LogPar: Logistic PARAFAC2 Factorization for Temporal Binary Data with Missing Values

Kejing Yin
Hong Kong Baptist University
cskjyin@comp.hkbu.edu.hk

Ardavan Afshar
Georgia Institute of Technology
aafshar8@gatech.edu

Joyce C. Ho
Emory University
joyce.c.ho@emory.edu

William K. Cheung
Hong Kong Baptist University
william@comp.hkbu.edu.hk

Chao Zhang
Georgia Institute of Technology
chaozhang@gatech.edu

Jimeng Sun
University of Illinois
Urbana-Champaign
jimeng@illinois.edu

ABSTRACT

Binary data with one-class missing values are ubiquitous in real-world applications. They can be represented by irregular tensors with varying sizes in one dimension, where value one means presence of a feature while zero means unknown (i.e., either presence or absence of a feature). Learning accurate low-rank approximations from such binary irregular tensors is a challenging task. However, none of the existing models developed for factorizing irregular tensors take the missing values into account, and they assume Gaussian distributions, resulting in a distribution mismatch when applied to binary data. In this paper, we propose Logistic PARAFAC2 (LogPar) by modeling the binary irregular tensor with Bernoulli distribution parameterized by an underlying real-valued tensor. Then we approximate the underlying tensor with a positive-unlabeled learning loss function to account for the missing values. We also incorporate uniqueness and temporal smoothness regularization to enhance the interpretability. Extensive experiments using large-scale real-world datasets show that LogPar outperforms all baselines in both irregular tensor completion and downstream predictive tasks. For the irregular tensor completion, LogPar achieves up to 26% relative improvement compared to the best baseline. Besides, LogPar obtains relative improvement of 13.2% for heart failure prediction and 14% for mortality prediction on average compared to the state-of-the-art PARAFAC2 models.

CCS CONCEPTS

• **Applied computing** → **Health informatics**; • **Computing methodologies** → **Factorization methods**; • **Information systems** → **Data mining**.

KEYWORDS

tensor factorization; PARAFAC2 factorization; binary tensor completion; computational phenotyping

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403213

ACM Reference Format:
Kejing Yin, Ardavan Afshar, Joyce C. Ho, William K. Cheung, Chao Zhang, and Jimeng Sun. 2020. LogPar: Logistic PARAFAC2 Factorization for Temporal Binary Data with Missing Values. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3394486.3403213

1 INTRODUCTION

Partially observed binary (*a.k.a.* one-class) data naturally arises in many real-world machine learning and data mining applications including event logs, transaction histories and patient records [13, 27], and they can often be represented by a *temporal binary irregular tensor* [2, 3, 28], *i.e.*, a set of binary matrices with one of their dimensions having the same size, while the other one varying between subjects. Such data usually comprises positive and unobserved entries with values of one and zero, respectively. However, the explicit negative entries are not recorded, meaning that the entries with value of zero could be either missing positives or real zeros. A typical example is the temporal electronic health records (EHR) [2]: the data of a particular patient can be represented by a binary matrix, where the entries of each row with value of one indicates the presence of the clinical features, *e.g.,* a confirmed disease diagnosis, recorded during a clinical visit. However, the zero entries do not always indicate absence of the diseases: but it just indicates that a specific disease diagnosis is not performed so the disease diagnosis is unknown. Other examples emerge in a variety of real-world applications, including collaborative filtering [27], spatio-temporal data modeling [1], recommender systems [15, 30], to name a few.

The **temporal irregularity** and the **absence of explicit negative observations** pose fundamental challenges in learning accurate low-rank approximations from such data. The positive unlabeled (PU) learning [7] was developed for binary classification tasks where the labels, instead of input features, follow the above missingness pattern. The PU learning framework was later extended to the matrix completion problem to handle the data of one single matrix [13], yet suffering severe overfitting problem [19]. In parallel, the PARAFAC2 factorization was developed as a practical solution to modeling the irregular tensors but does not handle one-class missing problem. Fig. 1 illustrates an application of PARAFAC2 for phenotype discovery from EHR [2, 29]. The extension of PARAFAC2 for binary input with one-class missing is difficult due to several challenges:

Distribution Mismatch. Existing PARAFAC2 models mostly minimize a reconstruction error defined by the square loss between the input and its reconstructions [29], which implies a Gaussian distribution of the reconstruction error. This is not ideal for binary input due to the mismatch of distribution [12] which could lead to suboptimal performance.

One-class missing data. Although missing data is in general inevitable, none of the existing PARAFAC2 models take the missing data into account. They implicitly assume that the data are fully observed in that their loss function is minimized over all entries, regardless of whether the entry is missing or not, leading to heavy inaccuracy in the factorization results. In our case, the missing values are concentrated in one-class which lead to additional challenges for the algorithms.

Temporal Irregularity. It is often desirable to learn factors that evolve smoothly over time to improve the interpretability. Existing method of decomposing the temporal factors as linear combinations of smooth basis functions [2] to impose such temporal smoothness relies on the square loss objective function, and is very sensitive to the number of basis functions.

To tackle these challenges, we propose the Logistic PARAFAC2 (LogPar) model, where the binary irregular tensor is assumed to be generated by Bernoulli distributions parameterized by a latent non-negative real-valued tensor, which is approximated with a non-negative PARAFAC2 factorization. Since the input tensor is either positive or unknown, we extend the positive-unlabeled (PU) learning method originally developed for classification problems [7] to the factorization model. We also introduce an effective uniqueness regularization and propose a time-aware temporal variation regularization for smoothing the temporal factor. We evaluated the proposed framework using three EHR datasets. Extensive experiments demonstrate that our proposed LogPar is superior in terms of both irregular tensor completion and downstream predictive tasks by outperforming all baselines consistently with a large margin. LogPar is also more robust against heavy missingness, and the ablation study also confirms the effectiveness of the regularization we incorporated.

2 BACKGROUND

In this section, we provide necessary background for developing our proposed model, including irregular tensor and its PARAFAC2 factorization. We also review some related work on binary matrix completion. We summarize the notations used in the paper in Table 1.

2.1 Irregular Tensor and PARAFAC2

Irregular Tensor. As shown in Fig. 1, an irregular tensor $\mathcal{X}$ comprises a set of K matrices $\{\mathbf{X}_k \in \mathbb{R}^{I_k \times J}\}_{k=1}^{K}$, where each *subject* is indexed with k [20]. A typical application of the irregular tensor is to describe the *temporal data*, where the matrices are composed of the same set of J features, but each subject may have distinct temporal length, denoted by I_k. We define the ℓ_1 norm and the inner product of irregular tensors as follows.

Definition 1 (ℓ_1 Norm). *The ℓ_1 norm of an irregular tensors is the sum of the ℓ_1 norm of its composing slices:* $\|\mathcal{X}\|_1 = \sum_{k=1}^{K} \|\mathbf{X}_k\|_1$.

Table 1: Notations used throughout the paper.

Symbol	Definition
$\mathcal{X}, \mathbf{X}, \mathbf{x}, x$	Tensor, matrix, vector, scalar
$\mathcal{X}$	The observed binary tensor with missing values
$\mathcal{Y}$	The unobserved ground-truth tensor
$\mathcal{M}$	The underlying real-valued tensor
Ω	The index set of positive entries in $\mathcal{X}$
$\widehat{\mathcal{M}}$, $\widehat{\mathcal{X}}$	The reconstruction of $\mathcal{M}$ and $\mathcal{X}$
$\mathbf{U}_k \in \mathbb{R}^{I_k \times R}$	The temporal factor matrix for the k^{th} subject
$\mathbf{s}_k \in \mathbb{R}^{R}$	The weighting vector for the k^{th} subject
$\mathbf{V} \in \mathbb{R}^{J \times R}$	The latent factor matrix for the features
I_k	The temporal length of the k^{th} subject
K, J	Number of subjects and features
R	Number of target rank
$\langle \cdot, \cdot \rangle$	The inner product
$\sigma(\cdot)$	The quantization probability function
$\mathcal{L}(x)$	The log likelihood of the observation
ℓ	The non-negative PU loss function

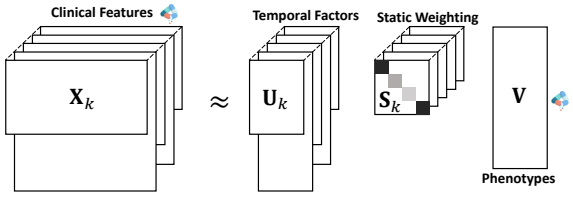


Figure 1: PARAFAC2 model for computational phenotyping: The input is a collection of binary matrices, with each of them corresponding to a patient. They have the same number of columns representing diseases, but different numbers of rows representing clinical visits. Value 1s in those matrices indicate confirmation of disease while value 0 means either the absence of the disease or missing diagnosis.

Definition 2 (Inner Product). *The inner product between two irregular tensors with the same size is given by:*

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{k=1}^{K} \sum_{i=1}^{I_k} \sum_{j=1}^{J} x_{k,i,j} * y_{k,i,j},$$

where $x_{k,i,j}$ is the (i,j)-th entry of the k^{th} slice $\mathbf{X}_k$.

PARAFAC2 Factorization. PARAFAC2 is a variant of tensor CP factorization [20] that applies to an irregular tensor by allowing the temporal factor to vary between subjects, namely each slice of the irregular tensor are mapped to a distinct temporal factor matrix. Fig. 1 illustrates the PARAFAC2 model [9]. Formally, the PARAFAC2 model solves the following optimization problem:

$$\begin{aligned} \underset{\{\mathbf{U}_k\},\{\mathbf{S}_k\},\mathbf{V}}{\arg\min} \quad & \sum_{k=1}^{K} \frac{1}{2} \left\| \mathbf{X}_k - \mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top \right\|_F^2 \\ \text{s.t.} \quad & \mathbf{U}_k = \mathbf{Q}_k \mathbf{H},\ \mathbf{Q}_k^\top \mathbf{Q}_k = \mathbf{I} \quad k = 1, \ldots, K, \end{aligned} \tag{1}$$

where $\mathbf{V} \in \mathbb{R}^{J \times R}$ is the latent factor matrix for the features, $\mathbf{U}_k \in \mathbb{R}^{I_k \times R}$ is the temporal factor matrix for the k^{th} subject, $\mathbf{S}_k \in \mathbb{R}^{R \times R}$

is a time-independent diagonal matrix capturing the overall weighting of each latent factor for the k^{th} subject, and R is the target rank. The constraint is imposed to ensure the uniqueness of the solutions, where $\mathbf{Q}_k \in \mathbb{R}^{I_k \times R}$ is column-wisely orthogonal and $\mathbf{H} \in \mathbb{R}^{R \times R}$ is a constant [16]. To enable this model to be applied to large-scale datasets, efficient algorithms have been developed for sparse input [29], and extended with more constraints to further enhance the interpretability, *e.g.*, the smoothness on $\mathbf{U}_k$ and the sparsity on $\mathbf{V}$ [2]. However, neither its application to binary input data, nor any extension to handling missing data has ever been studied for PARAFAC2 model.

2.2 Low-Rank Completion of Binary Matrix

Low-rank approximation is a principled framework to solve matrix and tensor completion problems for non-binary data [8, 23, 32, 37]. Specifically, a low-rank model can be fitted with only the observed entries, and then the unobserved ones can be estimated with the learned low-rank model. "1–bit matrix completion"[6] is an application of this framework, where the input 1-bit matrix contains both explicit positive and negative entries (with values of +1 and -1 respectively), and the zero entries are known to be missing. Thus a low-rank model can be fitted using only the observed entries.

Despite the promising results, it is in fact impossible to apply such a framework to our setting where the observations only contain a subset of the positive entries. In other words, no explicit negative entries (with value of -1) are observable. In such a setting, fitting a low-rank model over observed entries is infeasible, as obviously a rank-one solution with all entries being one is a trivial optimum.

The setting most relevant to ours is the one adopted by Sindhwani et al. [33], Yu et al. [36] and Hsieh et al. [13], where the observed entries contain only those with value 1s. The first one [33] developed a weighted non-negative matrix factorization by imposing different weights for the reconstruction error over the observed and the unobserved entries, and the second one [36] focused on sampling the zero entries as negative observations. The third one [13] adopted the PU learning strategy originally used for classification tasks and derived the unbiased PU learning loss function for the binary matrix factorization problem with superior performance demonstrated. With that being said, it is non-trivial to be extended to the PARAFAC2 framework. For instance, it relies on constraining the nuclear norm of the input matrix to enforce low-rankness. However, the nuclear norm for an irregular tensor is not well-defined. Besides, as pointed in [19], the formulation used by Hsieh et al. [13] could suffer from heavy overfitting, which is also empirically confirmed via our extensive experiments.

To the best of our knowledge, none of the prior works have investigated the tensor completion problem for the irregular tensors, and none of them have studied the PARAFAC2 factorization for binary data.

3 PROPOSED METHOD

3.1 Logistic PARAFAC2 Factorization

3.1.1 ***Observation model.*** We first introduce an observation model for the binary irregular tensors that accounts for the generation process underlying the binary data with missing values. Given a real-valued underlying irregular tensor $\mathcal{M}$, and a differentiable function $\sigma : \mathbb{R} \to [0, 1]$, we assume that the entries of the *hidden ground-truth tensor* $\mathcal{Y}$ are given as follows:

$$y_{k,i,j} = \begin{cases} 1 & \text{with probability } \sigma(m_{k,i,j}), \\ 0 & \text{with probability } 1 - \sigma(m_{k,i,j}), \end{cases} \tag{2}$$

where $\sigma(\cdot)$ is called the *quantization probability function* (QPF) [6] that maps each entry of $\mathcal{M}$ to a probability score between zero and one.

Due to the absence of explicit negative observations and the presence of missing values, we only partially observe the positive entries of $\mathcal{Y}$. In particular, we define the index set of the entries with value of one as $\Omega = \{(k, i, j) | x_{k,i,j} = 1\}$. The entries of the finally observed tensor $\mathcal{X}$ is thus given by:

$$x_{k,i,j} = \begin{cases} 1 & (k, i, j) \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The observation model defined above appears to be similar to, yet is essentially different from that defined by Davenport et al. [6], in that they assume explicit negative observations and the observations are sampled from both the positive and negative entries, whereas we consider the absence of only negative observations.

3.1.2 ***Formulation.*** Given the partial observation $\mathcal{X}$, we aim to learn its PARAFAC2 factorization. Different from the existing works [2, 29], we compute a low-rank factorization of the underlying tensor $\mathcal{M}$ instead of the $\mathcal{X}$ itself. Specifically, the k^{th} slice of the underlying tensor is approximated by:

$$\mathbf{M}_k \approx \widehat{\mathbf{M}}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top, \tag{4}$$

where $\mathbf{U}_k \in \mathbb{R}^{I_k \times R}$ is the temporal factor matrix for the k^{th} subject, $\mathbf{S}_k \in \mathbb{R}^{R \times R}$ is a diagonal matrix, and $\mathbf{V} \in \mathbb{R}^{J \times R}$ is the factor matrix for the features which is shared across all the subjects.

To ensure the interpretability of the model, we impose non-negativity constraint on all the factor matrices, namely $\mathbf{U}_k \; \forall k$, $\mathbf{S}_k \; \forall k$, and $\mathbf{V}$, leading to an additive model which is widely recognized as highly interpretable [21]. The reconstruction of the underlying tensor $\widehat{\mathcal{M}}$ is also non-negative; thus, we use a specific logistic function defined on the non-negative real numbers as the QPF, as shown below:

$$\sigma(m) = \frac{2}{1 + e^{-\gamma m}} - 1 \quad (m \geq 0), \tag{5}$$

where γ is a hyper-parameter controlling the steepness of the logistic curve.

Without considering the missingness, *i.e.*, assuming that we directly observe $\mathcal{Y}$, we can estimate the factor matrices by maximizing the log likelihood of the observation by treating all the unobserved entries as true zeros, leading to the following optimization problem:

$$\underset{\{\mathbf{U}_k\}, \{\mathbf{S}_k\}, \mathbf{V}}{\arg\min} \; \mathcal{L}(\widehat{\mathcal{X}}) \equiv \sum_{k=1}^{K} \sum_{i=1}^{I_k} \sum_{j=1}^{J} \ell(\widehat{x}_{k,i,j}, \; y_{k,i,j}) \tag{6}$$

$$\begin{aligned} \text{s.t.} \quad & \widehat{\mathbf{X}}_k = \sigma\left(\mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top\right), \\ & \mathbf{U}_1^\top \mathbf{U}_1 = \cdots = \mathbf{U}_K^\top \mathbf{U}_K = \mathbf{\Phi}, \\ & \mathbf{U}_k \geq \mathbf{0}, \; \mathbf{S}_k \geq \mathbf{0} \quad k = 1, \ldots, K, \\ & \mathbf{V} \geq \mathbf{0}, \end{aligned} \tag{7}$$

where $\ell(\widehat{x}_{k,i,j},\ y_{k,i,j})$ denotes the element-wise loss function, *i.e.*, the negative log likelihood of the observed entry $y_{k,i,j}$ parameterized by the reconstruction $\widehat{x}_{k,i,j}$, given by:

$$\ell(\widehat{x}_{k,i,j}, x_{k,i,j}) = \left(x_{k,i,j} \log \widehat{x}_{k,i,j} + (1 - x_{k,i,j}) \log(1 - \widehat{x}_{k,i,j})\right), \quad (8)$$

where we assume that $0 \log(\mu) = 0$ for all $\mu \geq 0$ to ease the notations [4].

3.2 Non-negative Positive-Unlabeled Loss

In practice, missing data is ubiquitous and estimating the factors by directly fitting the logistic PARAFAC2 (6) with partially observed input $\mathcal{X}$ could cause inevitable errors in that it does not account for the missing values in the unobserved data. Hsieh et al. [13] proposed the "unbiased PU learning" (uPU) for matrix completion by analogizing the missingness to the "noisy label" in classification problems [25], leading to the following objective function:

$$\hat{\ell}\left(\hat{x}_{ij}, x_{ij}\right) = \begin{cases} \frac{\ell'(\hat{x}_{ij},1) - \rho\ell'(\hat{x}_{ij},0)}{1-\rho} & \text{if } x_{ij} = 1 \\ \ell'(\hat{x}_{ij}, 0) & \text{if } x_{ij} = 0, \end{cases} \quad (9)$$

where $\ell'(t, y) = (t - y)^2$, $\mathbf{X}$ is the input matrix and $\widehat{\mathbf{X}}$ is the completion matrix, the variables to be solved for. ρ is a hyperparameter.

Direct application of the above uPU loss to our setting is challenging, because (a) the uPU loss is coupled with the mean square loss and its extension to the loss function Eq. (8) was never approached; (b) Eq. (9) is minimized subject to a nuclear norm constraint over $\widehat{\mathbf{X}}$ to enforce low-rankness, yet the nuclear norm of an irregular tensor is not well-defined; and (c) the unbiased PU learning is recently found to be less robust to overfitting [19] in classification tasks.

Inspired by the non-negative PU learning (nnPU) developed for classification tasks [19], we propose the following objective function:

$$\widetilde{\mathcal{L}}\left(\widehat{\mathcal{X}}\right) = \pi \frac{\langle \mathcal{X},\ \log(\widehat{\mathcal{X}})\rangle}{\|\mathcal{X}\|_1} + \max\left\{0,\ \frac{\langle 1 - \mathcal{X},\ \log(1 - \widehat{\mathcal{X}})\rangle}{\|1 - \mathcal{X}\|_1} - \pi \frac{\langle \mathcal{X},\ \log(1 - \widehat{\mathcal{X}})\rangle}{\|\mathcal{X}\|_1}\right\}, \quad (10)$$

where $\langle \mathcal{X},\ \log(\widehat{\mathcal{X}})\rangle$ computes the sum of the loss function Eq. (8) over the observed positive entries. The term inside the max operator stems from the unbiased PU learning with Eq. (8) as the pointwise loss function. However, it has been shown that when this term is negative, the estimation error bound of the unbiased PU learning is no longer tight [19]; thereby serious overfitting can occur. Therefore, the max operator is applied to ensure the non-negativity of Eq. (10). π is a hyperparameter, and $\|\cdot\|_1$ denotes the ℓ_1 norm of the irregular tensor.

3.3 Regularization

3.3.1 ***Uniqueness Regularization.*** Existing PARAFAC2 models tackle the uniqueness constraint, *i.e.*, $\mathbf{U}_k^\top \mathbf{U}_k = \mathbf{\Phi}\ (\forall k)$, by transforming them into a set of orthogonal Procrustes problems [2, 5, 16, 29, 31]. This approach implicitly requires that the temporal length of the data of each subject is larger than or equal to the rank, *i.e.*, $R \leq I_k\ \forall k$. Otherwise, the uniqueness of the solutions cannot be strictly guaranteed. However, in reality this requirement can be easily violated as a larger number of latent factors is in general desirable to accurately approximate large-scale real-world datasets. Moreover, the transformation into orthogonal Procrustes problems can only be carried out when the objective function is defined by the squared error; therefore, it cannot be applied to our objective function as defined in Eq. (10). In this paper, we regard $\mathbf{\Phi}$ as a variable and introduce a soft uniqueness constraint as follows:

$$\mathcal{R}_1 = \sum_{k=1}^{K} \frac{\mu}{2} \left\|\mathbf{U}_k{}^\top \mathbf{U}_k - \mathbf{\Phi}\right\|_F^2, \quad (11)$$

where $\mathbf{\Phi}$ is also a parameter to be learned.

3.3.2 ***Time-Aware Temporal Smoothing.*** Learning temporal factors that change smoothly over time is often desirable to improve the interpretability and alleviate the over-fitting to the missing data and the noise. Afshar et al. [2] incorporates the smoothness constraint by forcing the temporal factor $\mathbf{U}_k$ to be the linear combination of a set of temporal basis functions generated by M-spline. This method requires pre-computation of the spline functions; moreover, an efficient algorithm by projecting the input tensor can only be applied when the loss function is the mean square error (MSE), but not the logistic loss function as we developed in either Eq. (6) or Eq. (10). Instead, we propose a *time-aware temporal variation smoothness regularization*, formulated as follows:

$$\mathcal{R}_2 = \sum_{k=1}^{K} \sum_{i=2}^{I_k} e^{-\beta\delta_i} \left|\mathbf{u}_{k,t} - \mathbf{u}_{k,t-1}\right|, \quad (12)$$

where $\delta_i = t_i - t_{i-1}$ is the time gap between the i^{th} and its previous visit. We use an exponential term to adaptively weight the regularization based on the time gap between two visits with the intuition that steps closer in time generally should be closer in the latent space. β is a hyperparameter controlling the decay rate of the regularization strength over the time gap. Similar forms of regularization were also found promising in other applications, *e.g.*, image inpainting [22]. Yet few of them were ever extended to the temporal domain with the irregular time stamps being explicitly modeled.

3.4 Learning Algorithms

3.4.1 ***Optimization Problem.*** Given the observation $\mathcal{X}$, we aim at estimating the parameters of the PARAFAC2 factorization of its latent distribution by solving the following optimization problem:

$$\underset{\{\mathbf{U}_k\},\{\mathbf{S}_k\},\mathbf{V},\mathbf{\Phi}}{\arg\min} \quad \widetilde{\mathcal{L}} + \mu_1\mathcal{R}_1 + \mu_2\mathcal{R}_2 \quad (13)$$

$$\text{s.t.} \quad \mathbf{U}_k, \mathbf{S}_k \geq \mathbf{0} \quad \forall k, \quad (14)$$

$$\mathbf{V} \geq \mathbf{0}, \quad (15)$$

$$\|\mathbf{U}_k\|_\infty \leq \sqrt[3]{\alpha}, \quad \|\mathbf{S}_k\|_\infty \leq \sqrt[3]{\alpha} \quad \forall k, \quad (16)$$

$$\|\mathbf{V}\|_\infty \leq \sqrt[3]{\alpha}, \quad (17)$$

where μ_1 and μ_2 are the weightings of the two regularization terms. The infinity-norm constraints in Eq. (16-17) are imposed to enforce the underlying matrix $\widetilde{\mathcal{M}}$ not being too "spiky" and thus makes the recovery of the latent distributions well-posed [26].

3.4.2 ***Alternating Updates.*** We first alternate between the temporal matrix $\mathbf{U}_k$, the static vector $\mathbf{s}_k$, and the factor matrix $\mathbf{V}$, and update each parameter with others fixed using the gradient descent

based algorithm. Then, we update the uniqueness regularization parameter $\mathbf{\Phi}$ by minimizing Eq. (11), leading to the closed-form solution $\mathbf{\Phi} = \sum_k \mathbf{U}_k^\top \mathbf{U}_k / K$.

3.4.3 ***Mini-Batch Projected Stochastic Gradient Descent.*** To solve the optimization problem efficiently, we propose to use the optimization technique based on mini-batch stochastic gradient descent. However, the second term of the loss function in Eq. (10) cannot be decomposed point-wisely due to the max operator. Similar to the strategy used in Kiryo et al. [19], we observe that this term is in fact upper bounded by:

$$\Delta \leq \sum_{k=1}^{K} \max\left\{0, \frac{\langle 1 - \mathbf{X}_k,\ \log(1-\widehat{\mathbf{X}}_k)\rangle}{\|1-\mathbf{X}_k\|_1} - \pi \frac{\langle \mathbf{X}_k,\ \log(1-\widehat{\mathbf{X}}_k)\rangle}{\|\mathbf{X}_k\|_1}\right\}, \tag{18}$$

where Δ is the second term of the right-hand-side in Eq. (10). Therefore, we minimize this upper bound in each mini-batch instead.

We implemented the proposed method using PyTorch, and it can be efficiently trained end-to-end with GPU. Our implementation is publicly available at: https://github.com/jakeykj/LogPar.

3.5 Theoretical Analysis

We theoretically analyze the proposed model in terms of its capability of handling the missing values and recovering the underlying distribution generating the data. To ease the analysis, we focus on the loss function developed in Eq. (10), omitting the non-negative constraint and other regularization imposed for interpretation purposes, and use a standard sigmoid function as the QPF. We define the hypothesis class of LogPar as follows:

$$\mathcal{G} = \left\{\mathcal{M} \mid \mathbf{M}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top\ \forall k;\ \mathbf{U}_k \in \mathbb{R}^{I_k \times R}, \mathbf{S}_k \in \mathcal{S}^R, \mathbf{V} \in \mathbb{R}^{J \times R}\right\}, \tag{19}$$

where $\mathcal{S}^R = \{\mathbf{S} \mid \mathbf{S} = \text{diag}(\mathbf{s}),\ \mathbf{s} \in \mathbb{R}^R\}$.

It is difficult to directly bound the error of recovering $\mathcal{M}$ by solving (13). Therefore, we use the hidden ground-truth tensor $\mathcal{Y}$ as a bridge. Let $\mathcal{M}^* \in \mathcal{G}$ be the reconstruction of the minimizer of (6), and $\widetilde{\mathcal{M}} \in \mathcal{G}$ be that of (13). Let $\mathcal{L}$ denote the objective function defined as in Eq. (6), and let $\mathcal{L}_N^-(\mathcal{Z})$ be defined as: $\mathcal{L}_N^-(\mathcal{Z}) = \sum_{(i,j,k):y_{i,j,k}=0} \ell(z_{i,j,k}, 0)$, where ℓ is defined in Eq. (8).

We first show that the difference between the two solutions are bounded. Specifically, based on prior works on PU learning for classification tasks [19] and the generalization bound analysis technique for matrix completions [24], we have the following results.

Theorem 1. *Assume that* $\inf_{\mathcal{M}\in\mathcal{G}} \mathcal{L}_N^-(\sigma(\mathcal{M})) \geq \eta > 0$. *For any* $\delta > 0$, *with probability at least* $1-\delta$,

$$\mathcal{L}\big(\sigma(\widetilde{\mathcal{M}})\big) - \mathcal{L}\big(\sigma(\mathcal{M}^*)\big) \leq 16 L_\ell \pi \mathfrak{R}_\Omega(\mathcal{G}) + 8 L_\ell \mathfrak{R}_{\Omega^C}(\mathcal{G}) + 2C_\delta \left(\frac{2\pi}{\sqrt{|\Omega|}} + \frac{1}{\sqrt{|\Omega^C|}}\right) + 2C_l \pi \Psi,$$

where Ω^C *denotes the complementary of the index set* Ω, C_ℓ *is the upper-bound of* ℓ, L_ℓ *is the Lipschitz constant of* ℓ, *and* C_δ *is given by* $C_\delta = C_\ell \sqrt{\ln(1/\delta)/2}$. $\mathfrak{R}_\Omega(\mathcal{G})$ *is the empirical Rademacher complexity of* $\mathcal{G}$ *with respect to the index set* Ω, *defined by:*

$$\mathfrak{R}_\Omega(\mathcal{G}) = \frac{1}{|\Omega|}\mathbb{E}_\epsilon\left[\sup_{\mathcal{M}\in\mathcal{G}} \sum_{(i,j,k)\in\Omega} \epsilon_{i,j,k}\sigma(m_{i,j,k})\right],$$

where $\epsilon_{i,j,k}$*'s are independent random variables taking value 1 or -1 with probability 1/2.* Ψ *is given by* $\Psi = \exp(-2(\eta/C_\ell)^2/(\pi^2/\sqrt{|\Omega|} + 1/\sqrt{|\Omega^C|}))$.

Theorem 1 implies that given sufficiently large tensor, or sufficiently large number of subjects, $\mathcal{L}\big(\sigma(\widetilde{\mathcal{M}})\big) \to \mathcal{L}\big(\sigma(\mathcal{M}^*)\big)$. Furthermore, with the pointwise loss function ℓ defined as Eq. (8), we have $\widetilde{\mathcal{M}} \to \mathcal{M}^*$ [19].

Then we show that under mild conditions $\mathcal{M}^*$ recovers the underlying distribution generating the data by establishing the upper bound of the Hellinger distance $d_H^2(\sigma(\mathcal{M}^*), \sigma(\mathcal{M}))$. This step can be done by applying Theorem 6 from Davenport et al. [6]. The results are presented in Corollary 1 in the Appendix A.

4 EXPERIMENTS AND RESULTS

4.1 Datasets

We evaluate the proposed model using the following three large-scale datasets, two of which are publicly available.

(1) Sutter: This is a dataset collected from a large real-world health provider network, covering patients aged between 50 to 85 chosen for a heart failure (HF) study. We use the diagnoses and medications of each visit as the clinical features. We map the diagnosis codes to their third level of clinical classifications software (CCS)[1], and the medications to their fourth level of the Anatomical Therapeutic Chemical (ATC) classification system[2].

(2) CMS[3]: This is the publicly available CMS Linkable 2008-2010 Medicare Data Entrepreneurs' Synthetic Public Use File (DE-SynPUF), provided by the Centers for Medicare & Medicaid Services. We use the diagnoses of each clinical visit as the clinical features.

(3) MIMIC-III [14]: This is a large-scale, and de-identified ICU dataset which is publicly available, containing records related to more than forty thousand patients who stayed in the ICU at Beth Israel Deaconess Medical Center between 2001 and 2012. We use the medications and the abnormal laboratory tests as the clinical features.

The first two (Sutter and CMS) are longitudinal datasets, so we construct the irregular tensor based on the clinical visits, where the timestamp of each clinical visit is used to compute the temporal gaps between two visits. The latter one, MIMIC-III, contains data collected during the ICU stays of the patients. For MIMIC-III, we construct the irregular tensor on an eight-hour basis by accumulating the clinical features of every consecutive eight-hour time window; thus the temporal gaps between two steps are constant, whereas the number of time steps of each patients can be different. For all datasets, we extract patients with the number of temporal steps (I_k) between 20 and 100. We summarize the basic statistics of the four datasets in Table 1.

[1] https://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp
[2] https://www.whocc.no/atc/structure_and_principles/
[3] https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF

Table 2: Basic Statistics of the Datasets.

	Sutter	CMS	MIMIC-III
#Patients (K)	34,905	74,153	28,485
#Features (J)	328	319	405
Median(I_k)	26	26	22
Average(I_k)	30.5	29	28.4
#Positive entries	2.3M	4.5M	14.5M
Sparsity	0.80%	0.65%	4.43%
Single-feature visits	29.5%	37.4%	0.67%
Predictive task	Heart failure	–	Mortality
Positive label ratio	8.92%	–	8.86%

4.2 Baselines

We compare against the following baselines:

- **COPA**[4] [2], which is a state-of-the-art PARAFAC2 factorization model with a temporal smoothness constraint implemented by M-spline functions.
- **SPARTan**[5] [29], which is a PARAFAC2 factorization model developed for sparse input.
- **PU-MC**[6] [13], which is a matrix completion method based on PU learning. We matricize the irregular tensor by concatenating its slices along the time dimension to apply this model.
- **One-class MF (OCMF)**[7] [36], which is a state-of-the-art binary matrix completion method based on sampling zero entries as negative observations. We matricize the irregular tensor to run this baseline.

4.3 Tensor Completion

We are utmost interested in the quality of learned latent factors, however, they are difficult to be quantitatively evaluated due to lack of ground truth. Instead, we perform a tensor completion task. In general, the more accurate a model completes the unseen missing values, the better the latent factors explains the underlying patterns generating the data.

Evaluation Metric. Due to the binary nature of the tensor entries, and the imbalanced ratio between zeros and ones, we measure the PR-AUC (Area Under the Precision-Recall Curve) over the test subset to evaluate the completion performance.

4.3.1 ***Completion with varying target ranks.*** We first evaluate the completion performance with the target rank of the factorization model varying from 10 to 300. For evaluation purpose, we split the entries of the irregular tensor to training, validation and test set. Specifically, we extract 10% of the positive entries for hyperparameter tuning and hold out 20% of the positive entries for testing. For each positive entry, we randomly match ten negative entries to form the validation subset and test subset. Then we use the remaining 70% positive entries to construct the input irregular tensor. When sampling the validation and test subsets, we require that at least one positive entry remains in the training subset.

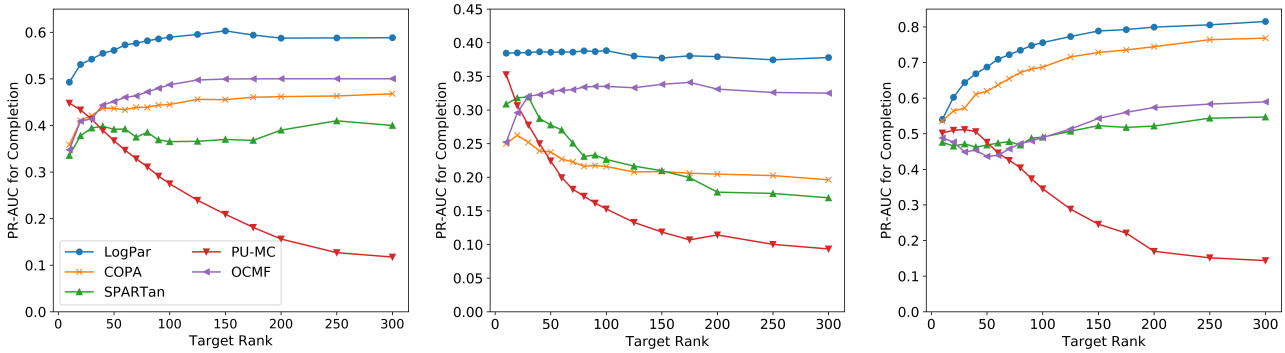
[4]available at: https://github.com/aafshar/COPA
[5]available at: https://github.com/kperros/SPARTan
[6]available at: http://www.cs.utexas.edu/~cjhsieh/biasMF_test.zip
[7]available at: https://www.csie.ntu.edu.tw/~cjlin/papers/one-class-mf/

Hyperparameter setting. We train LogPar using Adam [18], and the hyperparameter setting is summarized in Table A1 in the Appendix.

Results and Discussion. We visualize the performance of LogPar and the baselines for the three datasets in Fig. 2. The results show that LogPar outperforms all baselines consistently for all datasets. In particular, even with the smallest target rank of 10, LogPar obtains PR-AUC of 0.49 for Sutter, 0.38 for CMS, and 0.54 for MIMIC-III, achieving 9% and 8.5% relative improvement compared to the best baseline for Sutter and CMS, respectively. The completion performance of LogPar impressively increases with the target rank for the Sutter and the MIMIC-III datasets, with relative improvement of 19% and 51%, respectively, when comparing the target rank 300 and 10. This demonstrates that the expressive power of LogPar significantly enhances with increasing target rank. Yet it is noted that for CMS dataset, the improvement by increasing the target rank is negligible. This is due to the sparsity and large ratio of single-feature visits (37.4%), *i.e.,* rows with only one positive entry, in CMS dataset. Consequently, the latent factors tend not to capture the interactions between the clinical features. In other words, the factorization model overfits to the zero entries. Nevertheless, the completion performance of LogPar improves marginally from rank of 10 to rank of 100, and decreases only slightly after the target rank exceeding 100. In contrast, the completion performance obtained by all baselines for the CMS dataset decreases dramatically even when the target rank is smaller than 50. This clearly demonstrates that explicitly handling the missing values is of critical importance.

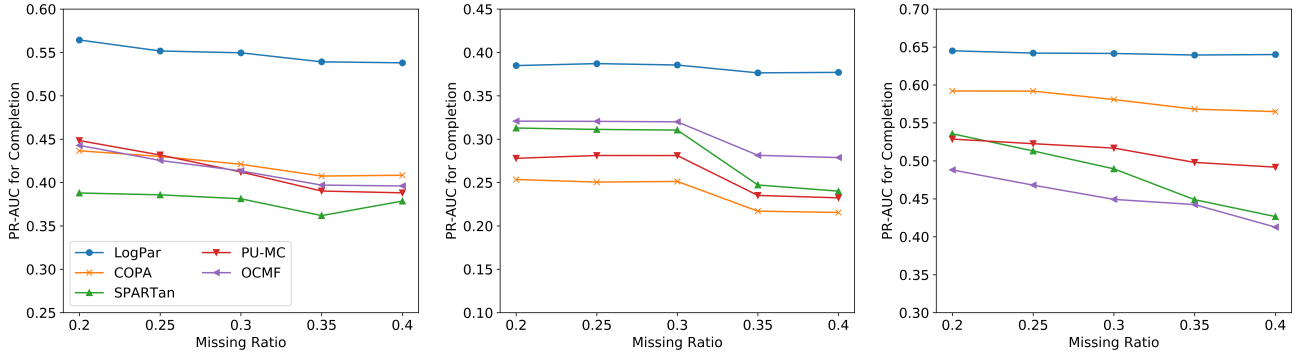
The completion performance of the PU-MC baseline decreases dramatically when the target rank increases for all datasets. The reason is twofold. First, the nuclear norm constraint for imposing the low-rankness is substituted with the matrix factorization during its optimization procedure [13]. Therefore, a smaller target rank actually imposes a more strict low-rankness regularization. Although a hyperparameter is available to control the Frobenius norm of the factor matrices to strengthen the low-rank regularization, tuning it does not improve the performance. Another equally important reason is that the "unbiased PU learning" technique used by PU-MC is known to be prone to overfitting [19]. With increasing target rank, PU-MC rapidly overfits to the missing values, resulting in a noticeable drop in its completion performance. When the target rank exceeds 50, the completion performance of SPARTan decreases marginally for the Sutter dataset, while that of COPA keeps increasing. This suggests that overfitting is incurred by SPARTan, but not by COPA. The major advance of COPA is its incorporation of a temporal smoothness by further decomposing the temporal factors into a set of smooth basis functions. This observation supports our motivation for incorporating the temporal smoothness regularization, and further analysis of this component is carried out in Section 4.5.

Moreover, by comparing the performance gap between LogPar and the best baseline across different datasets, we observe that the completion performance improvement is much more significant for the Sutter and the CMS datasets. This implies that LogPar is significantly more robust to sparse datasets, which is highly appealing in real-world applications due to the prevalence of such sparse datasets.



(a) Sutter Dataset (b) CMS Dataset (c) MIMIC-III Dataset

Figure 2: Tensor completion performance with different target ranks. PR-AUC is used as the evaluation metric as the tensors are binary. LogPar consistently outperforms all baselines for all datasets, and is more robust to overfitting when the target rank is large.



(a) Sutter Dataset (b) CMS Dataset (c) MIMIC-III Dataset

Figure 3: Tensor completion performance with different missing ratio. PR-AUC is used as the evaluation metric as the tensors are binary. LogPar consistently outperforms all baselines for all level of missingness and is robust to large missing rate.

4.3.2 ***Completion with varying missing ratio.*** We also empirically analyze the completion performance against different levels of missingness. We first sample 10% entries with value of one as validation set, and then vary the sampling ratio for the test set from 10% to 30% (total missing rate is from 20% to 40%), and use the remaining as the training subset. We fix the target rank of all models to be 30, and re-tune the hyperparameters for each missing rate as the statistic properties of the data change.

Fig. 3 shows the completion performance for different missing ratios. LogPar demonstrates superior robustness against heavy missingness in that its completion performance decreases by only 4.7%, 0.8% and 2% as the missing ratio increasing from 0.2 to 0.4 for Sutter, CMS and MIMIC-III datasets, respectively. In contrast, the best baselines, *i.e.,* COPA for Sutter, SPARTan for CMS and COPA for MIMIC-III, end up with a large decrease of 6.4%, 23.3% and 4.6%, respectively. The lack of explicit handling of the missing values is the main reason behind such dramatic performance drops for COPA and SPARTan. Although PU-MC models the missing values, its overfit-prone unbiased PU learning formulation makes it less robust against heavy missingness.

4.4 Downstream Predictive Tasks

The latent factors discovered by the low-rank factorization models can often be used as features for downstream predictive tasks. To evaluate the predictive performance of the latent factors discovered by LogPar, we perform two prediction tasks: the heart failure prediction for the PrimayCare dataset, and the mortality for the MIMIC-III dataset. We only compare the two PARAFAC2 baselines: COPA and SPARTan for predictive tasks. We first use the same procedure as described in Section 4.3.1 to hold out 10% positive entries for validation and another 20% for manually injecting missingness. We divide the patients into training set and test set and perform a five-fold cross validation. We train LogPar with the patients in the training set. Note that during the factorization step, no supervision information is available; thus we tune the parameters based on the completion performance over the validation entries. Then we fix the learned factor matrix ($\mathbf{V}$) and project the patients in the test data to the learned factor matrix. Similar to the existing works on PARAFAC2 model [28], we use the overall weighting vector, *i.e.*, the diagonal of $\mathbf{S}_k$, as the patient representation of the k^{th} patient, and

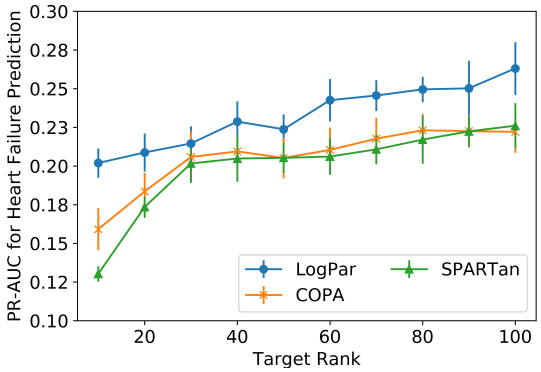


Figure 4: Performance of heart failure prediction using the Sutter dataset.

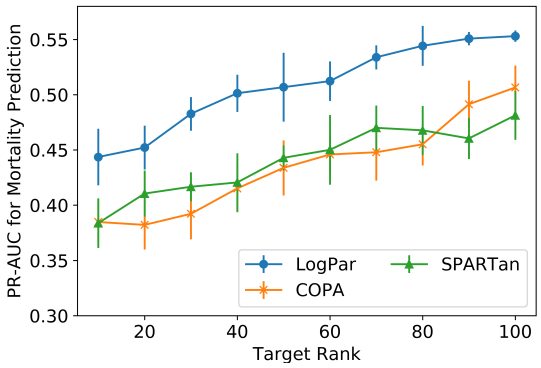


Figure 5: Performance of mortality prediction using the MIMIC-III dataset.

train a logistic regression model for prediction. We follow similar procedures for all baselines.

Evaluation Metric. Similar to evaluating the completion, we use the PR-AUC to evaluate the performance of the binary classification tasks. The PR-AUC is preferred over ROC-AUC because the datasets are imbalanced and the ratio of positive labels (case HF patient/deceased patients) is low, as shown in Table 2.

Results and Discussion. Fig. 4 and Fig. 5 show the predictive performance with different target ranks for the heart failure and mortality prediction tasks, respectively. LogPar outperforms the baselines consistently for all target ranks and both tasks. With the smallest target rank of 10, LogPar achieves 26.9% and 15.3% relative improvement over the best baseline for the heart failure and mortality prediction, respectively. As the target rank increases, the performance gap decreases marginally, yet LogPar still obtains an averaged relative improvement of 13.18% and 14% over all target ranks for the two prediction tasks. On the other hand, the predictive performance between the two state-of-the-art PARAFAC2 models, COPA and SPARTan, is marginal for all target ranks. The significant improvement of the predictive performance of LogPar comes from two aspects: First, it models the binary input with our developed logistic PARAFAC2, whereas the mean square objective function used by COPA and SPARTan does not well align with the binary data; thus the underlying factors generating the data are not well captured. Second, COPA and SPARTan do not account for the missing values in the input tensor, leading to inaccurate estimations of the latent factors. In contrast, LogPar adopts the non-negative PU learning technique to account for the missingness.

Table 3: The completion performance for the ablation study of LogPar in Sutter dataset, measured by PR-AUC. "Uni." and "Smth." are abbreviations for uniqueness regularization and temporal smoothness regularization, respectively.

	Model	Uni.	Smth.	R=10	R=30	R=50	R=70
1	COPA	✓	✓	0.36	0.43	0.44	0.44
2	LogPar (PN)	✓	×	0.45	0.50	0.51	0.56
3	LogPar (PU)	✓	×	0.48	0.52	0.55	0.57
4	LogPar (PU)	×	×	0.38	0.46	0.48	0.51
5	LogPar (PU)	✓	✓	0.50	0.55	0.57	0.58

4.5 Ablation Study

To further understand our model, we conduct an ablation study using the Sutter dataset to investigate the impact of each component to tensor completion performance.

Table 3 shows the results of one baseline, COPA, and our LogPar model with different combinations of loss function and the two regularization terms, where "PN" denotes the positive-negative loss function, *i.e.*, regarding all unobserved entries as true zeros, and applying Eq. (6) as the objective function. "PU" denotes the positive-unlabeled loss function as defined in Eq. (10). The comparison between the first two rows in Table 3 shows that the observation model and the distribution used in LogPar is of critical importance to accurate low-rank approximations for the binary input data. In the third row, we replace the loss function to the PU loss function developed in Eq. (10), and a relative improvement of around 6.7% is further achieved, demonstrating that our non-negative PU learning loss function is effective to handle the missing values.

We then examine the effectiveness of the uniqueness the temporal smoothness regularization. We switch off the both regularization terms and run LogPar to obtain the fourth row of Table 3, which surprisingly shows that the performance decrease by a large margin of more than 10% compared to that in the third row. This suggests that the uniqueness regularization is important for not only interpretation, but also accuracy. Comparing the third row and the fifth row, we can see that the temporal smoothness regularization can further improve the performance by around 5%.

To summarize, the correct distribution and the uniqueness regularization are the key factors to accurate low-rank approximation of binary irregular tensors. On top of that, the temporal smoothness regularization further improves the performance.

5 CONCLUSION

We present LogPar, a logistic PARAFAC2 model for learning low-rank factorization of binary irregular tensors. We assume that the binary input follow Bernoulli distributions parameterized by an underlying real-valued irregular tensor, and approximate the underlying tensor by a PARAFAC2 factorization. We introduced a

positive-unlabeled learning loss function to handle the one-class missingness and incorporated the uniqueness and temporal smoothness regularization to enhance the interpretability. We conducted extensive experiments using three large-scale datasets, and the results show that LogPar achieves better performance in binary irregular tensor completion and the downstream predictive tasks than the state-of-the-art PARAFAC2 and binary matrix completion models. The ablation study also confirmed the effectiveness of the regularization incorporated. For future work, we plan to extend LogPar to higher dimensions to capture the higher-order interaction between different modalities.

ACKNOWLEDGMENTS

This research is in part supported by General Research Fund RGC/HKBU12201219 and RGC/HKBU12202117 from the Research Grants Council of Hong Kong, the National Science Foundation award IIS-1418511, CCF-1533768, IIS-1838042 and IIS-1838200, the National Institute of Health award 1R01MD011682-01, R56HL138415 and 1K01LM012924-01.

REFERENCES

[1] Ardavan Afshar, Joyce C Ho, Bistra Dilkina, Ioakeim Perros, Elias B Khalil, Li Xiong, and Vaidy Sunderam. 2017. CP-ORTHO: An orthogonal tensor factorization framework for spatio-temporal data. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 1–4.

[2] Ardavan Afshar, Ioakeim Perros, Evangelos E Papalexakis, Elizabeth Searles, Joyce Ho, and Jimeng Sun. 2018. COPA: Constrained PARAFAC2 for sparse & large datasets. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 793–802.

[3] Ardavan Afshar, Ioakeim Perros, Haesun Park, Christopher deFilippi, Xiaowei Yan, Walter Stewart, Joyce Ho, and Jimeng Sun. 2020. TASTE: Temporal and static tensor factorization for phenotyping electronic health records. In *Proceedings of the ACM Conference on Health, Inference, and Learning*. 193–203.

[4] Eric C Chi and Tamara G Kolda. 2012. On tensors, sparsity, and nonnegative factorizations. *SIAM J. Matrix Anal. Appl.* 33, 4 (2012), 1272–1299.

[5] Jeremy E Cohen and Rasmus Bro. 2018. Nonnegative PARAFAC2: A flexible coupling approach. In *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 89–98.

[6] Mark A Davenport, Yaniv Plan, Ewout Van Den Berg, and Mary Wootters. 2014. 1-Bit matrix completion. *Information and Inference: A Journal of the IMA* 3, 3 (2014), 189–223.

[7] Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. 2014. Analysis of learning from positive and unlabeled data. In *Advances in Neural Information Processing Systems*. 703–711.

[8] Xiawei Guo, Quanming Yao, and James Tin-Yau Kwok. 2017. Efficient sparse low-rank tensor completion using the Frank-Wolfe algorithm. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[9] Richard A Harshman. 1972. PARAFAC2: Mathematical and technical notes. *UCLA working papers in phonetics* 22, 3044 (1972), 122215.

[10] Joyce C Ho, Joydeep Ghosh, Steve R Steinhubl, Walter F Stewart, Joshua C Denny, Bradley A Malin, and Jimeng Sun. 2014. Limestone: High-throughput candidate phenotype generation via tensor factorization. *Journal of biomedical informatics* 52 (2014), 199–211.

[11] Joyce C Ho, Joydeep Ghosh, and Jimeng Sun. 2014. Marble: High-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 115–124.

[12] David Hong, Tamara G Kolda, and Jed A Duersch. 2020. Generalized canonical polyadic tensor decomposition. *SIAM Rev.* 62, 1 (2020), 133–163.

[13] Cho-Jui Hsieh, Nagarajan Natarajan, and Inderjit S Dhillon. 2015. PU learning for matrix completion. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*. JMLR. org, 2445–2453.

[14] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data* 3 (2016), 160035.

[15] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. 2010. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*. 79–86.

[16] Henk AL Kiers, Jos MF Ten Berge, and Rasmus Bro. 1999. PARAFAC2—Part I. A direct fitting algorithm for the PARAFAC2 model. *Journal of Chemometrics: A Journal of the Chemometrics Society* 13, 3-4 (1999), 275–294.

[17] Yejin Kim, Robert El-Kareh, Jimeng Sun, Hwanjo Yu, and Xiaoqian Jiang. 2017. Discriminative and Distinct Phenotyping by Constrained Tensor Factorization. *Scientific Reports* 7, 1 (2017), 1114.

[18] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations (ICLR)*.

[19] Ryuichi Kiryo, Gang Niu, Marthinus C du Plessis, and Masashi Sugiyama. 2017. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems*. 1675–1685.

[20] Tamara G Kolda and Brett W Bader. 2009. Tensor decompositions and applications. *SIAM Rev.* 51, 3 (2009), 455–500.

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

[22] Xutao Li, Yunming Ye, and Xiaofei Xu. 2017. Low-rank tensor completion with total variation for visual data inpainting. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[23] Hanpeng Liu, Yaguang Li, Michael Tsang, and Yan Liu. 2019. CoSTCo: A Neural Tensor Completion Model for Sparse Tensors. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 324–334.

[24] Ken-ichiro Moridomi, Kohei Hatano, and Eiji Takimoto. 2018. Tighter generalization bounds for matrix completion via factorization into constrained matrices. *IEICE TRANSACTIONS on Information and Systems* 101, 8 (2018), 1997–2004.

[25] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. 2013. Learning with noisy labels. In *Advances in Neural Information Processing Systems*. 1196–1204.

[26] Sahand Negahban and Martin J Wainwright. 2012. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research* 13, May (2012), 1665–1697.

[27] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 502–511.

[28] Ioakeim Perros, Evangelos E Papalexakis, Richard Vuduc, Elizabeth Searles, and Jimeng Sun. 2019. Temporal phenotyping of medically complex children via PARAFAC2 tensor factorization. *Journal of Biomedical Informatics* 93 (2019).

[29] Ioakeim Perros, Evangelos E Papalexakis, Fei Wang, Richard Vuduc, Elizabeth Searles, Michael Thompson, and Jimeng Sun. 2017. SPARTan: Scalable PARAFAC2 for large & sparse data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 375–384.

[30] Steffen Rendle and Lars Schmidt-Thieme. 2010. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. 81–90.

[31] Marie Roald, Suchita Bhinge, Chunying Jia, Vince Calhoun, Tülay Adalı, and Evrim Acar. 2019. Tracing Network Evolution Using the PARAFAC2 Model. *arXiv preprint arXiv:1911.02926* (2019).

[32] Qiquan Shi, Haiping Lu, and Yiu-Ming Cheung. 2017. Rank-one matrix completion with automatic rank estimation via L1-norm regularization. *IEEE Transactions on Neural Networks and Learning Systems* 29, 10 (2017), 4744–4757.

[33] Vikas Sindhwani, Serhat S Bucak, Jianying Hu, and Aleksandra Mojsilovic. 2010. One-class matrix completion with low-density factorizations. In *2010 IEEE International Conference on Data Mining*. IEEE, 1055–1060.

[34] Kejing Yin, William K Cheung, Yang Liu, Benjamin C. M. Fung, and Jonathan Poon. 2018. Joint learning of phenotypes and diagnosis-medication correspondence via hidden interaction tensor factorization. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*. AAAI Press, 3627–3633.

[35] Kejing Yin, Dong Qian, William K. Cheung, Benjamin C. M. Fung, and Jonathan Poon. 2019. Learning phenotypes and dynamic patient representations via RNN regularized collective non-negative tensor factorization. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*. AAAI Press, 1246–1253.

[36] Hsiang-Fu Yu, Mikhail Bilenko, and Chih-Jen Lin. 2017. Selection of negative samples for one-class matrix factorization. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 363–371.

[37] Pan Zhou, Canyi Lu, Zhouchen Lin, and Chao Zhang. 2017. Tensor factorization for low-rank tensor completion. *IEEE Transactions on Image Processing* 27, 3 (2017), 1152–1163.

APPENDICES

A THEORETICAL ANALYSIS

A.1 Recovery of the latent distribution

Let $\mathcal{M}$ be the actual latent tensor generating the observations. We then seek to bound the difference between the distributions $\sigma(\mathcal{M})$ and $\sigma(\mathcal{M}^*)$. To ease the analysis, we first matricize $\mathcal{M}$ and $\mathcal{M}^*$ to $\mathbf{M}$ and $\mathbf{M}^*$ respectively, by concatenating each of their slices along the temporal dimension. $\sigma(\mathcal{M}^*)$ then can be regarded as a maximizer of the likelihood of the hidden ground-truth $\mathcal{Y}$. We can directly apply Theorem 6 from Davenport et al. [6] by defining the probability of sampling all entries to be constant 1, and obtain the following corollary.

Corollary 1. *Assume that* $\text{rank}(\mathbf{M}) \le R$, $\|\mathbf{M}\|_* \le \alpha\sqrt{R\sum_k I_k J}$ *and* $\|\mathbf{M}\|_\infty \le \zeta$. *Let* L_ζ *be defined as:*

$$L_\zeta = \sup_{|x|\le\zeta} \frac{|\sigma'(x)|}{\sigma(x)(1-\sigma(x))},$$

with probability at least $1 - C_1/(\sum_k I_k + J)$,

$$d_H^2(\sigma(\mathcal{M}^*), \sigma(\mathcal{M})) \le C_2 L_\zeta \alpha \sqrt{\frac{R}{\sum_k I_k J}} \sqrt{1 + \frac{(\sum_k I_k + J)\log(\sum_k I_k J)}{\sum_k I_k J}}, \quad (20)$$

where C_1 *and* C_2 *are absolute constants.*

A.2 Proof of Theorem 1

The remaining is to prove Theorem 1. Kiryo et al. [19] has established the consistency of the nnPU loss function for the binary classification problems. We extend their analysis to the PARAFAC2 model. We use the following lemma to complete the proof:

Lemma 1. *[19, Lemma 5] Assume that (1)* $\inf_{h\in H} R_n^-(h) \ge \eta > 0$*; (2)* $\mathcal{H}$ *is closed under negation. Then, for any* $\delta > 0$*, with probability at least* $1-\delta$*,*

$$\sup_{h\in\mathcal{H}} \left|\widetilde{R}_{\text{pu}}(h) - R(h)\right| \le 8L_f\pi_{\text{p}}\mathfrak{R}_{n_{\text{p}},p_{\text{p}}}(\mathcal{H}) + 4L_f\mathfrak{R}_{n_{\text{u}},p}(\mathcal{H}) + C'_\delta\cdot\chi_{n_{\text{p}},n_{\text{u}}} + C_f\pi_{\text{p}}\exp\left(\frac{-2(\eta/C_f)^2}{\pi_{\text{p}}^2/\sqrt{n_{\text{p}}} + 1/\sqrt{n_{\text{u}}}}\right),$$

where $\chi_{n_{\text{p}},n_{\text{u}}} = 2\pi_{\text{p}}/\sqrt{n_{\text{p}}} + 1/\sqrt{n_{\text{u}}}$, n_{p} *and* n_{u} *are the number of positive and unlabeled data, respectively.* π_{p} *is the class prior for the positive data.* $\mathcal{H}$ *is the classifier function class,* $h\in\mathcal{H}$ *is a classifier,* $R_n^-(g)$ *is the empirical loss of the negative samples evaluated with the negative label,* $\mathfrak{R}_{n_{\text{p}},p_{\text{p}}}(\mathcal{H})$ *and* $\mathfrak{R}_{n_{\text{u}},p}(\mathcal{H})$ *are the Rademacher complexities of* $\mathcal{H}$ *for the sampling of size* n_{p} *from the distribution of positive data* $p_{\text{p}}(x)$ *and of size* n_u *from the data distribution* $p(x)$*, respectively.* $\widetilde{R}_{\text{pu}}(h)$ *is the empirical loss obtained by the nnPU loss function for classification and* $R(h)$ *is that obtained by the positive-negative loss function* f. L_f *is the Lipschitz constant of* f.

The above lemma is developed in the context of PU learning for binary classification, and here we show how this lemma can be mapped to the tensor factorization setting. First, the loss function ℓ defined in Eq. (8) is Lipschitz continuous with Lipschitz constant L_ℓ. With the infinity norm constraints imposed on the factor matrices, ℓ is upper-bounded. Therefore, for the loss function ℓ and our hypothesis class $\mathcal{G}$ defined in Eq. (19), we obtain the following corollary for PARAFAC2 factorization by applying Lemma 1.

Corollary 2. *Let* ℓ *be defined as in Eq. (8), and the hypothesis classG be defined as in Eq. (19). Assume that* $\inf_{\mathcal{M}\in\mathcal{G}} \mathcal{L}_N^-(\sigma(\mathcal{M})) \ge \eta > 0$*. For any* $\delta > 0$*, with probability at least* $1-\delta$*,*

$$\sup_{\mathcal{M}\in\mathcal{G}} \left|\widetilde{\mathcal{L}}(\sigma(\mathcal{M})) - \mathcal{L}(\sigma(\mathcal{M}))\right| \le 8L_\ell\pi\mathfrak{R}_\Omega(\mathcal{G}) + 4L_\ell\mathfrak{R}_{\Omega^C}(\mathcal{G}) + C_\delta\left(\frac{2\pi}{\sqrt{|\Omega|}} + \frac{1}{\sqrt{|\Omega^C|}}\right) + C_l\pi\Psi,$$

where Ω^C *is the complementary of* Ω, $\Psi = \exp(-2(\eta/C_\ell)^2/(\pi^2/\sqrt{|\Omega|} + 1/\sqrt{|\Omega^C|}))$. $\mathfrak{R}_\Omega(\mathcal{G})$ *is the empirical Rademacher complexity of* $\mathcal{G}$ *with respect to the index set* Ω*, defined by:*

$$\mathfrak{R}_\Omega(\mathcal{G}) = \frac{1}{|\Omega|}\mathbb{E}_\epsilon\left[\sup_{\mathcal{M}\in\mathcal{G}}\sum_{(i,j,k)\in\Omega}\epsilon_{i,j,k}\sigma(m_{i,j,k})\right],$$

where $\epsilon_{i,j,k}$*'s are independent random variables taking value 1 or -1 with probability 1/2.*

With this corollary, we can prove Theorem 1 as follows.

Proof of Theorem 1.

$$\begin{aligned}
&\mathcal{L}(\sigma(\widetilde{\mathcal{M}})) - \mathcal{L}(\sigma(\mathcal{M}^*)) \\
&= \left(\widetilde{\mathcal{L}}(\sigma(\widetilde{\mathcal{M}})) - \widetilde{\mathcal{L}}(\sigma(\mathcal{M}^*))\right) + \left(\mathcal{L}(\sigma(\widetilde{\mathcal{M}})) - \widetilde{\mathcal{L}}(\sigma(\widetilde{\mathcal{M}}))\right) \\
&\quad + \left(\widetilde{\mathcal{L}}(\sigma(\mathcal{M}^*)) - \mathcal{L}(\sigma(\mathcal{M}^*))\right) \\
&\le 0 + 2\sup_{\mathcal{M}\in\mathcal{G}}\left|\widetilde{\mathcal{L}}(\sigma(\mathcal{M})) - \mathcal{L}(\sigma(\mathcal{M}))\right| \\
&\le 16L_\ell\pi\mathfrak{R}_\Omega(\mathcal{G}) + 8L_\ell\mathfrak{R}_{\Omega^C}(\mathcal{G}) + 2C_\delta\left(\frac{2\pi}{\sqrt{|\Omega|}} + \frac{1}{\sqrt{|\Omega^C|}}\right) + 2C_l\pi\Psi,
\end{aligned}$$

□

B HYPERPARAMETER SETTING

We summarize the hyperparameter setting of LogPar used in the experiments in Table A1. μ_1 and μ_2 are the weighing parameters for the uniqueness constraint and the temporal smoothness constraint, respectively. π is the class prior of the positive observations in Eq. (10), α is the parameter controlling the upper-bound of the infinity norm of the factor matrices. β is the shape parameter of the temporal smoothness regularization, and γ is the shape parameter in the QPF. The hyperparameters of baselines are also carefully tuned by grid search.

C PHENOTYPE CASE STUDY

Extracting computational phenotypes from EHR has been identified as a fundamental task [10] and an important application of tensor factorization models [2, 11, 17, 28, 34, 35]. In particular, a computational phenotype refers to a clinically relevant and interpretable combination of clinical features, *e.g.*, diagnoses and medications. With the non-negative tensor factorization model, the latent factor matrix (*i.e.*, $\mathbf{V}$ in our model) can be interpreted as the definition of the phenotypes. Each column of $\mathbf{V}$ represents one phenotype defined by its elements with positive values.

Table A1: Hyperparameter setting for different datasets.

	Sutter	CMS	MIMIC-III
Learning Rate	1e-4	1e-3	1e-4
Batch Size	32	128	32
μ_1	0.001	0.001	0.001
μ_2	0.1	0.5	0.1
π	0.007	0.005	0.015
α	8	8	8
β	1	1	1
γ	1	1	1

Table A2: Three examples of the phenotypes extracted from the Sutter dataset. The weights inside the parentheses after the phenotype index is the logistic regression coefficient for predicting case patients for heart failure. "Dx" denotes for diagnoses and "Rx" denotes for medications.

Phenotype #16 (weight=5.46)
Dx_Cardiac dysrhythmias [106.]
Dx_Congestive heart failure; nonhypertensive [108.]
Dx_Phlebitis; thrombophlebitis and thromboembolism [118.]
Rx_Coumarin Anticoagulants
Rx_Beta Blockers Cardio-Selective
Rx_Direct Factor Xa Inhibitors
Phenotype #25 (weight=4.95)
Dx_Coronary atherosclerosis and other heart disease [101.]
Dx_Peripheral and visceral atherosclerosis [114.]
Dx_Hypopotassemia
Rx_Potassium
Rx_Platelet Aggregation Inhibitors
Rx_Alpha-Beta Blockers
Phenotype #4 (weight=-2.59)
Dx_Other ear and sense organ disorders [94.]
Dx_Other upper respiratory infections [126.]
Dx_Other upper respiratory disease [134.]
Rx_Nasal Steroids
Rx_Azithromycin
Rx_Glucocorticosteroids

We construct the irregular tensor following the same procedure as described in Section 4.3.1 using the Sutter dataset and run LogPar with target rank of 30. This number is selected because it achieves a good balance between performance and interpretation: the completion and prediction performance with target rank of 30 is acceptable and the clinical relevance of 30 phenotypes is feasible to be manually examined. Table A2 lists three examples of the phenotypes, which are chosen based on the coefficient of the logistic regression for heart failure prediction, shown inside the parentheses after the phenotype indices. The listed three examples have the largest absolute coefficients, where positive values indicate that patients having this phenotype are more likely to be diagnosed heart failure in the future.

Phenotypes #16 and #25 are two most important features for predicting heart failure onset. The first includes three cardiovascular diseases, one of which is in fact the heart failure diagnosis code, suggesting that a patient with this diagnosis code yet does not meet the heart failure onset criteria is highly likely to develop heart failure in the future. The medications are used for treating the diagnoses in clinical practice. Phenotype #25 describes a condition of atherosclerosis, which is in fact one of the leading causes of heart failure.[8] The diagnoses and medications discovered are highly relevant in clinical practice.

Phenotype #4 represents a clinical condition of sense organ disorder and infections, associated with nasal medications, antibiotics and anti-inflammatory drugs, which is a commonly appearing condition in primary care. In general, this condition is less relevant to heart failure, and thus its coefficient for heart failure prediction is negative.

[8] https://www.heart.org/en/health-topics/heart-failure/causes-and-risks-for-heart-failure/causes-of-heart-failure