PartialFibers: An efficient method for predicting Drug-Drug Interactions

Aysegul Bumin[1][0000-0001-5654-9439], Kejun Huang[1][0000-0002-6460-6365], and Tamer Kahveci[1][0000--0002-4403-8612]

University of Florida, Gainesville, FL, USA
{aysegul.bumin, kejun.huang, tkahveci}@ufl.edu

Abstract. Drug resistance is one of the fundamental challenges in modern medicine. Using combinations of drugs is an effective solution to counter drug resistance as is harder to develop resistance to multiple drugs simultaneously. Finding the correct dosage for each drug in the combination remains to be a challenging task. Testing all possible drug-drug combinations on various cell lines for different dosages in wet-lab experiments is infeasible since there are many combinations of drugs as well as their dosages yet the drugs and the cell lines are limited in availability and each wet-lab test is costly and time-consuming. Efficient and accurate in silico prediction methods are surely needed. Here we present a novel computational method, *PartialFibers*, to address this challenge. Unlike existing prediction methods *PartialFibers* takes advantage of the distribution of the missing drug-drug interactions and effectively predicts the dosage of a drug in the combination. Our results on real datasets demonstrate that *PartialFibers* is more flexible, scalable, and achieves higher accuracy in less time than the state of the art algorithms.

Keywords: drug-drug combination · drug resistance · tensor completion · fiber-based approach

1 Introduction

One of the fundamental challenges in modern medicine is that over time, many diseases develop resistance to drugs. For infectious diseases, this often happens as the microbial species evolve, and as a result, drug resistant species replace those that are not resistant [3]. For genetic disorders, such as cancer, drug resistance arises naturally as each cancer is a result of unique mutations for individuals, while drugs target the condition of an average patient belonging to a population of cancer patients of varying mutation types [25]. New mutations yield cancer cells, which may not fit into the average cancer profile, leading to drug resistant cancer cells [12].

The effectiveness of anti-cancer drugs is often measured by their effects on the growth and death of cancer cells relative to control samples. Two standard metrics for measuring drug potency and efficacy are the half-maximal inhibitory concentration (IC50) and the fraction of viable cells at the highest drug concentration (Emax) [11]. One way to combat drug resistance is to use combinations

of drugs [2,13,17]. Doing so aims to increase the efficacy of the drugs by taking advantage of drug-drug interactions, and thus yields an effect on the targeted disorder more than the effect of the individual drugs if they were used independently. This improvement in drug efficacy happens for several reasons. First, the drug combination reduces the chance of developing resistance, for when cancer cells are exposed to multiple drugs, it is more difficult for them to develop resistance to all of them at the same time. This is because, a synergic drug combination accesses context specific multi-target mechanisms, and makes it harder for a cancer cell to develop resistance to multiple mechanisms of action [14]. Secondly, it reduces the risk of toxicity side effects as synergistic combination often reduces the dosage needed for each drug [14,16,22].

There are several databases which maintain the interactions among drugs. Among them, GDSC^2 hosts the efficacy of approximately 2000 clinically relevant drug combinations. These combinations encompass 125 characterized breast, colorectal, and pancreatic cancer cell lines [13]. The synergy between drugs is rare and highly context-dependent. However, when the targeted drugs are combined, they are more likely to be synergistic [13]. At the heart of the drug-drug interaction identification problem lies the fact that trying all possible drug-drug combinations on various cell lines and finding the right dosage of the drug-drug combination sufficient to eliminate a targeted percentage of given cancer cell lines is not feasible. This is because there are many drug-drug combinations, let alone the amounts of cell lines to be tested for each drug. This challenge necessitates developing computational strategies for estimating the right drug-drug combinations to be effective on the targeted cell lines. Many prediction methods for single drug response are developed with efforts of National Cancer Institute (NCI) and AstraZeneca-Sanger Drug Combination Prediction Dialog for Reverse Engineering Assessments and Methods (DREAM) Challenge [7]. Some of them are based on matrix factorization [26], recommender system based on cell line similarities [23], drug/cell line network [27] or manifold learning [1]. As we include drug-drug combinations, the problem size increases, bringing further challenges. Existing methods for predicting drug-drug interactions consider this data as a set of independent samples [9,28]. The data available for drug-drug interactions however varies both in terms of quantity and distribution across different drugs as well as cell lines. This inherent nature of variability of drug-drug interaction data introduces substantial disadvantages to existing methods in terms of their accuracy and efficiency.

Our contributions. In this paper, we consider the problem of predicting drug-drug interactions. We develop a novel algorithm named *PartialFibers* to efficiently and accurately predict the response of a given cell line to a given pair of drugs. More specifically, assume that we are given an incomplete three dimensional dataset, where the first dimension corresponds to a drug (called *library drug*), the second dimension also corresponds to a drug called *anchor drug* (i.e., a drug with a specific dosage,) and the third dimension corresponds to cell lines. The value at each entry of this tensor denotes IC50 value of the library drug for that cell line when that library drug is combined with the corresponding

anchor drug. *We aim to estimate the missing values in this dataset accurately and efficiently.* Our method takes advantage of the topological properties of the three dimensional drug-drug interaction data. Fiber structure (i.e., the vector consisting of all values in the given 3D matrix, tensor, when the coordinates of two of the dimensions are fixed) is a common topological pattern in the missing data [5, 19, 24]. Drug-drug interaction data, however, does not have a uniform fiber structure as some values may be known and others may be missing in each fiber. We call such patterns *partial fibers.* Our method allows us to learn from multiple dimensions of the given data. Given 3D drug-drug interaction data, at each iteration, our algorithm randomly picks one of the three dimensions and learns from the partial fibers along that dimension. Our method provides a simultaneous update for all the samples along the same partial fiber. Our results demonstrate that *PartialFibers* efficiently and accurately predicts the drug responses for the cell lines and it shows superior performance as compared to state-of-the-art methods in terms of both accuracy and running time performance. It benefits from the intrinsic structure of the data for leading to an efficient per-iteration update performing a faster and more accurate imputation of drug-drug interactions.

2 Methods

In this section, we describe our *PartialFibers* method for efficient and accurate imputation of the missing drug-drug interactions data. In Section 2.1, we introduce our notations and the preliminary concepts of tensor completion needed to explain our algorithm. In Section 2.2, we describe our *PartialFibers* algorithm in detail.

2.1 Notation and preliminaries

Here, we introduce the notation that we use for describing the core concepts required to understand our algorithm.

Baseline tensor completion. The three-dimensional tensor completion problem aims to estimate the missing values in a given tensor, such that a nonempty strict subset of the entries of this tensor is known apriori. The objective here is to find three-factor matrices such that their inner product produces a tensor of the same size, and the values at the tensor locations corresponding to the apriori known values are as close as possible to the given known values. Once such a tensor is produced, this technique estimates the values generated in the positions corresponding to the missing values as the generated values at those positions of the new tensor.

Let us denote a three-dimensional tensor with I rows, J columns and K depth with $\boldsymbol{X} \in \mathbb{R}^{I \times J \times K}$, and the value of the entry in this tensor at index (i, j, k) with $x_{i,j,k}$. We denote the rank of this tensor as R ($R \leq \min\{I, J, K\}$ is a positive integer), and the three factor matrices as $\boldsymbol{A} \in \mathbb{R}^{I \times R}$, $\boldsymbol{B} \in \mathbb{R}^{J \times R}$, $\boldsymbol{C} \in \mathbb{R}^{K \times R}$. We denote the ith row of the factor matrix $\boldsymbol{A}$ with $\boldsymbol{a}_i$, the jth row of the factor matrix $\boldsymbol{B}$ with $\boldsymbol{b}_j$ and the kth row of the factor matrix $\boldsymbol{C}$ with $\boldsymbol{c}_k$. Similarly, we

represent the value at index (i,r) of the factor matrix $\boldsymbol{A}$ with $\boldsymbol{a}_{ir}$, the value at index (j,r) of the factor matrix $\boldsymbol{B}$ with $\boldsymbol{b}_{jr}$ and the value at index (k,r) of the factor matrix $\boldsymbol{C}$ with $\boldsymbol{c}_{kr}$. A nonempty subset of the entries in $\boldsymbol{X}$ are known. We denote the set of indices (i,j,k) for which the values of $x_{i,j,k}$ are available with $\boldsymbol{\Omega}$. Using the notation above, the traditional CANDECOMP/PARAFAC (CP) formulation of tensor completion [6,10] is

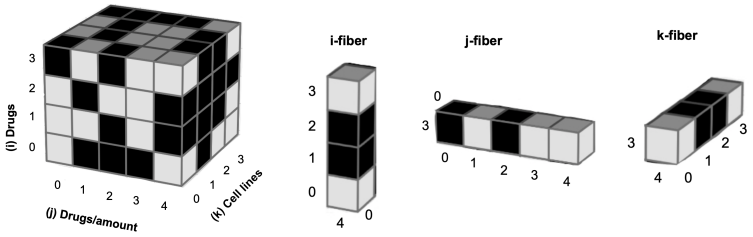
$$\underset{\boldsymbol{a}_i,\boldsymbol{b}_j,\boldsymbol{c}_k}{\text{minimize}} \sum_{(i,j,k)\in\Omega} \|x_{ijk} - \sum_r \boldsymbol{a}_{ir}\boldsymbol{b}_{jr}\boldsymbol{c}_{kr}\|^2. \tag{1}$$

In (1), we present the optimization problem to find the factor matrices $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$ that are going to be used to complete the missing entries in $\boldsymbol{X}$. Classic CP tensor completion algorithm iterates over the available tensor entries ($\boldsymbol{\Omega}$) one by one ($\forall(i,j,k)\in\boldsymbol{\Omega}$) and updates the corresponding entries of the three factor matrices $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$ [4]. Once the objective (1) converges, the complete data matrix can be constructed using the learned factor matrices.

Tensor completion with fibers. The distribution of the missing entries over the data tensor $\boldsymbol{X}$ is not always uniform across the three dimensions. This provides opportunities to benefit from the structure of the missing data. A common structure for the missing data is *fiber*. A fiber is the vector of all index values $(i,\ j,\ k)$ when the values of any two of these three variables are fixed. For instance, given the values of i and j, fiber is the vector of all the values along the kth dimension [15]. Thus, a fiber in a three dimensional tensor may consist of a varying index in any of the three dimensions. Recent research shows that the optimization function (1) can be computed more efficiently when the missing values of the tensor are along the same dimension (i.e., they make up a fiber) as multiple entries of the factor matrices can be computed in one iteration [5].

In this paper, we formulate the drug-drug interaction prediction problem, as the tensor completion problem. The fundamental challenge here is the tensor corresponding to this data does not follow the fiber structure we explain above. To solve this problem, we define a multi-dimensional notation which has more expressive power than the classic tensor notation. We use the "*" symbol to represent the dimension at which the fiber extends. For instance, we denote the fiber which has fixed values j and k along the last two dimensions and varies in the first dimension with $x_{*jk} = [x_{1,j,k}, x_{2,j,k}, \ldots, x_{I,j,k}]^T$. Similarly, we denote the fiber which has fixed values i and k along the first and third dimensions and varies in the second dimension with $x_{i*k} = [x_{i,1,k}, x_{i,2,k}, \ldots, x_{i,J,k}]^T$. Finally, we denote the fiber which has fixed values i and j along the first and second dimensions and varies in the third dimension with $x_{ij*} = [x_{i,j,1}, x_{i,j,2}, \ldots, x_{i,j,K}]^T$. Assume that, regardless of the direction, for all the fibers, within a fiber, either all entries are missing or all entries are available. Let us represent the Hadamard (element-wise) product with $\otimes$. Let us represent the set of all available fibers along the first dimension with $\boldsymbol{\Psi}_1$. Similarly, let us represent the set of all available fibers along the second and third dimensions with $\boldsymbol{\Psi}_2$ and $\boldsymbol{\Psi}_3$, respectively.

Using this notation, depending on the direction of the fiber, we reiterate the original fiber update formulation and rewrite (1) as follows. If the fiber is along the first dimension, we write



(a) 3D drug-drug interaction data

(b) Partial Fibers

Fig. 1. (a) Three dimensional tensor representation and the partial fibers. Rows, columns, and depth represent the library drugs, anchor drugs with amounts, and cell lines respectively. Entries with black color represent the available values, and the entries with gray color represent the missing values. (b) Examples of partial fibers, i-fiber, j-fiber and k-fiber. The i-fiber is a fiber where the j and k th dimensions are fixed and fiber is built using the dimension i. The same logic applies to the j-fiber and k-fiber.

$$\underset{\boldsymbol{A},\boldsymbol{b}_j,\boldsymbol{c}_k}{\text{minimize}} \sum_{(*,j,k)\in\Psi_1} \|\boldsymbol{x}_{*jk} - \boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)\|^2. \tag{2}$$

Similarly, we rewrite the optimization function based on the second dimension and the third dimension to update the factor matrices, $\boldsymbol{B}$ and $\boldsymbol{C}$. We update the factor matrices iteratively until the factor matrices represent the actual data with a small loss, i.e., until convergence.

Updating the factor matrices in fiber provides faster convergence than the classic CP tensor completion algorithm because the entire fiber can be updated simultaneously. However, this strategy works only if all the values in a given fiber are missing. Thus, it cannot be applied to solve the drug-drug interaction prediction problem considered in this paper, as this data does not have this characteristic.

2.2 Tensor completion with partial fibers

We first introduce the drug-drug interaction dataset and highlight the properties of the dataset. Then, we discuss our *PartialFibers* algorithm designed for drug-drug interaction dataset.

We use the drug-drug interaction data in the GDSC^2 dataset [13] for pairs of drugs. We refer to the first drug in a pair of drugs in this dataset as *library drug*. Unlike the library drug, this dataset associates the second drug with a specific dosage of that drug. We refer to the second drug at that dosage as *anchor drug*. For the given dosage of the anchor drug, this dataset provides the IC50 value of the library drug when the two drugs are applied on a given cell line. We represent this data using a three-dimensional tensor with I rows, J columns and K depth, $\boldsymbol{X} \in \mathbb{R}^{I\times J\times K}$. Row i, represents the ith library drug. Column j, represents the jth anchor drug along with its dosage. Depth k, represents the kth cell line. In summary, each entry $x_{i,j,k}$, denotes the dosage of the ith library drug required

Algorithm 1 *PartialFibers*, SGD for tensor completion using partial fibers

Input: $\boldsymbol{X}$ (the 3D incomplete data)

1: initialize $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$
2: **repeat**
3: Randomly select a dimension i, j, or k. (Assume that the dimension is i)
4: Randomly draw an i-fiber, $(*, j, k)$ from $\boldsymbol{\Psi}_{*,j,k}$
5: Initialize $\boldsymbol{D}$
6: Calculate the gradients
7: $\nabla_{\boldsymbol{A}} f_{*jk} = -(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k))(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)^\top$
8: $\nabla_{\boldsymbol{b}_j} f_{*jk} = -\boldsymbol{D}\boldsymbol{A}^\top(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)) \otimes \boldsymbol{c}_k$
9: $\nabla_{\boldsymbol{c}_k} f_{*jk} = -\boldsymbol{D}\boldsymbol{A}^\top(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)) \otimes \boldsymbol{b}_j$
10:
$$\begin{cases} \boldsymbol{D}\boldsymbol{A} \leftarrow \boldsymbol{D}\boldsymbol{A} + \gamma(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k))(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)^\top, \\ \boldsymbol{b}_j \leftarrow \boldsymbol{b}_j + \gamma \boldsymbol{D}\boldsymbol{A}^\top(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)) \otimes \boldsymbol{c}_k, \\ \boldsymbol{c}_k \leftarrow \boldsymbol{c}_k + \gamma \boldsymbol{D}\boldsymbol{A}^\top(\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)) \otimes \boldsymbol{b}_j. \end{cases}$$
11: **until** convergence

to eliminate 50% of the cell line k, when it is combined with the pre-defined dosage of anchor drug j. Figure 1 illustrates a representation of the drug-drug interaction data. Black and gray entries represent available and missing data, respectively.

We categorize fibers in two groups: *complete fiber* and *partial fiber*. We say that a fiber is *complete* if all the entries of that fiber are available or they are all missing. Otherwise, we call it a *partial fiber*. Orthogonal to this grouping, we further categorize fibers into 3 groups based on their directions; i-fibers, j-fibers, and k-fibers. I-fibers represent all library drugs for a fixed anchor drug/amount, j, and for a fixed cell line. For instance, in Figure 1(b), the i-fiber represents all the library drugs for the anchor drug/amount in index 4, and the cell line in index 0. Similarly, the j-fiber represents all the anchor drugs/amounts for the library drug in index 3, and the cell line in index 0. The k-fiber represents all the cell lines for the library drug in index 3, and the anchor drug/amount in index 4. In Figure 1(b), all three fibers are partial fibers since some values are available and some are missing in each. As the figure demonstrates, each fiber in the 3D representation may have a different number of missing entries. Also, the missing and present entries can be in any position as well as the (positive) number of positions.

We solve the optimization problem (1) using partial fiber updates. Algorithm 1 presents our *PartialFibers* method. The algorithm takes the incomplete tensor $\boldsymbol{X}$ as input. We start by randomly initializing the factor matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ (line 1). At each iteration, we randomly select a dimension among the three dimensions with equal probability (line 3). Upon selecting the dimension, we randomly choose one fiber among all the fibers in the selected direction with a uniform probability of being selected (line 4). In order to keep Algorithm 1 short, we write the subsequent steps assuming that we select the first dimension. However, it is important to note that any of the three dimensions can be chosen at each iteration. The subsequent steps still work the same by using the correct dimensions as indices in the formulation. Selecting the first dimen-

sion implies that the current fiber will be selected among all available i-fibers. Thus, a fiber along the first dimension for a randomly selected j, k is the vector $[x_{1jk}, x_{2jk}, \ldots, x_{Ijk}]^T$. Next, we construct an $I \times I$ diagonal matrix $\boldsymbol{D}$ as follows. For all indices $i \in 1, 2, \ldots, I$, if the value x_{ijk} in the given tensor is known (i.e., $(i, j, k) \in \Omega$), then we set the diagonal entry $\boldsymbol{D}[i, i] = 1$. We set all the remaining entries in $\boldsymbol{D}$ to 0 (line 5 of Algorithm 1). The purpose of this diagonal matrix is to mask the unavailable entries within a fiber and to focus on only the available entries. We represent the missing data with the underscore "_". As an example, let us assume that $I = 4$, and the first and third indices along dimension i are missing. We represent the (j,k)th fiber with $x_{*jk} = [\ _\ , x_{2,j,k}, \ _\ , x_{4,j,k}]$. The corresponding diagonal matrix $\boldsymbol{D} \in R^{4\times4}$ for this example is, $\boldsymbol{D} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.The next step is to perform parameter update, i.e. we need to update the indices of each factor matrix by using the information we learned from the available data. We use traditional stochastic gradient descent (SGD) as a baseline stochastic optimization algorithm [20]. Traditional SGD requires using one sample at a time. We redesign the traditional SGD to perform an efficient update by simultaneously learning from all the selected fiber entries. We design our SGD algorithm following from the observation that samples come in the form of $\boldsymbol{x}_{*jk}$ fibers, which involves the entire matrix $\boldsymbol{A}$ and the variables $\boldsymbol{b}_j$, $\boldsymbol{c}_k$. We represent the objective function for the (j,k)th fiber with f_{*jk} and rewrite the optimization function as; $\text{minimize}_{\boldsymbol{A},\boldsymbol{b}_j,\boldsymbol{c}_k} \quad \sum_{(*,j,k)\in\Psi_{*,j,k}} f_{*jk} = \|\boldsymbol{x}_{*jk} - \boldsymbol{D}\boldsymbol{A}(\boldsymbol{b}_j \otimes \boldsymbol{c}_k)\|^2$.We represent the gradient of the objective function f_{*jk} with respect to $\boldsymbol{A}$, $\boldsymbol{b}_j$, and $\boldsymbol{c}_k$ with $\nabla_{\boldsymbol{A}} f_{*jk}$, $\nabla_{\boldsymbol{b}_j} f_{*jk}$ and $\nabla_{\boldsymbol{c}_k} f_{*jk}$ respectively. The stochastic gradients are as in the lines 7, 8, 9 of Algorithm 1.

We represent the step size in SGD with γ. Then, the SGD algorithm takes the form as in line 10 of Algorithm 1. We repeat iterations (lines 2 to 11) until convergence, i.e. until the factor matrices are representative of the available data. The same convergence guarantees for the SGD apply for the partial fibers too, hence after convergence, we construct the complete data matrix using the learned factor matrices. Notice that it is also possible to adapt (2.2) to other stochastic gradient descent algorithms, such as Adam [18], Adagrad [8], and SPPA [4] through similar algebraic manipulations.

Advantages. There are three main advantages of partial fibers over the complete fiber setting:

– **Data flexibility.** If a specific entry in the tensor (i, j, k) is missing, the complete fiber approach requires at least one of the fibers $\boldsymbol{x}_{*jk}$, $\boldsymbol{x}_{i*k}$ or $\boldsymbol{x}_{ij*}$ to be complete. This is an extremely restrictive constraint which is often not observed in drug-drug interaction datasets. *PartialFibers* provides a flexible condition which works for any distribution of missing data. We allow the data to have a different number of available entries, the entries can be in any order along the same dimension, we benefit by grouping the missing items together and perform the partial fiber update efficiently.

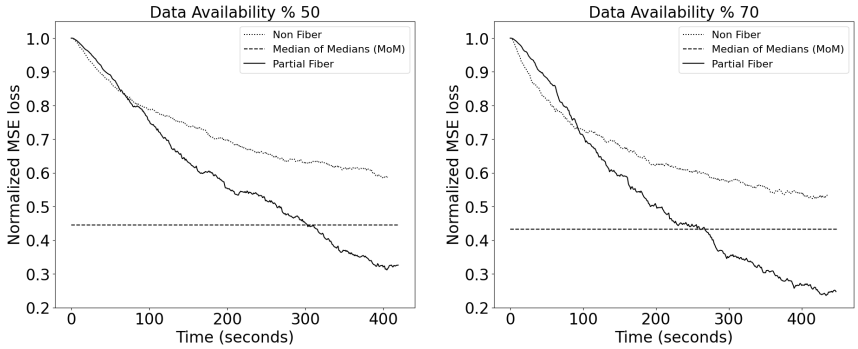
– **Algorithmic flexibility.** Complete fiber approach requires all fibers to be along the same dimension. *PartialFibers*, however, does not have this requirement. It selects the fiber dimension randomly with the same probability to benefit from every dimension equally.
– **Running time.** *PartialFibers* provides further improvement in running time as it performs a parameter update using all the samples within the same fiber. It allows these samples to be non-sequential and of any size. The more number of available items would mean more simultaneous updates, which results in further time improvement. In the worst case, there will be only one such training sample which is equivalent to considering each sample one by one. (Fully stochastic, non-fiber state).

3 Experimental Evaluation

Dataset. We use the $GDSC^2$ dataset which contains anchor drugs, library drugs, and their impact on a set of cell lines along with their IC50 values [13]. The dataset contains 160 anchor drugs, 65 library drugs, and 125 cell lines. Approximately 77% of the data is missing.
Experimental setup. We separate the available drug-drug interaction data into two disjoint sets: training and testing, for a given percentage $p \in (0 : 100)$ as $p\%$ and $(100 - p)\%$ of the available drug-drug interaction data, respectively. In our experiments, we select the p to be 85, and this designates 15% of the available data to be used for testing and the remaining for training. However, we do not use all of the remaining data for training. In order to test the potential of our method for different scenarios of data availability, we further sample from the 85% of the available data to cover $50\%, 60\%, 70\%, 80\%$ of the entire available data. For instance, for 50%, we only use a randomly selected subset of only 60% of the data (although more data is available, and hide the rest of the data from *PartialFibers*). To ensure the consistency in our experiments, as we grow the training dataset with more samples, we made sure that every anchor drug, library drug, and cell line pair in 50% is contained within 60%, every sample in 60% is contained within 70%, and every sample in 70% is contained within 80%.
Methods compared. We compare the prediction performance of *PartialFibers* with two state-of-the-art methods. These are non-fiber, well-known tensor completion algorithm [6,10] and Medians of Means (MoM) [21] approaches.
Other details. We implement all the algorithms in Python and perform the experiments on a Mac machine with 8-Core Intel Core i9 processors running at 2.3 GHz, and 16GBs RAM.

3.1 Advantage of partial fiber structure in loss

We start by evaluating how well *PartialFibers* predicts the missing values. We apply our *PartialFibers* algorithm, non-fiber and MoM algorithms separately to compute the missing values in drug-drug interactions dataset. Then, we calculate the normalized mean squared prediction error over all test samples with



(a) 50% Data Availability (b) 70% Data Availability

Fig. 2. Change of loss values over time (seconds) for different algorithms using different percentages of data availabilities. Figure 2(a), and Figure 2(b) represent the normalized mean squared loss values for the Partial Fiber, the Non Fiber and MoM Algorithms using 50%, and 70% of the available data respectively.

different percentages of data availability. Figure 2 presents the normalized mean squared loss values in the presence of different data availabilities. We observe that regardless of how much data is available, *PartialFibers* performs better than non-fiber [10] and MoM [21] algorithms. *PartialFibers* reaches to a smaller loss value than both methods in less than 400 seconds. Increasing the available data from 50% to 70%, we see that the normalized MSE loss value decreases the most for *PartialFibers* among the three methods. This implies that as partial fibers get denser (i.e., has more data points) our algorithm converges faster. We observe that the three compared algorithms benefit from the increasing data availabilities in different amounts. The converged loss value for *PartialFibers*, Non Fiber, and MoM [21] decreases by 24.2%, 15.80% and 1.86% respectively as we increase the available data percentage from 50% to 70%. We observe the smallest change in MoM. This implies that the newly added data do not have too much of an impact on the overall prediction error of MoM. We conjecture that this is because MoM selectively uses only subsets of the new data depending on the tensor entry to be predicted.

We hypothesize that the more data we have, the more simultaneous updates *PartialFibers* can provide. This implies that we expect that *PartialFibers* will converge faster with more data available. Figure 3 shows the convergence of our method when 50% and 80% of data is available. We observe that for both data availabilities, *PartialFibers* successfully achieves a small normalized mean squared loss value. This is because, as we provide more data, the number of indices that are updated simultaneously is expected to increase. This allows us to get a smaller loss value in less time. Our results support our hypothesis: Our method for 80% data availability is always below that for 50% in Figure 3. This means that we can reach the same loss value in less time by using more data.

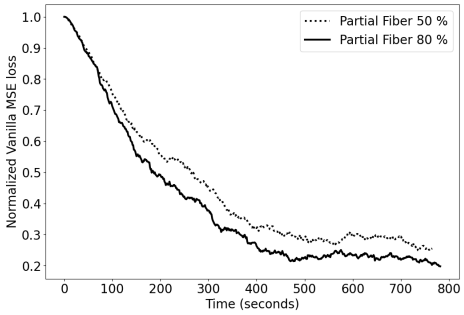
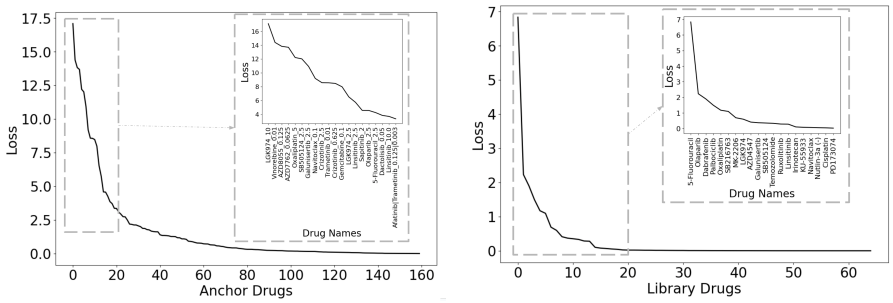


Fig. 3. The normalized mean squared error, loss values over time (seconds) for *PartialFibers* with different data availabilities. The solid line represents the *PartialFibers* with 80% data availability, and the dotted line represents the *PartialFibers* with 50% data availability.



(a) Anchor Drugs - Partial Fiber (b) Library Drugs- Partial Fiber

Fig. 4. The loss values for each anchor drug and each library drug after convergence and the sorted loss values with a focus on the top 20 drugs with the highest loss values. Figure 4(a) and Figure 4(b) represent the loss values for anchor drugs and library drugs after completion with *PartialFibers*.

3.2 Drugs which have the most/least accurate predictions

Here, we present our results that focus on the drugs which have the most/least accurate predictions. We calculate the individual loss values for each of the anchor drugs and library drugs, both with partial fiber setting and non-fiber setting. We sort the drugs based on their loss values and share the name of the drugs that have the highest loss in each setting. We share the detailed results in Figure 4. We observe that more than half ($\sim$ 20) of the anchor drugs, and more than half of the library drugs ($\sim$ 40) reach a very small loss value, in both fiber and non-fiber settings. This implies that both of the methods are efficiently predicting the drug response for most of the drugs. Our results demonstrate that loss values are smaller for the *PartialFibers* for both anchor and library drugs. When we focus on 20 drugs which have the highest loss, for anchor drugs we see that 12 out of 20 drugs are common for both *PartialFibers* and non-fiber

approaches, and if we narrow down and look at the top 10 drugs we see that 7 of them are same for both of the algorithms. The order might be different but for the same set of drugs we observe poor prediction performance, and we see that the loss is always less with *PartialFibers*. The common anchor drugs are as follows; Crizotinib, Trametinib, Oxaliplatin, SB505124, LGK974, AZD8055, Vinorelbine, AZD7762, 5-Fluorouracil, Navitoclax, Olaparib, Sapitinib. Each drug in this set targets different type of cancer. It is important to note the weaknesses before we can make a trustworthy data completion.

4 Conclusion

In this paper, we presented a novel algorithm called *PartialFibers* for predicting the drug dosage in drug-drug combination setting. Given partial information on the dosage of drug-drug combinations, *PartialFibers* effectively predicts the IC50 values for the missing entries by specifically benefiting from the structure of the data without further assumptions. We showed that *PartialFibers* is a time efficient, flexible, scalable computational method which can provide accurate predictions.

References

1. Ahmadi Moughari, F., Eslahchi, C.: Adrml: anticancer drug response prediction using manifold learning. Scientific reports **10**(1) (2020)
2. Al-Lazikani, B., Banerji, U., Workman, P.: Combinatorial drug therapy for cancer in the post-genomic era. Nature biotechnology **30**(7) (2012)
3. Ayukekbong, J.A., Ntemgwa, M., Atabe, A.N.: The threat of antimicrobial resistance in developing countries: causes and control strategies. Antimicrobial Resistance & Infection Control **6**(1) (2017)
4. Bumin, A., Huang, K.: Efficient implementation of stochastic proximal point algorithm for matrix and tensor completion. In: European Signal Processing Conference. IEEE (2021)
5. Bumin, A., Ritz, A., Slonim, D., Kahveci, T., Huang, K.: FiT: fiber-based tensor completion for drug repurposing. In: ACM International Conference on Bioinformatics (2022)
6. Carroll, J.D., Chang, J.J.: Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. Psychometrika **35**(3) (1970)
7. Costello, J.C., Heiser, L.M., Georgii, E., Gönen, M., Menden, M.P., Wang, N.J., Bansal, M., Ammad-Ud-Din, M., Hintsanen, P., Khan, S.A., et al.: A community effort to assess and improve drug sensitivity prediction algorithms. Nature biotechnology **32**(12) (2014)
8. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. Journal of machine learning research **12**(7) (2011)
9. Fan, K., Cheng, L., Li, L.: Artificial intelligence and machine learning methods in predicting anti-cancer drug combination effects. Briefings in bioinformatics **22**(6) (2021)
10. Harshman, R.A.: Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. UCLA Working Papers in Phonetics **16** (1970)

11. Havel, H., Finch, G., Strode, P., Wolfgang, M., Zale, S., Bobe, I., Youssoufian, H., Peterson, M., Liu, M.: Nanomedicines: from bench to bedside and beyond. The AAPS journal **18** (2016)
12. Housman, G., Byler, S., Heerboth, S., Lapinska, K., Longacre, M., Snyder, N., Sarkar, S.: Drug resistance in cancer: an overview. Cancers **6**(3) (2014)
13. Jaaks, P., Coker, E.A., Vis, D.J., Edwards, O., Carpenter, E.F., Leto, S.M., Dwane, L., Sassi, F., Lightfoot, H., Barthorpe, S., et al.: Effective drug combinations in breast, colon and pancreatic cancer cells. Nature **603**(7899) (2022)
14. Keith, C.T., Borisy, A.A., Stockwell, B.R.: Multicomponent therapeutics for networked systems. Nature reviews Drug discovery **4**(1) (2005)
15. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM review **51**(3) (2009)
16. Lehár, J., Krueger, A.S., Avery, W., Heilbut, A.M., Johansen, L.M., Price, E.R., Rickles, R.J., Short Iii, G.F., Staunton, J.E., Jin, X., et al.: Synergistic drug combinations tend to improve therapeutically relevant selectivity. Nature biotechnology **27**(7) (2009)
17. Lopez, J.S., Banerji, U.: Combine and conquer: challenges for targeted therapy combinations in early phase trials. Nature reviews Clinical oncology **14**(1) (2017)
18. Madani Tonekaboni, S.A., Soltan Ghoraie, L., Manem, V.S.K., Haibe-Kains, B.: Predictive approaches for drug combination discovery in cancer. Briefings in bioinformatics **19**(2) (2018)
19. Orekhov, V.Y., Ibraghimov, I., Billeter, M.: Optimizing resolution in multidimensional nmr by three-way decomposition. Journal of biomolecular NMR **27**(2) (2003)
20. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747 (2016)
21. Sapashnik, D., Newman, R., Pietras, C.M., Zhou, D., Devkota, K., Qu, F., Kofman, L., Boudreau, S., Fried, I., Slonim, D.K.: Cell-specific imputation of drug connectivity mapping with incomplete data. Plos one **18**(2) (2023)
22. Sharom, J.R., Bellows, D.S., Tyers, M.: From large networks to small molecules. Current opinion in chemical biology **8**(1) (2004)
23. Suphavilai, C., Bertrand, D., Nagarajan, N.: Predicting cancer drug response using a recommender system. Bioinformatics **34**(22) (2018)
24. Tomasi, G., Bro, R.: Parafac and missing values. Chemometrics and Intelligent Laboratory Systems **75**(2) (2005)
25. Vasan, N., Baselga, J., Hyman, D.M.: A view on drug resistance in cancer. Nature **575**(7782) (2019)
26. Wang, L., Li, X., Zhang, L., Gao, Q.: Improved anticancer drug response prediction in cell lines using matrix factorization with similarity regularization. BMC cancer **17**(1) (2017)
27. Wei, D., Liu, C., Zheng, X., Li, Y.: Comprehensive anticancer drug response prediction based on a simple cell line-drug complex network model. BMC bioinformatics **20**(1) (2019)
28. Wu, L., Wen, Y., Leng, D., Zhang, Q., Dai, C., Wang, Z., Liu, Z., Yan, B., Zhang, Y., Wang, J., et al.: Machine learning methods, databases and tools for drug combination prediction. Briefings in bioinformatics **23**(1) (2022)