LEARNING MIXED DIVERGENCES IN COUPLED MATRIX AND TENSOR FACTORIZATION MODELS

Umut Şimşekli, Ali Taylan Cemgil, Beyza Erimiş

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 $\beta$-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

\[
X_1(i, m) \approx \hat{X}_1(i, m) = \sum_k Z_1(i, k) Z_3(k, m) \\
X_2(j, m) \approx \hat{X}_2(j, m) = \sum_k Z_2(j, k) Z_3(k, m) \tag{1}
\]

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:

\[
Z_{1:3}^* = \arg \min_{Z_1, Z_2, Z_3} \left[ \frac{1}{\phi_1} D_1(X_1 || Z_1 Z_3) + \frac{1}{\phi_2} D_2(X_2 || Z_2 Z_3) \right] \tag{2}
\]

where $D_1$ and $D_2$ are divergence functions measuring the approximation error and the dispersion parameters $\phi_1$ and $\phi_2$ 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:

\[
X_1(i, j, k) \approx \hat{X}_1(i, j, k) = \sum_r Z_1(i, r) Z_2(j, r) Z_3(k, r) \\
X_2(i, m) \approx \hat{X}_2(i, m) = \sum_r Z_1(i, r) Z_4(m, r) \\
X_3(j, n) \approx \hat{X}_3(j, n) = \sum_r Z_2(j, r) Z_5(n, r) \tag{3}
\]

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 $N_x$ different latent factors $\{Z_\alpha\}_{\alpha=1}^{N_x}$. 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 $N_x = 3$ latent factors. Similarly, in Eq.3, we have $N_x = 3$ observed tensors and $N_x = 5$ latent factors. The output tensors $\hat{X}_\nu$ are model-specific functions of the latent factors $Z_\alpha$ 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:

\[
Z_{1:N_x}^* = \arg \min_{Z_1, \ldots, Z_{N_x}} \sum_{\nu=1}^{N_x} \frac{1}{\phi_\nu} D_\nu(X_\nu || \hat{X}_\nu) \tag{4}
\]

However, in practice the dispersion parameters and the divergence functions are not known. In coupled models the success of a method may hinge critically 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 $\phi_\nu$ play a key role in coupled factorizations as they form the balance between the approximation error to $X_\nu$ for example observations may have

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.
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 \( D_\nu \) becomes even more critical in coupled models due to the data heterogeneity, where the observed tensors \( X_\nu \) 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 dispersion 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 dispersion 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 dispersion 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 dispersion 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 dispersion 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 dispersion 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.
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:

$$Z^{(i+1)} = \arg \max_{x \in \mathcal{X}} \sum_{\nu=1}^{N_C} \log P(x|Z_1,N_\nu,\phi^{(i)}_\nu,p^{(i)}_\nu), \quad \forall \alpha \quad (7)$$

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

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

Given the dispersions and the power parameters, the first problem (Eq.7) reduces to the well-known problem of minimizing the $\beta$-divergence between the observations $X_\nu$ and the model outputs $\hat{X}_\nu$ with respect to $Z_1,N_\nu$ (see Eq.4). Therefore, any standard algorithm that minimizes the $\beta$-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 $\beta$-Divergences

The maximum likelihood estimate of $\phi$ has analytical solution only for the Gaussian and the inverse Gaussian distributions. Inferring $\phi$ 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 $\phi$ for the cases $p \in \{0, 1, 2, 3\}$ is given as follows: [11]

$$\hat{\phi}_p = \left( \sum_{i=1}^{N_C} d_{p}(x_{\nu}(i)|\hat{x}_{\nu}(i)) \right) + \eta_\phi \quad \forall \nu \quad (10)$$

where $d_p(\cdot)$ is the $\beta$-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]

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

where

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

and $\xi_p = 1$ for $p \in \{1, 2\}$ and $\xi_p = \sigma$ 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 $\mathcal{T} W_p(x; \cdot) = \exp(\frac{\hat{x}^{2-p}}{(\phi(p-2))})$ 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]

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

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

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:

$$\frac{\partial g(\phi_\nu)}{\partial \phi_\nu} = \frac{1}{\phi_\nu} \sum_{i=1}^{S_\nu} \frac{\hat{x}_{\nu}(i)^{2-p_\nu} + \phi_{\nu} x_{\nu}(i) \hat{x}_{\nu}(i)^{1-p_\nu}}{2 - p_\nu - 1} + \kappa_\nu$$

$$- \frac{1}{\phi_\nu} \left[ c_{\nu} \sum_{i=1}^{S_\nu} \sum_{k=1}^{\infty} k V_k(\nu_\nu(i), p_\nu, \phi_\nu) \right] + \tau_\nu + 1 \quad (13)$$

where $g(\phi_\nu) = - \log P(x_{\nu}, \phi_\nu | \hat{x}_{\nu}, p_\nu)$ and $c_{\nu} = 1 - p_\nu$ for $p_\nu < 0$ and $c_{\nu} = 1/(p_\nu - 1)$ 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
In order to evaluate our models, we have conducted several test cases. In all our experiments the audio is subdivided into frames of size yielding a simple concatenative synthesis algorithm and then we have measured 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 \(k\), 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:

\[
X_1(i,j,k) = \hat{X}_1(i,j,k) = \sum_s Z_1(i,r)Z_2(j,r)Z_3(k,r),
\]

\[
X_2(i,m) = \hat{X}_2(i,m) = \sum_r Z_1(i,r)Z_4(m,r),
\]

\[
X_3(j,n) = \hat{X}_3(j,n) = \sum_r Z_2(j,r)Z_5(n,r),
\]

\[
X_4(i,p) = \hat{X}_4(i,p) = \sum_r Z_1(i,r)Z_6(p,r),
\]

\[
X_5(k,s) = \hat{X}_5(k,s) = \sum_r Z_3(k,r)Z_7(s,r)
\]

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 (\(Z_1.N_1, \phi_1.N_2, p_1.N_2\)) 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.

### 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.

---

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

<table>
<thead>
<tr>
<th>Method</th>
<th>MSE (power)</th>
<th>MSE (dispersion)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parafac</td>
<td>0.0822</td>
<td>0.9087</td>
</tr>
<tr>
<td>Parafac-MF</td>
<td>0.0563</td>
<td>0.6933</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.0635</td>
<td>0.2763</td>
</tr>
</tbody>
</table>

---
6. REFERENCES


