FiT : Fiber-based tensor completion for drug repurposing

Aysegul Bumin
aysegul.bumin@ufl.edu
University of Florida
Gainesville, Florida, USA

Anna Ritz
aritz@reed.edu
Reed College
Portland, Oregon, USA

Donna Slonim
slonim@cs.tufts.edu
Tufts University
Medford, Massachusetts, USA

Tamer Kahveci
tkahveci@cise.ufl.edu
University of Florida
Gainesville, Florida, USA

Kejun Huang
kejun.huang@ufl.edu
University of Florida
Gainesville, Florida, USA

ABSTRACT

Drug repurposing aims to find new uses for existing drugs. One drug repurposing approach, called "Connectivity Mapping," links transcriptomic profiles of drugs to profiles characterizing disease states. However, experimentally evaluating the transcriptomic effects of drug exposure in particular cells is a costly process. Characterizing drug-cell combinations widely is further hindered because primary tissue samples may not be abundant, leading to many gaps in drug-cell databases. To best find drugs relevant for particular conditions, we may therefore want to impute the transcriptomic impact of a given drug on an unassayed cell type or types. This step deviates from classic data completion problems, however, because of the fundamental bottleneck that state of the art data imputation techniques for this problem do not consider the unique characteristics of the data. The missing values in the data are not randomly distributed, and the genes are not independent entities, but rather they interact with and affect the transcription rates of one another. Here, we address the first and one of the most fundamental parts of the connectivity map data imputation problem to enable drug repurposing. We develop a novel method, named FiT (Fiber-based Tensor Completion) to impute the transcription values for missing drug-cell line combinations in a highly sparse drug-cell line dataset accurately and efficiently, while exploiting the distribution of missing values as well as the interactions among genes. Our results demonstrate that even on a sparse dataset, where approximately 75% of the data is missing, FiT outperforms existing approaches and obtains more accurate results in a significantly shorter amount of time.

KEYWORDS

drug repositioning, drug connectivity mapping, integrative tensor completion

ACM Reference Format:
Aysegul Bumin, Anna Ritz, Donna Slonim, Tamer Kahveci, and Kejun Huang. 2022. FiT : Fiber-based tensor completion for drug repurposing . In *13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '22), August 7–10, 2022, Northbrook, IL, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3535508.3545527

BCB '22, August 7–10, 2022, Northbrook, IL, USA

ACM ISBN 978-1-4503-9386-7/22/08…$15.00
https://doi.org/10.1145/3535508.3545527

1 INTRODUCTION

Drug repositioning (repurposing) is the process of finding new uses for drugs or compounds already in development. Repurposing existing drugs for novel indications reduces drug development time and cost, and can decrease the risk of failure, as developing new drugs and compounds for different disease-affected cells with unique traits is costly and can take decades, making drug repurposing a necessity [32]. One promising way to address drug repurposing is to adopt *connectivity mapping* [21]. This is a technique which *maps* drugs to the diseases by comparing their impact on the expression values. Briefly, connectivity mapping links a drug to a particular disease if the drug reverses the disease's impact on the expression values of the genes. Thus, this strategy conjectures that drugs identified by the connectivity mapping have a high potential for therapeutic efficacy in diseases [21, 45, 47].

Transcriptional profiling of drug-exposed cells underlies connectivity mapping methods. The profiling technologies have long enabled surveillance of the expression levels of many genes under different conditions [13, 38, 43]. Formulating the drug repurposing problem as connectivity mapping however introduces unique challenges. This is because an existing drug is often tested on a small set of cell lines, yet what kind of transcriptional response the same drug triggers on many other cell types is not known as each such evaluation requires expensive wet-lab tests and most importantly requires access to limited physical resources (such as frozen tissue samples from the same tumor) which may be impossible to obtain. [6, 8, 35] As a result, the transcription values obtained through these techniques are often incomplete for a potentially large set of cell/drug combinations. It is of utmost importance to have reliable expression values since the rest of the process, connectivity mapping, and the drug repositioning steps rely on the transcriptional response of cells to drugs.

Drug repurposing requires imputing missing transcription values for cells when they are administered each of the given potentially large collection of drugs. One obvious way to approach this problem is to leverage classic data completion methods for high dimensional data. Some of the general expression imputation methods include k-nearest neighbors [43], local least squares optimization [18], and Bayesian prediction [28]. There are additional methods using time

series information present in the data [33], and collaborative filtering methods are also available for data imputation [34, 44].

Drug repurposing through transcriptome analysis requires studying data with unique characteristics. First, the dataset has a three-dimensional structure, where the dimensions indicate drugs, cells, and genes. Thus, we denote the data as a three-dimensional tensor $\boldsymbol{X}$; transcription value of the kth gene for the ith cell line, upon application of the jth drug is represented with $x_{i,j,k}$. Second, the distribution of the missing transcription values over this tensor is not uniform across the three dimensions. This is because if the experiment for the jth drug is missing for the ith cell line, the transcription values of all the genes for that configuration is missing. In other words, $x_{i,j,k}$ is missing for all values of k. Figure 1 illustrates this. Let us denote the total number of genes with g. Given a fixed value of i, and j, we use the term *fiber* to describe the vector of all the values $[x_{i,j,1}, x_{i,j,2}, \ldots, x_{i,j,g}]^T$ as this term is commonly used in the tensor studies [20]. Tensor completion problem has been considered in the literature to exploit the latent structure and to predict the missing values [23]. Tensor completion algorithms are specifically designed to perform well when the data available is sparse [37]. These methods however do not take into account the biased distribution of the missing information encountered in the transcriptome data for drug repurposing due to organization of missing parts of the tensor as fibers. This makes standard tensor completion inefficient for such datasets, especially as the dataset size grows. Moreover, biological data has various information hidden in terms of the relationship of entities, such as regulatory interactions among genes as well as their functional similarities, which may not be explicitly available in other types of data [7]. To the best of our knowledge, tensor completion has not fully been exploited for drug repurposing.

Within the context of estimating connectivity mapping, there are two main studies focusing specifically on the imputation of the missing transcription data. One of the studies imputes missing transcription values by combining the local and global information using k-nearest neighbor (KNN) and the tensor completion [16]. The other study shows certain advantages on using cell line specific approach along with Two-Way Algorithm (where expression values are averaged across drugs and cell lines), KNN, and singular value decomposition (SVD) [35]. However, there are two main problems with these imputation methods. First, their performance is highly dependent on the amount of missing data. Second, they are mainly designed to handle two-dimensional data (matrices) instead of three-dimensional data (tensors), which characterize the data used for drug repurposing. As we explain later in this paper in detail, these shortcomings makes existing methods ineffective for the problem considered in this paper.

Contributions. In this paper, we introduce a novel algorithm, called the FiT (**Fi**ber-based **T**ensor Completion) Algorithm, for predicting drug connectivity from three dimensional incomplete transcriptome data. Our method exploits the topological properties of the missing data and the interactions among genes to efficiently and accurately impute missing transcription values. We make three observations in developing FiT. 1. The missing data has a specific structure (i.e., they are organized as fibers), representing the presence or absence of entire experiments rather than representing random noise. 2. We have access to gene interaction data, which we refer as external information. 3. The genes can be grouped based on the similarities of their expression values for further improving the imputation performance. Using these three observations, we make three specific contributions.

Our first contribution is leverages on the structure of the missing data. We modify the tensor completion algorithm so that the algorithm benefits from the prior knowledge that the missing data is organized as fibers. This particular structure is not limited to the biology domain. Different fields like; spectroscopy, multidimensional nuclear magnetic resonance (NMR) analysis also experience data missing in fibers due to different reasons like machine failures, and sparse sampling frequencies [29, 42].

Our second contribution is to integrate the topology of interactions among the genes to tensor completion. Leveraging external information has already been considered in naive tensor completion algorithms and is shown to have higher imputation accuracy [1, 22, 27, 48]. Existing methods often use the external binary information (such as interactions between pairs of entities) to favor the predicted values of the interacting entities to be equal. This strategy however does not work for drug repurposing as interaction between two genes does not imply that they have same transcription values. We introduce a novel regularization term that promotes to use the external information as an indication of correlation between transcriptions of interacting genes.

Our third contribution is to cluster similar genes and perform tensor completion on the different clusters separately. We conjecture that clustering by genes will bring associated genes into the same sub-tensor and thus reduce noise in data imputation. The sub-tensors help to structure the data based on underlying biology, similarity of genes. We build the sub-tensors by clustering the genes using k-means clustering on their transcription values over all the drug and cell combinations and perform the tensor completion algorithm for each of these sub-tensors, and ultimately combine them back to one tensor that has the same order of genes as the initial data.

We compare the FiT Algorithm with four existing methods for their ability to predict drug connectivity. We showcase the performance of our algorithm using an incomplete cell-specific drug response data set [35]. We report three different evaluation metrics: a) accuracy of drug connectivity, b) difference between the imputed and actual values, and c) the correlation between the imputed and actual values. Our results demonstrate that the accuracy of drug connectivity generated by the values imputed by the FiT Algorithm is higher than other algorithms, especially for the cell lines which are not tested on many drugs. We also achieve higher correlation between real and imputed values using the FiT Algorithm. Moreover, using the FiT Algorithm, we get smaller cumulative difference between the imputed and actual values compared to other methods, resulting in predictions closer to the actual values. The high correlation and smaller distance between imputed values and the actual values allow us to successfully find the unassayed drugs that reverse gene expression signatures found in diseases. Lastly, we achieve these results in a significantly shorter time than other methods. FiT Algorithm exploits the inherent fiber structure of the

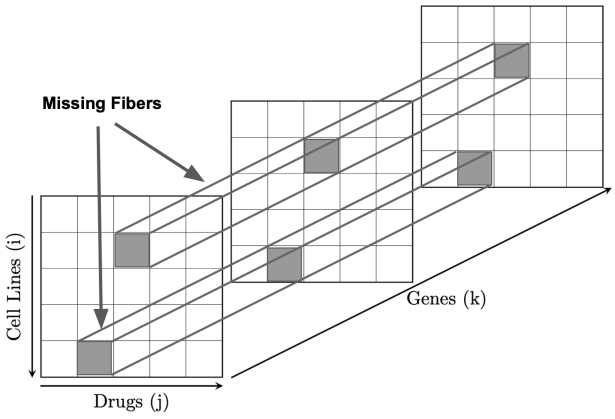


Figure 1: Representation of the 3D data tensor. Rows/columns/depth represent the cell lines/drugs, compounds/genes respectively. Shaded entries represent thee missing values. Missing entries span the entire depth (fibers) for subsets of row/column combinations.

missing data. This results in imputing the missing values at least five times faster than existing algorithms.[1,2]

2 METHODS

In this section, we describe our FiT method for efficient and accurate imputation of the missing data in the form of fiber. We describe the foundation of our algorithm which formulates the tensor completion with fiber updates in Section 2.1. We discuss how we integrate protein interaction networks in our method in Section 2.2. We finally explain how our algorithm takes advantage of clustering of sets of genes with similar transcription patterns in Section 2.3.

2.1 Tensor completion with fiber update

One of the defining characteristics of transcriptional drug repositioning is that the missing data has a unique topological structure: If the experimental results for a combination of cell line and drug are not available, then the transcription values are missing for the entire set of genes for that combination (see Figure 1). Our method exploits the prior knowledge that the missing transcription values are organized as fibers in a tensor.

The objective of three-dimensional tensor completion is to find three factor matrices such that their inner product gives the actual tensor. Let us represent the transcription data with a three-dimensional tensor with I rows, J columns and K depth, $\boldsymbol{X} \in \mathbb{R}^{I\times J\times K}$, and denote the transcription of the kth gene for the ith cell line, upon application of the jth drug with $x_{i,j,k}$. We define the rank as R, 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 $\boldsymbol{c}_{kr}$. We denote the domain of the entries (i, j, k) for which the values of $x_{i,j,k}$ are available with Ω. Using the notation above, we write the traditional formulation of tensor completion [4, 15] as

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

Let us denote the number of genes in the given dataset with g. We denote the (i, j)th fiber of the tensor $\boldsymbol{X}$ with $\boldsymbol{x}_{ij} \in \mathbb{R}^g$, and the Hadamard (element-wise) product with $\circledast$ symbol. Recall that for a given triplet (i, j, k), if $(i, j, k) \in \Omega$, then $\forall r, 1 \le r \le g$, we have $(i, j, r) \in \Omega$. Hence, we define another set Ψ that denotes the set of the indices (i, j) of the known fibers as the projection of the set Ω on the first two values. Using this notation, we formulate a minimization equation equivalent to Equation 1 as

$$\underset{\boldsymbol{a}_i,\boldsymbol{b}_j,\boldsymbol{C}}{\text{minimize}} \sum_{(i,j)\in\Psi} \|\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)\|^2. \tag{2}$$

We design a stochastic gradient descent (SGD) algorithm assuming the samples come in the form of $\boldsymbol{x}_{ij}$ fibers, which involves the variables $\boldsymbol{a}_i, \boldsymbol{b}_j$, and the entire matrix $\boldsymbol{C}$. We represent the objective function for the (i, j)th fiber with f_{ij} (i.e. $f_{ij} = \|\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)\|^2$), and represent the gradient of the objective function f_{ij} with respect to $\boldsymbol{a}_i$, $\boldsymbol{b}_j$, and $\boldsymbol{C}$ with $\nabla_{\boldsymbol{a}_i} f_{ij}$, $\nabla_{\boldsymbol{b}_j} f_{ij}$ and $\nabla_{\boldsymbol{C}} f_{ij}$ respectively. The stochastic gradients are

$$\begin{aligned}
\nabla_{\boldsymbol{a}_i} f_{ij} &= -\boldsymbol{C}^\top(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)) \circledast \boldsymbol{b}_j,\\
\nabla_{\boldsymbol{b}_j} f_{ij} &= -\boldsymbol{C}^\top(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)) \circledast \boldsymbol{a}_i,\\
\nabla_{\boldsymbol{C}} f_{ij} &= -(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j))(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)^\top.
\end{aligned}$$

Let us denote the step size in SGD with γ. The SGD algorithm takes the form

$$\begin{cases}
\boldsymbol{a}_i \leftarrow \boldsymbol{a}_i + \gamma \boldsymbol{C}^\top(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)) \circledast \boldsymbol{b}_j,\\
\boldsymbol{b}_j \leftarrow \boldsymbol{b}_j + \gamma \boldsymbol{C}^\top(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)) \circledast \boldsymbol{a}_i,\\
\boldsymbol{C} \leftarrow \boldsymbol{C} + \gamma(\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j))(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)^\top.
\end{cases}$$

Notice that it is possible to adapt Equation 2 to other stochastic gradient descent algorithms, such as Adam [19], Adagrad [25], and SPPA [2] through similar algebraic manipulations.

As we demonstrate later in Section 3.6, updating factor matrices by one fiber at a time instead of one tensor entry provides a significant time advantage since instead of iterating over every index of the fiber one by one, the fiber update does the same operation using the properties of matrix multiplication. There are two main benefits. First of all, the time it takes to perform the update for the entire fiber is significantly lower, because with fiber update we do not need to iterate over all the entries in a fiber one by one. Second, when there are many values within the fiber (for our envisioned application, fibers could easily be as long as the number of genes in the entire human genome, $\approx 20K$), it may not be computationally feasible to perform a non-fiber update, whereas the fiber update offers an efficient and feasible alternative.

2.2 Incorporating interactions into FiT

Genes affect the transcription levels of other genes through regulatory and other interactions [17]. Given this observation, we conjecture that using prior knowledge of interactions between

[1]This paper is partially funded by NSF under Award Number 2111679.
[2]We thank the T-Tripods institute (NSF grant 1934553) for inviting Tamer Kahveci and Anna Ritz to the workshop where we learned about the problem and commenced collaboration.

genes might improve the accuracy of transcriptomic imputation. We formulate this relationship with the hypothesis that the transcription values of interacting genes are correlated. Positive/negative correlation respectively indicates activation/suppression of their transcription.

We use protein-protein interaction data to represent the connectivity of the genes. More specifically, we use the STRING database [40], which provides interactions with confidence values and supporting evidence. The STRING database collects evidence from different resources (text mining of the literature, experimental data, computational interaction predictions using co-expression) and integrates this information, presenting a protein to protein interaction database that covers physical as well as functional associations [41]. We therefore filtered the protein-protein interactions to include those based on experimental evidence only, and we selected the score threshold to be greater than 900 out of 1,000, thus selecting only the highest confidence interactions. At this level of stringency, only 56 genes of the measured genes are connected, via 70 connections.

We incorporate the protein interactions into the optimization problem formulation provided in Equation 2 by adding each interaction as a within-mode regularization term, as discussed in [27]. Let us denote the Graph Laplacian matrix with $\boldsymbol{L}$, and the (m,n)th entry in $\boldsymbol{L}$ with $\boldsymbol{L}_{(m,n)}$. If there is an interaction between gene m and gene n and $m \neq n$ then $\boldsymbol{L}_{(m,n)} = -1$, and when $m = n$ then $\boldsymbol{L}_{(m,n)}$ is equal to the degree of m (i.e., number of edges connected to node n). We denote the regularization constants for the regularization terms of the parameters $\boldsymbol{a}_i, \boldsymbol{b}_j$ and $\boldsymbol{C}$ with λ_a, λ_b and λ_C respectively. We represent the trace function in linear algebra with $Tr(\dot{})$. The optimization problem becomes

$$\underset{\boldsymbol{a}_i,\boldsymbol{b}_j,\boldsymbol{C}}{\text{minimize}} \sum_{(i,j)\in\Psi} \|\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)\|^2 + \lambda_a\|\boldsymbol{a}_i\|^2 + \lambda_b\|\boldsymbol{b}_j\|^2 + \lambda_C \operatorname{Tr}(\boldsymbol{C}^T\boldsymbol{L}\boldsymbol{C}),$$

The within-mode regularization term motivates the two connected genes to have the same gene expression value. However, if the two genes are connected, they do not necessarily have the exact same expression value. Therefore, we can only say that if two genes are found to be connected via external sources, then their expression values are expected to be *related*.

The convention to include binary external information (such as absence/presence of interactions between two variable) to the objective function is by using regularization terms [27]. In the literature, often the regularization term minimizes the Euclidean distance between the values of two vectors for the connected entities. This formulation provides an equation for which we can obtain partial derivatives easily. This formulation however does not work for drug repurposing as the interaction between two genes does not indicate equality of their transcription values. It rather indicates their correlation. To express this, we replace the Euclidean distance with the cosine distance between the transcription values of these two genes. [14]. As we explain below, this formulation needs to be treated carefully as the partial derivatives no longer yields linear equations.

Recall that we represent the rank with R. We represent the size g vector with all entries equal to one with $\mathbf{1}$ and its transpose with $\mathbf{1}^T$ (i.e., $\mathbf{1} = [1, 1, \ldots, 1]^T$). We define matrix $\boldsymbol{D} \in \mathbb{R}^{g\times g}$ as a helper matrix where $\boldsymbol{D} = (\boldsymbol{I} - \frac{1}{R}\mathbf{1}\mathbf{1}^T)$. Note that, multiplying the matrix $\boldsymbol{D}$ with any vector $\boldsymbol{c} \in \mathbb{R}^{g\times R}$, we get the mean subtracted version of the vector $\boldsymbol{c}$. We represent the external network with the matrix $\tilde{\boldsymbol{A}}$. Recall that we represent the mth row of the factor matrix $\boldsymbol{C}$ as $\boldsymbol{c}_m$. In the problem definition, if there is an interaction between gene m and gene n and $m \neq n$ then the (m,n)th entry of the external network matrix $\tilde{\boldsymbol{A}}$ represents this interaction as $\tilde{\boldsymbol{A}}_{(m,n)} = \frac{1}{\|\boldsymbol{D}\boldsymbol{c}_m\|\|\boldsymbol{D}\boldsymbol{c}_n\|}$, and otherwise $\tilde{\boldsymbol{A}}_{(m,n)} = 0$.

The information that we want to capture is cosine distance (i.e., -cosine similarity) rather than Euclidean distance, hence we make some further changes in the problem formulation and present it as

$$\underset{\boldsymbol{a}_i,\boldsymbol{b}_j,\boldsymbol{C}}{\text{minimize}} \sum_{(i,j)\in\Psi} \|\boldsymbol{x}_{ij} - \boldsymbol{C}(\boldsymbol{a}_i \circledast \boldsymbol{b}_j)\|^2 + \lambda_a\|\boldsymbol{a}_i\|^2 + \lambda_b\|\boldsymbol{b}_j\|^2 - \lambda_C \operatorname{Tr}(\boldsymbol{D}\boldsymbol{C}^T\tilde{\boldsymbol{A}}\boldsymbol{C}\boldsymbol{D}). \quad (3)$$

By minimizing the negative of the cosine similarity, we aim to maximize the correlation between two genes if they are known to be connected in the external network. The external network can be built using any external information representing prior knowledge about the genes. It can also be applied to any domain where there is access to certain external information on the connectivity of the entities that represents their correlation. We present the related empirical results in the Experimental Results section.

2.3 Integration of gene clustering to FiT

Our final contribution follows from the conjecture that the transcription value of a gene introduces noise in imputing the transcription of another gene if the two genes exhibit significantly different behaviors. Inversely, genes with similar characteristics can yield more accurate imputations of each other. We mathematically formulate this conjecture by clustering genes with similar behaviors (i.e., similar transcription values in the training dataset). It is worth noting that other criteria can also be adopted to define clusters, such as the similarity of the gene sequences, or similarity of their known functions.

Let us assume that the data for p drug-cell line pairs are available in the training data. We represent each gene (say mth gene) with a vector $\boldsymbol{g}_m$ of size p, $\boldsymbol{g}_m \in \mathbb{R}^p$, and cluster the vectors using the k-means algorithm. We construct sub-tensors using the k-means clusters, where each gene belongs to only one sub-tensor. We perform tensor completion for each of these sub-tensors, and we ultimately combine them back to one tensor with the same gene order as the initial data.

It is essential to consider external information in the clustering process, because if two interacting genes end up in different clusters, the algorithm also cannot fully benefit from the interaction of those genes. Hence, we modify the clustering algorithm as follows. If the induced subnetwork of a subset of genes constitute a connected component in the interaction network, we replace the vectors for all the genes in that subset with a single aggregate vector as their element-wise average. We perform the clustering using the resulting vector. Once we build clusters, we assign all the genes in that subset to the cluster their aggregate vector belongs to. This way, we ensure that all the interacting genes end up in the same cluster.

3 EXPERIMENTAL EVALUATION

In this section, we describe the dataset and summarize the competing methods. We further give details about the experimental setup.

Dataset. We use the sparse dataset from D.Sapashnik et al. [35], containing cell-specific drug responses in a subset of the LINCS connectivity data [39] and included in records GSE70138 and GSE92742 in the Gene Expression Omnibus (GEO) [9]. The data subset consists of a combination of 80 cell lines (60 cancer cell lines, 6 immortalized normal cell lines, 4 stem cell lines and 10 primary cell lines), 1330 drugs, and the expression data from the 978 genes directly measured by the L1000 assay [39], whenever expression data is available for a given drug/cell line combination. However, the expression data for about 75% of the drug/cell line combinations in this dataset are missing, meaning the drug was not assayed in that cell. The data is available online (https://bcb.cs.tufts.edu/cmap/)

.

Competing methods. We compare FiT with four methods, namely the state of the art matrix completion method of Candes and Plan (CP) [3], and the Tissue Agnostic, Two Way, and k-nearest neighbor methods used by D.Sapashnik et al.[35]. CP computes each missing entry by computing two factor matrices. The Two Way Algorithm imputes the missing entry by calculating the median of the entries which are in the same column as the missing entry and the median of the entries which are in the same row as the missing entry and then computes the mean of these two median values. The Tissue Agnostic Algorithm takes the median of the gene expressions that belong to the same drug (column) as the missing data, and imputes the missing data with the median of the expression values. The k-Nearest Neighbor Algorithm finds the k-nearest drugs based on their cosine similarity (using all cells for which both drugs in a pair have data) and predicts the median of the expression values in those drugs.

Experimental setup. We first randomly choose 20% of the drug/cell pairs to be our test dataset for which the transcription values are available, and the remaining 80% for training. We withhold that information from our method as well as the competing methods and impute those values using each method. We do this using 4-fold cross-validation.

We use the actual transcription values as well as the imputed values to infer the connectivity using the connectivity mapping method [21]. This method returns a list of drugs that have positive connectivity scores as well as a list of drugs with negative connectivity scores. Finally, we compute the accuracy of each method as the Weighted Spearman Rank Correlation between the positive (negative) connectivity scores obtained by the imputed transcription values of each method with those of the true transcription values.

Other Details. We implement all the algorithms in Python. We perform the experiments on a Linux machine equipped with 2 AMD EPYC 75F3 32-core processors running at 2.95GHz and 512GBs RAM.

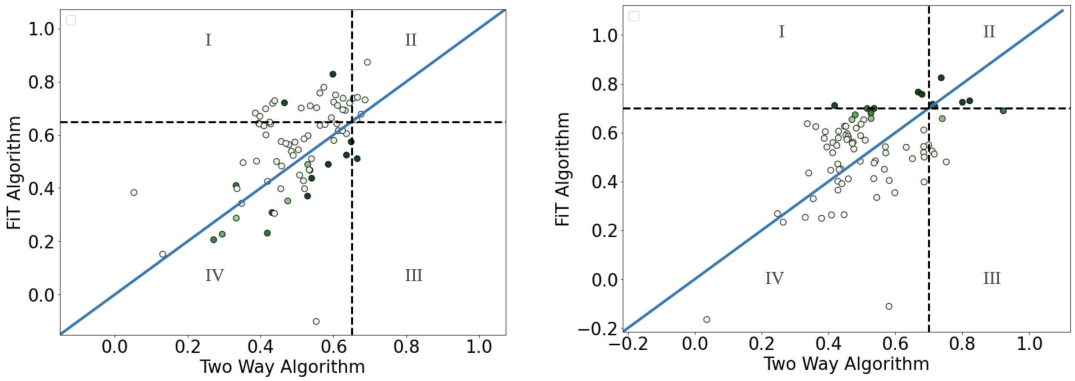
3.1 Accuracy of drug connectivity

We first evaluate how well our method estimates the drug connectivity. Figures 2(a) and 2(b) present the results for positive and negative connectivity respectively. We compare our method to the Two Way algorithm as it uses the most number of available drugs and cell lines for imputing transcription values among all the competing methods. Each circle represents one cell line. We color each circle in gray scale proportional to the number of drugs for which the transcription values are available for the cell line corresponding to that circle. The cell lines with larger number of drugs available have darker color. Diagonal line represents the $x = y$ line. The dashed horizontal and vertical lines represent a threshold for significantly high Weighted Spearman Correlation. Quadrant I represents the cell lines for which FiT Algorithm gives significantly high connectivity correlation values, while the Two Way Algorithm fails. Quadrant II represents the cell lines for which both FiT Algorithm and Two Way Algorithm yield high connectivity correlation values. The quadrant III represents the cell lines for which the Two Way Algorithm produces high connectivity correlation values, and FiT Algorithm fails. Quadrant IV represents the cell lines for which both of the algorithms yield low correlation values.

Figure 2(a) demonstrates that quadrant I contains many more cell lines as compared to quadrant III. This implies that FiT Algorithm is significantly better for identifying positively correlated contact between drugs and cell lines that the Two Way Algorithm. Furthermore, most of the points are above the diagonal, thus FiT outperforms the Two Way Algorithm across all cell lines even for the cell lines for which prediction is harder. We observe that most of the points in quadrant I (and above diagonal) have lighter colors. This implies that FiT Algorithm is more robust to increase in the amount missing. Finally, most of the points with darker color have low correlation values, implying that having fewer data (in this case drugs) often leads to lower performance. In Figure 2(b), we observe that in both quadrant I and quadrant III there are a few points. This implies that estimation of negatively correlated drug connectivity is harder and possibly needs more training data. The clustering of darker points at quadrant II supports this conjecture.

3.2 Cross validation of the predictions

We conjecture that the reason behind our method's success in Figures 2(a) and 2(b), is that it yields more accurate predictions of the transcription values. To verify this, we take a closer look at our method and all the four competing methods and report the root mean square error (RMSE) value of the estimated transcription values. We call this the *loss value*. We calculate the loss value of all the four competing methods as well as that of our algorithm under four settings; 1, 2, 4, and 8 clusters. The 1 cluster setting stands for the tensor completion algorithm that treats the entire set of genes as one big cluster. In the 2 clusters setting, we cluster the entire set of genes into two clusters using the k-means algorithm coded in the scikit-learn implementation [30]. In four and eight cluster settings, we hierarchically split each cluster to get four and eight clusters. We repeat this 10 times with different random seeds and report the average. Table 1 shows the results. We observe the best loss value using FiT under the 2 clusters setting. The change in the loss value of FiT is however negligible for different number of clusters. This suggests that there may be opportunities for better clustering algorithms to bring genes together which have more predictive power of each other's transcription values. Surprisingly, the CP



(a) Positive Connectivity Correlation

(b) Negative Connectivity Correlation

Figure 2: Connectivity Correlation between Two Way Algorithm and FiT Algorithm across all genes and drugs for each of the 80 cell lines. Figure 2(a), and Figure 2(b) represent the positive and the negative connectivity correlation respectively. The diagonal line represents $x = y$ line. The horizontal dashed line denotes the threshold for FiT Algorithm, the cell lines of interest are the ones above the horizontal line. The vertical dashed line denotes a threshold for Two Way Algorithm. The dashed lines (both horizontal and vertical) are at 0.65 for the Figure 2(a) and are at 0.7 for the Figure 2(b). The color tones represent the data available, the color code represent the number of drugs lightest node being 11 drugs and the darkest node color being 1330 drugs.

	Loss values
Two Way	0.944
Tissue Agnostic	0.940
CP	0.859
K-Nearest Neighbor	0.821
FiT- No Cluster	0.767
FiT- 2 Clusters	0.766 ± 0.002
FiT- 4 Clusters	0.767 ± 0.002
FiT- 8 Clusters	0.779 ± 0.001

Table 1: Loss values of different algorithms over the same test dataset.

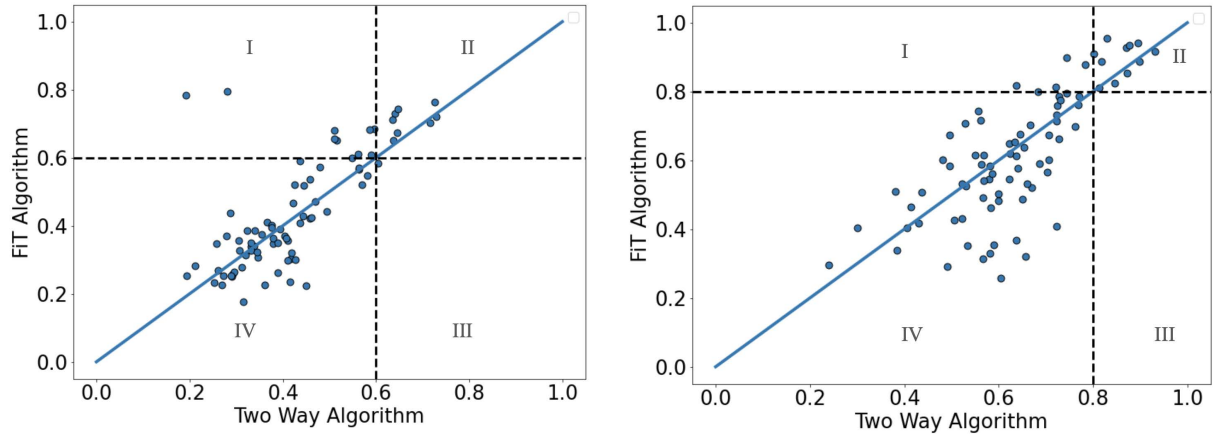
method yields very high loss value. Recall that CP is the traditional matrix completion which treats tensor as a collection independent matrices (i.e., independently acting genes). This supports the value of FiT as it captures the dependence among all the three dimensions (i.e., cell lines, drugs, genes). Finally, we observe that the three local methods (Two Way, Tissue Agnostic, and K-Nearest Neighbor algorithms) all yield very high loss values as compared to FiT, supporting our earlier results that a holistic view of the data used by our algorithm is more promising than local algorithms.

3.3 Correlation evaluation

Next, we zoom in on each cell line and evaluate the correlation between the imputed expression value and the actual expression value for the test data. We run each algorithm and predict transcription values for each drug/cell line combination in test data. We then calculate a correlation value per cell line using two different correlation formulas; weighted Spearman correlation and Pearson correlation. In Figures 3(a) and 3(b) each dot represents one of 80 cell lines when we use Pearson and Spearman correlation values respectively. In each figure, the x-axis denotes the correlation values for the Two Way Algorithm, and the y-axis denotes the correlation values for the FiT Algorithm. We organize the points in the figures into four quadrants the same way as we did in Section 3.1. Quadrant I represents the cell lines with higher correlation values for the FiT Algorithm and lower correlation values for the Two Way Algorithm. Quadrant II has high correlation values for both of the algorithms. Quadrant III has high correlation values for the Two Way Algorithm and low correlation values for the FiT Algorithm. Quadrant IV has low correlation values for both of the algorithms.

In Figure 3(b), we observe that there is no point in quadrant III. It means that no particular cell line has a higher than 0.8 weighted rank Spearman correlation value for the Two Way Algorithm while having a smaller than 0.8 weighted rank Spearman correlation value for the FiT Algorithm. On the other hand, there are some points in quadrant I, where the values predicted by FiT are more correlated with the actual values when evaluated by the weighted Spearman correlation metric. In Figure 3(b), quadrant II is also essential because all the points (cell lines) are either on the line or above the line; meaning that the values predicted by the FiT Algorithm are either more correlated than or equally correlated as the Two Way Algorithm. Quadrant III is again not of interest since both algorithms have smaller correlation values (less than 0.8).

In Figure 3(a), we observe that there is only one point in quadrant III for which the Two Way algorithm gives slightly more correlated prediction than the FiT Algorithm. However, in quadrants I and II, many points have either the same or better correlation value using the FiT Algorithm. Different from the other Figures we have seen so far, there are two particular points (cell lines) which smaller Pearson correlation values ranging between 0.2 and 0.4 when the predictions are made by the Two Way Algorithm whereas approximately 0.8 Pearson correlation value with the FiT Algorithm.



(a) Pearson Correlation

(b) Weighted Rank Spearman Correlation

Figure 3: The correlation value between the actual value and the predicted value across all genes and drugs in test set. We calculate the correlation using two different methods. Figure 3(a), and Figure 3(b) represent the Pearson correlation and Weighted Rank Spearman correlation respectively. The diagonal line represents $x = y$ line. The horizontal dashed line denotes the threshold for FiT Algorithm, the cell lines of interest are the ones above the horizontal line. The dashed lines (both horizontal and vertical) are at 0.6 for the Figure 3(a) and are at 0.8 for the Figure 3(b). The vertical dashed line denotes a threshold for Two Way Algorithm.

3.4 Robustness of the Clusters

The predicted value of a certain gene might be more accurate when the gene is clustered with a certain group of genes. Clustering the gene with a different group of genes might change the accuracy of the prediction. Hence it is important to perform a robustness test on the clusters. We use 10 different seeds (0-9) to perform the k-means clustering. Each seed gives a different clustering and clustering a particular gene with different set of genes may result in different results. We calculate the average random index between the clusters generated using 10 different seeds. Table 2 shows the results.

For the FiT Algorithm with two clusters, we observe that even though the seeds are different, the clusters generated are very similar. The minimum average random index of among 10 seeds is 0.995. As expected, the FiT Algorithm with four clusters and eight clusters, there small differences between the clusters as we increase the number of clusters the genes are clustered in slightly different ways. However, still for most of the seed pairs the cluster similarity measured by average random index is high. Table 1 shows that the standard deviation among all 10 seeds is very low for each clustering setting, which implies that the clusters identified by our clustering algorithm are robust, and thus selecting different seeds would not effect the clusters substantially. It is however worth noting that different clustering algorithms may yield different levels of robustness, which is beyond the scope of this paper.

	Min	Max	Mean	Std Dev
FiT- 2 Clusters	0.995	1.000	0.998	0.001
FiT- 4 Clusters	0.964	1.000	0.990	0.010
FiT- 8 Clusters	0.686	1.000	0.890	0.093

Table 2: Class similarity in terms of average random index for different number of clusters.

3.5 Gene enrichment analysis

We present a qualitative analysis of the performance of our algorithm. The genes that we are interested are the genes for which the FiT Algorithm has smaller RMSE value than the Two Way Algorithm. Hence we sort those genes based on the RMSE values of FiT Algorithm in ascending order and take the top k best performing genes.

We perform gene enrichment analysis on the top k genes [12], where we select k as 10, 20, 50, and 100. We observe that the pathways found for the top k genes are too generic when the value of k is 10 or 20. Hence, we only report the combined results from the top 50 and 100 genes. We threshold the results that have less than 1500 pathway genes; we aim to focus on the functionalities that are not very common. Table 3 presents the thresholded enrichment results. In this table, the Pathway column represents the functionalities that are enriched for the top k genes for which the FiT Algorithm has better prediction than the Two Way Algorithm. Our results demonstrate that the genes for which FiT Algorithm has better prediction than the Two Way Algorithm serve in important functions like the response to oxidative stress, immune effector process, or response to abiotic stimulus. We observe that the pathways are mostly related to response to stress. This coincides with the biological studies in drug response. For instance, reactive oxygen species are found at high levels in tumor cells, and reactive oxygen species-sensitive polymeric nanocarriers improve drug efficacy. [11] They are also known to play a critical role in response to inflammation. [10] Also, recent results demonstrate that HSP70-2 gene exhibit substantial differential expression under oxidative stress for multiple sclerosis patients. [31]. These results suggest that our method is successful in imputing the transcriptional behaviors of the genes that tend to get highly affected by the drug applied on that particular cell line,

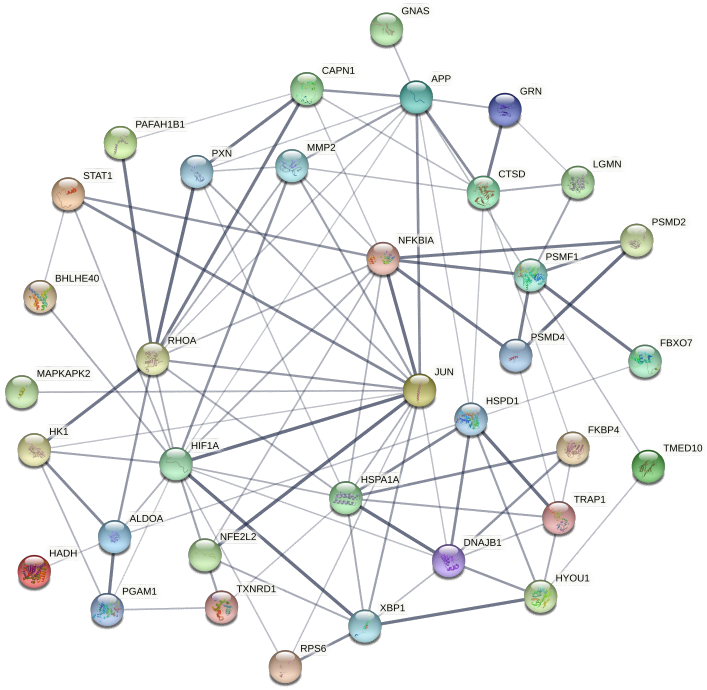


Figure 4: The protein interaction network of the genes for which the FiT Algorithm has smaller loss value than the Two Way algorithm.

and thus there is great potential that FiT can help in solving drug repurposing problem, which is the central goal in this paper.

Next, we focus on the connectivity of the genes that gave the enrichment results in Table 3. First, we create a list consisting of these genes, there are 44 distinct genes in this list. Then, we feed this particular list to the STRING database [41] to further see the connectivity of the genes, and we get the Figure 4. The nodes represent the genes, and they are labeled with the name of the genes. The edges in this network represent the confidence of the connectivity of the genes and the connectivity is decided based on experimental evidence. If the edge is shown with a thick line, the connection has a higher confidence value. The confidence score that we used to plot the Figure 4 is 0.400, and the colors of the nodes are unique to the gene. There are 44 nodes, 98 edges in the network and the average node degree is 4.45. The genes NFKBIA, APP and HIF1A have highest connectivity. Our observation aligns with the existing literature. NFKBIA is one of the genes that potentially provide high confidence drug target candidates for drug repositioning for acute radiation syndrome (ARS) [26]. The amyloid precursor protein (APP) is studied in depth in the context of slowing the rate of disease progression since it produces the Amyloid beta ($A\beta$) peptide which plays an essential role in Alzheimer's disease (AD) [24, 46]. The HIF1A stimulates the transcription of multiple genes related to a wide range of diseases, including cancer [36]. Knockdown of H1F1A is shown to reduce the response of ouabain contributing to the potential antitumor effect of ouabain in NSCLC cells [5]. In summary, the genes for which FiT shows great success interact with each other and they are disease associated. This validates our results in Table 3 that our method has great potential to assist drug repurposing problem.

3.6 Efficiency of the proposed algorithm

So far, in our experiments we have demonstrated the accuracy of our algorithm. Next, we focus on the running time performance. More specifically, we explore the impact of using fiber update strategy used in FiT. To do that, we run two variants of FiT Algorithm with no clustering; with and without fiber update. To distinguish these two variants, we call them FiT-Fiber and FiT-No Fiber respectively. We use the Adagrad [25] algorithm to perform the optimization. We perform two sets of experiments. The first set of experiments aims to test how much the algorithms' running time gets affected by the number of items within a fiber. The second set of experiments aims to test how fast the algorithm reaches a small RMSE value.

For the first set of experiments, we ran both variants of FiT for 100 iterations as both of the algorithms take 100 iterations to converge. Our focus is mainly on the difference between the per iteration performance of the two algorithms. Figure 5 shows the average time each iteration takes with fiber and non-fiber update. One iteration of non-fiber update takes around 5000 seconds when there are 7000 genes. Considering that the algorithm converges in 100 iterations it is expected to take 5000×100 seconds to converge which is approximately 6 days. The human genome has approximately 20K genes, and the non-fiber update is expected to take 18 days to give the same result that the fiber update is giving in less than 3 days. We also observe that the fiber update is not affected as much as the non-fiber update when we increase the number of genes. Furthermore,

Enrichment FDR	nGenes	Pathway Genes	Fold Enrichment	Pathway
9.98E-07	11	217	11.91	Response to reactive oxygen species
9.98E-07	15	490	7.19	Response to oxidative stress
9.98E-07	24	1329	4.24	Response to abiotic stimulus
3.82E-06	17	1489	5.42	Secretion by cell
6.23E-06	13	847	7.28	Cell activation involved in immune response
3.78E-05	12	843	6.76	Leukocyte activation involved in immune response
4.51E-05	14	1245	5.34	Immune effector process

Table 3: Enrichment analysis for the genes that have smaller RMSE value using the FiT Algorithm

as we increase the number of genes, the time gap between fiber and non-fiber updates increases. We thus conclude that our fiber update strategy dramatically improves the performance of drug repurposing.

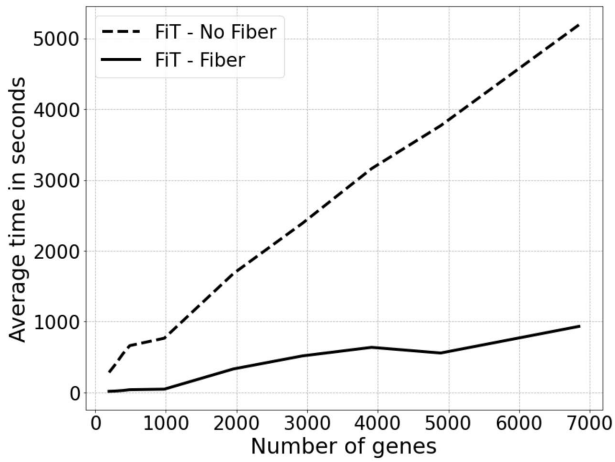


Figure 5: The average time (in seconds) it takes to run one iteration of fiber update (the solid line) and non-fiber update (the dashed line) for different number of genes.

We run both algorithms for the second set of experiments for 100 iterations using the actual dataset, which has 978 genes in each fiber. It takes the non-fiber algorithm 76, 773 seconds to complete 100 iterations, whereas the fiber algorithm takes 12, 758 seconds to complete the same number of iterations. We present the Figure 6 to show the decrease of the RMSE values over time. The fiber update makes the algorithm converge to its minimum value in less time. In Figure 6, we see that after x many seconds, the RMSE value starts increasing. We calculate the RMSE value on the test data, and after a certain number of iterations, the increase of the RMSE value indicates that the algorithm overfits after that many iterations.

Indeed, the fiber update can be further implemented such that the update is performed in parallel for the entire vector of genes, which will introduce further improvement in the time complexity in theory. Considering the computational complexity, the fiber update performs the same update in $\mathcal{O}(1)$ time, whereas the non-fiber update has $\mathcal{O}(n)$ complexity where n is the number of genes (the depth of the tensor). The optimization requires this update to be performed at every iteration, assuming that we have i iterations, the time complexities will be multiplied with i, resulting in a significant difference between algorithms.

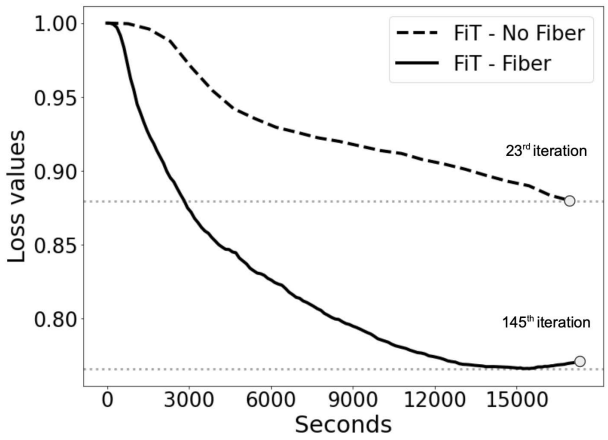


Figure 6: The convergence of fiber update and non-fiber update. The loss values over time (seconds) for fiber update (the solid line) and non-fiber update (the dashed line) with 978 genes. The dotted horizontal lines represents the minimum loss value obtained after 17500 seconds for both fiber and non-fiber update.

4 CONCLUSION

We considered the problem of imputing the impact of a given drug on a cell type, such that although we have prior knowledge about the outcome of another set of drug-cell interactions, we do not know that particular targeted drug-cell interaction. We addressed the first and one of the most fundamental parts of the problem of drug repurposing problem through connectivity mapping. We presented a novel method, named FiT (Fiber-based Tensor Completion) to impute the transcription values for missing drug-cell line combinations in a highly sparse drug-cell dataset accurately and efficiently, while exploiting the distribution of missing values as well as the interactions among genes. Based on our results, FiT outperformed existing approaches and obtained more accurate results in a significantly shorter amount of time, even on sparse datasets where approximately 75% of the data is missing.

REFERENCES

[1] Dimitris Bertsimas and Colin Pawlowski. 2019. Tensor completion with noisy side information for the predcition of anti-cancer drug response.
[2] Aysegul Bumin and Kejun Huang. 2021. Efficient Implementation of Stochastic Proximal Point Algorithm for Matrix and Tensor Completion. In *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, Dublin,Ireland, 1050–1054.
[3] Emmanuel J Candes and Yaniv Plan. 2010. Matrix completion with noise. *Proc. IEEE* 98, 6 (2010), 925–936.

[4] J Douglas Carroll and Jih-Jie Chang. 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika* 35, 3 (1970), 283–319.

[5] Feixiong Cheng, Weiqiang Lu, Chuang Liu, Jiansong Fang, Yuan Hou, Diane E Handy, Ruisheng Wang, Yuzheng Zhao, Yi Yang, Jin Huang, et al. 2019. A genome-wide positioning systems network algorithm for in silico drug repurposing. *Nature communications* 10, 1 (2019), 1–14.

[6] Chia-Chun Chiu, Shih-Yao Chan, Chung-Ching Wang, and Wei-Sheng Wu. 2013. Missing value imputation for microarray data: a comprehensive comparison study and a web tool. *BMC systems biology* 7, 6 (2013), 1–13.

[7] National Research Council. 2005. *Catalyzing Inquiry at the Interface of Computing and Biology*. The National Academies Press, Washington, DC. https://doi.org/10.17226/11480

[8] Alexandre G De Brevern, Serge Hazout, and Alain Malpertuy. 2004. Influence of microarrays experiments missing values on the stability of gene groups by hierarchical clustering. *BMC bioinformatics* 5, 1 (2004), 1–12.

[9] Ron Edgar, Michael Domrachev, and Alex E Lash. 2002. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic acids research* 30, 1 (2002), 207–210.

[10] Danielle Fokam and David Hoskin. 2020. Instrumental role for reactive oxygen species in the inflammatory response. *Frontiers in bioscience (Landmark edition)* 25 (03 2020), 1110–1119. https://doi.org/10.2741/4848

[11] Fengxiang Gao and Zhengrong Xiong. 2021. Reactive Oxygen Species Responsive Polymers for Drug Delivery Systems. *Frontiers in Chemistry* 9 (2021). https://www.frontiersin.org/article/10.3389/fchem.2021.649048

[12] Steven Xijin Ge, Dongmin Jung, and Runan Yao. 2020. ShinyGO: a graphical gene-set enrichment tool for animals and plants. *Bioinformatics* 36, 8 (2020), 2628–2629.

[13] Rajeshwar Govindarajan, Jeyapradha Duraiyan, Karunakaran Kaliyappan, and Murugesan Palanisamy. 2012. Microarray and its applications. *Journal of pharmacy & bioallied sciences* 4, Suppl 2 (2012), S310.

[14] Jiawei Han, Micheline Kamber, and Jian Pei. 2011. *Data Mining: Concepts and Techniques* (3rd ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[15] R. A. Harshman. 1970. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics* 16 (1970), 1–84.

[16] Rachel Hodos, Ping Zhang, Hao-Chih Lee, Qiaonan Duan, Zichen Wang, Neil R. Clark, Avi Ma'ayan, Fei Wang, Brian A. Kidd, Jianying Hu, David A. Sontag, and Joel T. Dudley. 2018. Cell-specific prediction and application of drug-induced gene expression profiles. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* 23 (2018), 32 – 43.

[17] Trey Ideker, Owen Ozier, Benno Schwikowski, and Andrew F Siegel. 2002. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics* 18, suppl_1 (2002), S233–S240.

[18] Hyunsoo Kim, Gene H Golub, and Haesun Park. 2005. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics* 21, 2 (2005), 187–198.

[19] Diederik Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations* abs/1412.6980 (12 2014).

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

[21] Justin Lamb. 2007. The Connectivity Map: a new tool for biomedical research. *Nature reviews cancer* 7, 1 (2007), 54–60.

[22] Hemank Lamba, Vaishnavh Nagarajan, Kijung Shin, and Naji Shajarisales. 2016. Incorporating Side Information in Tensor Completion. In *Proceedings of the 25th International Conference Companion on World Wide Web* (Montréal, Québec, Canada) *(WWW '16 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 65–66.

[23] Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. 2012. Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence* 35, 1 (2012), 208–220.

[24] Justin M Long and David M Holtzman. 2019. Alzheimer disease: an update on pathobiology and treatment strategies. *Cell* 179, 2 (2019), 312–339.

[25] Agnes Lydia and Sagayaraj Francis. 2019. Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci* 6, 5 (2019), 566–568.

[26] Robert Moore, Bhanwar Lal Puniya, Robert Powers, Chittibabu Guda, Kenneth W Bayles, David B Berkowitz, and Tomáš Helikar. 2021. Integrative network analyses of transcriptomics data reveal potential drug targets for acute radiation syndrome. *Scientific reports* 11, 1 (2021), 1–14.

[27] Atsuhiro Narita, Kohei Hayashi, Ryota Tomioka, and Hisashi Kashima. 2012. Tensor factorization using auxiliary information. *Data Mining and Knowledge Discovery* 25, 2 (2012), 298–324.

[28] Shigeyuki Oba, Masa-aki Sato, Ichiro Takemasa, Morito Monden, Ken-ichi Matsubara, and Shin Ishii. 2003. A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics* 19, 16 (2003), 2088–2096.

[29] Vladislav Yu Orekhov, Ilghiz Ibraghimov, and Martin Billeter. 2003. Optimizing resolution in multidimensional NMR by three-way decomposition. *Journal of biomolecular NMR* 27, 2 (2003), 165–173.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[31] Cristiana Pistono, Maria Cristina Monti, Chiara Boiocchi, Francesca Gigli Berzolari, Cecilia Osera, Giulia Mallucci, Mariaclara Cuccia, Alessia Pascale, Cristina Montomoli, and Roberto Bergamaschi. 2020. Response to oxidative stress of peripheral blood mononuclear cells from multiple sclerosis patients and healthy controls. *Cell Stress and Chaperones* 25, 1 (2020), 81–91. https://doi.org/10.1007/s12192-019-01049-0

[32] Sudeep Pushpakom, Francesco Iorio, Patrick A Eyers, K Jane Escott, Shirley Hopper, Andrew Wells, Andrew Doig, Tim Guilliams, Joanna Latimer, Christine McNamee, et al. 2019. Drug repurposing: progress, challenges and recommendations. *Nature reviews Drug discovery* 18, 1 (2019), 41–58.

[33] Sujay Saha, Kashi Nath Dey, Riddhiman Dasgupta, Anirban Ghose, and Koustav Mullick. 2013. Missing value estimation in DNA microarrays using B-splines. *Journal of Medical and Bioengineering* 2, 2 (2013), 88–92.

[34] Sujay Saha, Praveen Kumar Singh, and Kashi Nath Dey. 2016. *Missing Value Estimation in DNA Microarrays using Linear Regression and Fuzzy Approach*. De Gruyter, Trivandrum, India, 254–268. https://doi.org/10.1515/9783110450101-024

[35] Diana Sapashnik, Rebecca Newman, Christopher Michael Pietras, Fangfang Qu, Lior Kofman, Sean Boudreau, Inbar Fried, and Donna K. Slonim. 2021. Cell-specific imputation of drug connectivity mapping with incomplete data. *bioRxiv* 1, 1 (2021), 2020–08. https://doi.org/10.1101/2020.08.10.231720 arXiv:https://www.biorxiv.org/content/early/2021/05/20/2020.08.10.231720.full.pdf

[36] Gregg L Semenza. 2003. Targeting HIF-1 for cancer therapy. *Nature reviews cancer* 3, 10 (2003), 721–732.

[37] Qingquan Song, Hancheng Ge, James Caverlee, and Xia Hu. 2019. Tensor completion algorithms in big data analytics. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 13, 1 (2019), 1–48.

[38] Paul T Spellman, Gavin Sherlock, Michael Q Zhang, Vishwanath R Iyer, Kirk Anders, Michael B Eisen, Patrick O Brown, David Botstein, and Bruce Futcher. 1998. Comprehensive identification of cell cycle–regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. *Molecular biology of the cell* 9, 12 (1998), 3273–3297.

[39] A. Subramanian et al. 2017. A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles. *Cell* 171, 6 (Nov 2017), 1437–1452.

[40] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. 2019. STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research* 47, D1 (2019), D607–D613.

[41] Damian Szklarczyk, Annika L Gable, Katerina C Nastou, David Lyon, Rebecca Kirsch, Sampo Pyysalo, Nadezhda T Doncheva, Marc Legeay, Tao Fang, Peer Bork, et al. 2021. The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic acids research* 49, D1 (2021), D605–D612.

[42] Giorgio Tomasi and Rasmus Bro. 2005. PARAFAC and missing values. *Chemometrics and Intelligent Laboratory Systems* 75, 2 (2005), 163–180. https://doi.org/10.1016/j.chemolab.2004.07.003

[43] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17, 6 (2001), 520–525.

[44] Bo-Wen Wang, Vincent S Tseng, et al. 2012. Improving missing-value estimation in microarray data with collaborative filtering based on rough-set theory. *International Journal of Innovative Computing, Information and Control* 8, 3 (2012), 2157–2172.

[45] Guo Wei, David Twomey, Justin Lamb, Krysta Schlis, Jyoti Agarwal, Ronald W Stam, Joseph T Opferman, Stephen E Sallan, Monique L den Boer, Rob Pieters, et al. 2006. Gene expression-based chemical genomics identifies rapamycin as a modulator of MCL1 and glucocorticoid resistance. *Cancer cell* 10, 4 (2006), 331–342.

[46] Yun Xu, Jiming Kong, and Pingzhao Hu. 2021. Computational Drug Repurposing for Alzheimer's Disease Using Risk Genes from GWAS and Single-Cell RNA Sequencing Studies. *Frontiers in Pharmacology* 12 (2021).

[47] Chong Zhang, Yong-Ku Ryu, Taylor Z Chen, Connor P Hall, Daniel R Webster, and Min H Kang. 2012. Synergistic activity of rapamycin and dexamethasone in vitro and in vivo in acute lymphoblastic leukemia via cell-cycle arrest and apoptosis. *Leukemia research* 36, 3 (2012), 342–349.

[48] Tengfei Zhou, Hui Qian, Zebang Shen, Chao Zhang, and Congfu Xu. 2017. Tensor Completion with Side Information: A Riemannian Manifold Approach *(IJCAI'17)*. AAAI Press, Melbourne, 3539–3545.