

Since the true values of the latent variables and the global optimum of Eq.6 might not coincide, in order to approximate the global optimum of Eq.6, we first conduct 'oracle' experiments where we assume that the global optimum would be near the true values of the variables. In these experiments, we initialize all the variables



(a) (b)

Fig. 1. a) General sketch of the proposed model. The blocks visualize the tensors that are defined in the model. The lower-case letters and arrows near the blocks represent the indices of a particular tensor. b) F-measure comparison of the proposed method and the state-of-the-art.

$(Z_{1:N_z},\ \phi_{1:N_x},\ p_{1:N_x})$ to their true values and run the proposed method in order to find the local optimum that is closest to the true values of the variables. We treat the oracle estimates as the global optimum. Then, we re-run the proposed method by initializing the variables randomly. We measure the mean squared error (MSE) between the oracle values of the power and dispersion parameters and the values that we obtain with random initialization.

In our experiments, we set the sizes of the observed indices equal to each other: $|i| = |j| = |m| = s$ and we set $|k| = 1$. We explore three different values for s: 25, 50, and 100 and repeat the experiments 100 times for each configuration of s. Table 1 shows the results. The results show that, even with a small amount of data, our method is capable of estimating the power and the dispersion parameters accurately. Besides, the MSE is gracefully degrading as the size of data increases.

4.2. Link Prediction

In this section, we address the missing link prediction task, where the aim is to predict missing parts of an observed tensor. We evaluate our method on the UCLAF dataset [4]. This dataset has a main tensor X_1 of size $146 \times 168 \times 5$, which encapsulates *user-location-activity* informations, where $X_1(i,j,k) = 1$ if the user i visits location j and performs activity k there and $X_1(i,j,k) = 0$ otherwise. The dataset also includes additional side information: the user-location preferences matrix X_2, the location-feature matrix X_3, the user-user similarity matrix X_4, and the activity-activity matrix X_5. The aim in this application is to predict the missing parts of X_1.

By following a similar approach to [7], we model this dataset by using the following coupled factorization model:

$$
\begin{aligned}
X_1(i,j,k) = \hat{X}_1(i,j,k) &= \sum\nolimits_r Z_1(i,r) Z_2(j,r) Z_3(k,r), \\
X_2(i,m) = \hat{X}_2(i,m) &= \sum\nolimits_r Z_1(i,r) Z_4(m,r), \\
X_3(j,n) = \hat{X}_3(j,n) &= \sum\nolimits_r Z_2(j,r) Z_5(n,r), \\
X_4(i,p) = \hat{X}_4(i,p) &= \sum\nolimits_r Z_1(i,r) Z_6(p,r), \\
X_5(k,s) = \hat{X}_5(k,s) &= \sum\nolimits_r Z_3(k,r) Z_7(s,r)
\end{aligned}
$$

where X_1 is decomposed by using a Parafac model and the remaining observed tensors are decomposed by using matrix factorization (MF) models. Fig.1(a) visualizes the general structure of the model. In our experiments, we erase random parts of X_1 at varying amounts (i.e., $\{10\%, 30\%, 50\%, 70\%, 90\%\}$) and evaluate our method on the prediction of the erased parts. We set the number of components to $|r| = 5$ and use the *F-measure* as the evaluation metric.

Table 1. The results of the experiments on synthetical data.

	$s = 25$	$s = 50$	$s = 100$
MSE (power)	0.0822	0.0563	0.0635
MSE (dispersion)	0.9087	0.6933	0.2763

We compare our method with two other methods: 1) a Parafac model with Euclidean cost [22] that makes use of only X_1, 2) the complete model (Parafac-MF) with with Euclidean cost and unitary dispersions [8] ($p_{1:5} = 0$ and $\phi_{1:5} = 1$). The second method can also be considered as extended versions of [7, 23]. We also compare our dispersion estimation method with two different dispersion estimators that have not been explored for coupled factorization models; yet commonly used in the generalized linear models literature, namely the Pearson and mean deviance estimators [24]. For these estimators, we replace Eq.8 with one of these estimators and use the same approach for estimating the other variables. Note that, each step of our method (Eqs.7-9) monotonically increases the likelihood, whereas the Pearson and deviance estimators do not have such guarantee (for non-Gaussian cases); the likelihood might fluctuate over the iterations when they are used in our iterative schema.

Figure 1(b) visualizes the results of this experiment. We can observe that, both benchmark methods (Parafac and Parafac-MF) perform poorly, where introducing side information (Parafac-MF) results in a tiny improvement in the performance. As we can also observe, apart from monotonically increasing the likelihood and achieving a consistent and sound method, the proposed approach also outperforms the other estimators, in particular when the percentage of missing data is low. Joint estimation of $p_{1:5}$ and $\phi_{1:5}$ yields significant performance improvement, where we have 40% F-measure improvement even when half of the data is missing.

5. CONCLUSION

We presented an algorithmic framework for coupled tensor factorization to simultaneously estimate latent factors, specific divergences and their relative weights in an overall additive cost function, where the number of observed tensors, the number of latent factors, and the model topologies can be arbitrary. We applied our method on synthetic and real datasets where we outperformed the state-of-the-art by a significant margin on a link prediction application.

6. REFERENCES

[1] L. Le Magoarou, A. Ozerov, and N. Q. K. Duong, "Text-informed audio source separation using nonnegative matrix partial co-factorization," in *MLSP*, 2013.

[2] T.F. Wilderjans, E. Ceulemans, I. Van Mechelen, and R.A. van den Berg, "Simultaneous analysis of coupled data matrices subject to different amounts of noise.," *Br J Math Stat Psychol*, vol. 64, pp. 277–90, 2011.

[3] E. Acar, G. Gurdeniz, M. A. Rasmussen, D. Rago, L. O. Dragsted, and R. Bro, "Coupled matrix factorization with sparse factors to identify potential biomarkers in metabolomics.," *IJKDB*, vol. 3, no. 3, pp. 22–43, 2012.

[4] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative filtering meets mobile recommendation: A user-centered approach," in *AAAI'10*, 2010.

[5] M. Kim, J. Yoo, K. Kang, and S. Choi, "Nonnegative matrix partial co-factorization for spectral and temporal drum source separation," *J. Sel. Topics Signal Processing*, vol. 5, no. 6, pp. 1192–1204, 2011.

[6] U. Simsekli and A. T. Cemgil, "Score guided musical source separation using generalized coupled tensor factorization," in *EUSIPCO*, 2012.

[7] Beyza Ermis, Evrim Acar Ataman, and Taylan Cemgil, "Link prediction in heterogeneous data via generalized coupled tensor factorization," *Data Mining and Knowledge Discovery*, 2014.

[8] Y. K. Yılmaz, A. T. Cemgil, and U. Şimşekli, "Generalised coupled tensor factorisation," in *NIPS*, 2011.

[9] T. Barker, T. Virtanen, and O. Delhomme, "Ultrasound-coupled semi-supervised nonnegative matrix factorisation for speech enhancement," in *ICASSP*, 2014.

[10] B. Jørgensen, *The Theory of Dispersion Models*, Chapman & Hall/CRC Monographs on Statistics & Applied Probability, 1997.

[11] U. Şimşekli, B. Ermiş, A. T. Cemgil, and E. Acar, "Optimal weight learning for coupled tensor factorization with mixed divergences," in *EUSIPCO*, 2013.

[12] Z. Lu, Z. Yang, and E. Oja, "Selecting β-divergence for nonnegative matrix factorization by score matching," in *Proceedings of 22nd International Conference on Artificial Neural Networks (ICANN 2012)*, Lausanne, Switzerland, 2012, vol. 7553 of *Lecture Notes in Computer Science*, pp. 419–426, Springer.

[13] Onur Dikmen, Zhirong Yang, and Erkki Oja, "Learning the information divergence," *CoRR*, 2014.

[14] U. Şimşekli, A. T. Cemgil, and Yılmaz K., "Learning the beta-divergence in tweedie compound poisson matrix factorization models," in *ICML*, 2013.

[15] S. Godsill, "Mcmc and em-based methods for inference in heavy-tailed processes with alpha-stable innovations," in *IEEE Workshop on Higher-Order Stats.*, 1999.

[16] G. Xiaohu, Z. Guangxi, and Z. Yaoting, "On the testing for alpha-stable distributions of network traffic," *Comput. Commun.*, vol. 27, pp. 447–457, 2004.

[17] J. B. Krawczyk, "Dependence of left-skewed payoff distributions on risky-asset price uncertainty," in *Quantitative Methods in Finance Conference*, 2005.

[18] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited-memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, pp. 1190–1208, 1994.

[19] P. K. Dunn and G. S. Smyth, "Series evaluation of tweedie exponential dispersion model densities," *Stats. & Comp.*, vol. 15, pp. 267–280, 2005.

[20] U. Şimşekli, A. T. Cemgil, and Beyza Ermis, "Learning mixed divergences in coupled matrix and tensor factorization models: Supplementary document," `http://www.cmpe.boun.edu.tr/~umut/icassp15_divergencelearning`.

[21] Y. Zhang, "Likelihood-based and bayesian methods for tweedie compound poisson linear mixed models," *Statistics and Computing*, vol. accepted, 2012.

[22] R. Bro, "PARAFAC. Tutorial and applications," *Chemometrics and Intelligent Laboratory Systems*, 1997.

[23] Evrim Acar, Tamara G. Kolda, and Daniel M. Dunlavy, "All-at-once optimization for coupled matrix and tensor factorizations," in *MLG'11: Proceedings of Mining and Learning with Graphs*, 2011.

[24] C. E. McCulloch and J. A. Nelder, *Generalized Linear Models*, Chapman and Hall, 2nd edition, 1989.