MATRIX COMPLETION TECHNIQUES IN BIOINFORMATICS

by

AANCHAL MONGIA

Under the Supervision of Dr. Angshul Majumdar

Indraprastha Institute of Information Technology Delhi

September, 2020

MATRIX COMPLETION TECHNIQUES IN BIOINOFRMATICS

by

AANCHAL MONGIA

Submitted

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

to the

Indraprastha Institute of Information Technology Delhi

September, 2020

Certificate

This is to certify that the thesis titled *Matrix completion techniques in Bioinformatics* being submitted by *Aanchal Mongia* to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of Doctor of Philosophy, is an original research work carried out by her under my supervision. In my opinion, the thesis has reached the standard fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree or diploma.

September, 2020

Dr. Angshul Majumdar

Indraprastha Institute of Information Technology Delhi

New Delhi 110020

Acknowledgements

I take this opportunity to express my heart-felt gratitude for my amazing supervisor Dr. Angshul Majumdar for his consistent support, immense knowledge, scientific inputs and dedicated help. I don't think I would be able to thank him enough for the encouragement, research freedom and patience he has extended. He has been exceptionally motivating, understanding and a guide in true sense. I am really pleased to have been associated with a person like Dr. Angshul in my life. During the last three years, I have learnt extensively from him including how to raise new possibilities, how to regard an old question from a new perspective, time management, etc. To put in simple words, I could not have a better supervisor.

My special words of thanks should also go to Dr. Debarka Sengupta without the valuable guidance of whom this work would not have been possible. He has extended his valuable time, biological insights and expert guidance during the initial years of my Ph.D.

I would also like to thank Dr. Emilie Chouzenoux for research collaboration and the theoretical expertise she has provided in this dissertation.

I would like to thank my family, especially the pillar of my life, my father for always believing in me, supporting my decisions, constantly backing me with his immense care and love throughout my life.

I owe a special thanks to the members of SALSA lab for being a support system to my professional and personal life in the past three years.

I also express my regards to Miss Priti Patel (Assistant manager) in our university. She has never failed to provide timely advice and support to all the administrative matters involved right from the beginning. I am also grateful to the IT helpdesk team of our university for always providing timely technical support whenever needed and the Indraprastha Institute of Information Technol-

ogy for providing excellent infrastructure and research environments.

Abstract

Data analytics and computational techniques applied to biological sciences aid rapid technological advances, swift discoveries, and reliable analysis. A broad range of bountiful tools and algorithms have played pivotal roles in a variety of biological applications. One such class of algorithms: "Matrix completion", motivated from recommender systems, has been used to solve different kinds of biological problems. This dissertation proposes the use of novel low-rank matrix completion algorithms and their variants as a contribution for two fields: scRNA-sequencing and drug re-positioning. Specifically, biological problems such as scRNA-seq imputation, drug-target interaction prediction, drug-disease association prediction, and the most motivating one, virus-drug prediction (proposed to contribute towards a cure for COVID-19) have been modeled as matrix completion frameworks bridging the gap between two seemingly disjoint research fields, collaborative filtering, and bioinformatics, initiating a symbiotic or deep collaborative relationship between the two.

Firstly, this dissertation proposes one of the early tools for the imputation of scRNA-seq gene expression data. The single-cell RNA seq technology allows the measurement of gene expression at a single-cell resolution but has a disadvantage of a low amount of mRNA in individual cells. This eventually leads to dropouts in the single-cell gene expression data hindering the single-cell downstream analysis. We handle the dropouts problem by modeling scRNA-seq imputation as a missing-value prediction problem, employing a novel deep matrix completion framework.

The second contribution is largely incremental in terms of biological application but novel when looked at from an algorithmic perspective. With the aim of drug re-positioning/drug re-purposing (predicting new targets/diseases for existing drugs), we propose techniques for drug-disease association and drug-target interaction prediction. Both take into account the side-information associated with the drug and target entities and deploy graph regularized matrix completion

frameworks for the aforesaid tasks.

Apart from this, the third application has consequentially sprouted from the algorithmic contributions of this thesis which finds its direct mapping to predict anti-viral treatments/effective against SARS-Cov-2. We put forward a matrix completion framework based on a manually curated drug-virus association dataset, which uses variants of matrix completion methods (including the proposed ones) for virus-drug association prediction. This work interestingly covers the entire spectrum of tasks ranging from data curation to algorithms and biological implications.

The fourth and the last contribution of this dissertation is a new framework which can collaboratively perform matrix completion, finding its application in imputation on combined proteomics and transcriptomics data obtained from RNA sequencing methods such as CITE-seq in which the RNA data is expected to have relatively more dropouts (due to higher amounts of protein in a cell).

RNA ribonucleic acid

scRNA-seq single cell RNA sequencing

DTI drug target interaction

DDA drug disease association

DVA drug virus association

1L 1 layer

2L 2 layer

3L 3 layer

SVD singular value decomposition

MM majorization minimization

ADMM alternating direction method of multipliers

PPXA parallel proximal algorithm

S similarity matrix

L row laplacian matrix

N neighborhood matrix

SARS-CoV severe acute respiratory syndrome coronavirus

COVID-19 corona virus disease-19

CV cross validation

MPV maximum precision value

ROC receiver operating characteristic curve

AUC area under ROC curve

AUPR area under precision recall curve

ARI adjusted rand index

Tr Trace

$\circ$ hadamard product

$\|\|.\|\|_1$	l_2 norm
$\|\|.\|\|_2$	l_2 norm
$\|\|.\|\|_F$	Frobenius norm
$\|\|.\|\|_*$	Nuclear norm
$soft$	soft thresholding operator
$(.)^T$	transpose
$(.)^{-1}$	inverse

Contents

Acknowledgements

Abstract **i**

Acronyms and Symbols **iii**

List of Tables **x**

List of Figures **xii**

1 Introduction **2**

1.1 Underlying frameworks 3

1.1.1 Matrix factorization 4

1.1.2 Nuclear norm minimization 6

1.1.3 Deep matrix factorization 9

1.1.4 Graph regularization 10

1.2 Problems in brief 11

2 scRNA-seq imputation using deep matrix completion **15**

2.1 Introduction 16

2.2 Dataset 18

2.3 Methodology 19

2.3.1 Data preprocessing 19
2.3.2 Proposed framework 20
2.4 Results 26
2.4.1 Contribution 26
2.4.2 Improvement in clustering accuracy 27
2.4.3 Improved differential genes prediction 28
2.4.4 Improvement in cell type separability 30
2.5 Conclusion 31

3 Drug-target interaction prediction using multi graph regularized nuclear norm minimization 33

3.1 Introduction 34
3.2 Dataset 39
3.3 Methodology 41
3.3.1 Data preprocessing 42
3.3.2 Proposed framework 43
3.4 Results 48
3.4.1 Experimental setup 48
3.4.2 Parameter settings 51
3.4.3 Interaction prediction 51
3.4.4 Validation of multiple similarities 52
3.5 Conclusion 53

4 Drug-disease association prediction using graph-regularized one bit matrix completion 57

4.1 Introduction 58
4.2 Dataset 60

4.3 Methodology 62
4.3.1 Data preprocessing 62
4.3.2 Proposed framework 62
4.4 Results 67
4.4.1 Evaluation criteria 67
4.4.2 Comparison with benchmark techniques 69
4.4.3 Parameter settings 72
4.4.4 Case study to predict novel associations 73
4.5 Conclusion 74

5 Drug-virus association database: anti-viral drug prediction using matrix completion 75

5.1 Introduction 76
5.2 Dataset 79
5.2.1 Drug-virus association compilation 79
5.2.2 Similarity computation 80
5.3 Methodology 82
5.3.1 Data preprocessing 82
5.3.2 Proposed framework 82
5.3.3 Setting of hyperparameters 86
5.4 Results 87
5.4.1 Overview: DVA prediction 88
5.4.2 Empirical evaluation 90
5.4.3 Association prediction for new drugs 91
5.4.4 SARS-CoV-2 prediction 92
5.4.5 Predictions evolution with mutating novel coronavirus . 94

5.5 Conclusion 96

6 Transcriptomic-proteomic expression completion using collaborative matrix completion 98

6.1 Introduction 99

6.2 Dataset 100

6.3 Methodology 100

6.3.1 Data preprocessing 101

6.3.2 Proposed framework 101

6.4 Results 104

6.4.1 Cell visualization 105

6.5 Conclusion 107

7 Conclusion 108

7.1 Summary of contribution 108

7.1.1 scRNA-seq imputation using matrix completion frameworks 108

7.1.2 Drug-target Interaction prediction using multi graph regularized nuclear norm minimization 109

7.1.3 Drug-disease association prediction using graph-regularized one bit matrix completion 109

7.1.4 Drug-virus association database: anti-viral drug prediction using Matrix completion 110

7.1.5 Transcriptomic-proteomic expression completion using collaborative matrix completion 110

7.2 Future work 111

References 113

Appendices **136**

.1 Plot of singular value decay 137

.2 Majorization minimization 137

List of Tables

1.1 Summary of the problems of interest along with the models deployed. 14

3.1 Drugs, Targets and Interactions in each dataset used for validation. 41

3.2 AUPR results for interaction prediction under validation setting CVS1. 54

3.3 AUC results for interaction prediction under validation setting CVS1. 54

3.4 AUPR results for interaction prediction under validation setting CVS2. 54

3.5 AUC results for interaction prediction under validation setting CVS2. 54

3.6 AUPR results for interaction prediction under validation setting CVS3. 56

3.7 AUC results for interaction prediction under validation setting CVS3. 56

4.1 A summary of the number of associations, drugs and diseases in each dataset used. 61

4.2 Average AUC across 10-fold cross-validation for various techniques while predicting drug-disease associations. 69

4.3 Average AUPR across 10-fold cross-validation for various techniques while predicting drug-disease associations. 70

4.4 Top 5 predicted diseases for Levodopa, Doxorubicin, Amantadine, Flecainide and Metformin with their evidence in CTD database 72

5.1 Results for association prediction for all techniques under the 3 cross validation settings. 91

5.2 Number and percentage of drugs predicted with MPV=1 by the matrix completion methods. 92

5.3 Top-10 drugs predicted for SARS-Cov-2 by the DVA computational methods. 93

5.4 Top-10 drugs predicted for three isolates of SARS-Cov-2 (collected at an interval of 2 months) by the DVA computational methods. 97

List of Figures

1.1 Matrix factorization as product of two latent factor matrices . . . 4

1.2 Graph regularization: Modeling rows and columns as nodes and their side information as edges of the graphs 10

2.1 Overview of deepMc pipeline for imputing single cell RNA sequencing data. 21

2.2 ARI values obtained after applying k-means clustering post various imputation techniques. 28

2.3 Plot showing ROC curve and AUC depicting how well do the DE genes predicted from scRNA and matching bulk RNA-Seq data agree. DE calls were made on expression matrix imputed using various methods. 29

2.4 Plot showing the variation of CTS with various imputation strategies. 31

3.1 Drug repositioning using DTI 35

3.2 Bar plots depicting that incorporating all the similarities for drugs and targets for prediction task yields best results for every dataset **(a)** E **(b)** IC **(c)** GPCR and **(d)** NR under the three cross-validation settings in comparison to the cases where each type of similarity was considered separately. Here, *standard* represents the case when only the chemical structure similarity for drugs and genomic sequence similarity for targets were taken into account and *COMBINED* refers to the use case where all the similarity matrices (standard similarity, Cosine similarity, Correlation, Hamming similarity and Jaccard similarity) were considered. 52

4.1 A schematic overview of GR1BMC for predicting drug-disease assocations 61

4.2 Convergence plot for GR1BMC on Fdataset 67

4.3 Converegence plot for GR1BMC on C dataset 68

4.4 ROC curves obtained for all the 10 folds after applying GR1BMC on Fdataset 70

4.5 ROC curves obtained for all the 10 folds after applying GR1BMC on Cdataset 71

5.1 Schematic diagram depicting the DVA framework 89

6.1 Schematic diagram depicting the collabrative imputation framework 100

6.2 t-SNE representation of cells from unimputed gene expression data 105

6.3 t-SNE representation of cells from imputed gene expression data 106

6.4 t-SNE representation of cells from unimputed protein expression data 106

6.5 t-SNE representation of cells from imputed protein expression data 107

1 Singular value decay plot of each of the dataset taken from (a) scRNA-seq imputation (b) DTI prediction (c) DDA prediction (d) DVA prediction (e) gene expression matrix and (f) protein expression matrix in transcriptomic-proteomic prediction problem 139

2 Majorization Minimization - Schematic Diagram: 140

Chapter 1

Introduction

This dissertation is explicit and the first work to model a range of Bioinformatics problems using matrix completion frameworks. All biological prediction or imputation problems where the data can be structured as a low-rank matrix and has limited noise may be approached via matrix completion technique or any of its variants. If the data to be recovered is accompanied by side-information or metadata associated with the row and column entities, one can take leverage the use of graph regularized Matrix completion frameworks too (refer section 1.1.4).

In this chapter, we give an overview of algorithmic frameworks that form the backbone of our work (first section). We also describe the contribution of this dissertation in brief in the next section.

1.1 Underlying frameworks

The problem of completing a partially observed matrix X is called Matrix completion. The complete matrix is constituted by the known and the yet unknown values. We can assume that the data that we have acquired, Y is a sampled version of the complete expression matrix X. Mathematically, this is expressed as,

$$Y = M \circ (X) \tag{1.1}$$

Here M is the sub-sampling operator and $\circ$ is the Hadamard product (element-wise multiplication operator). M is binary and has 0's where the complete data X has not been observed and 1's where it has been. The values of M are element-wise multiplied ($\circ$) to the complete matrix X so that Y (the sub-sampled data) is a sparse representation of X and has values only at positions where values are observed. Our problem is to recover X, given the observations Y, and the sub-sampling mask A. It is known that X is of low-rank (refer Appendices section 1).

It should be noted that matrix completion is a well-studied framework [1]. Below, we consider two algorithms for data completion: Matrix factorization [2] and Nuclear norm minimization [3].

1.1.1 Matrix factorization

Matrix factorization is the most straightforward way to address the low-rank matrix completion problem; it has previously been used for finding lower dimensional decompositions of matrices [4]. Say X is of dimensions $m \times n$, but is known to have a rank r ($<m, n$). In that case, one can express $X_{m\times n}$ as a product of two matrices $U_{m\times r}$ and $V_{r\times n}$. Therefore the complete problem (1.1) can be formulated as,

$$Y = M \circ (X) = M \circ (UV) \tag{1.2}$$

Estimating U and V from (1.2) tantamount to recovering X.

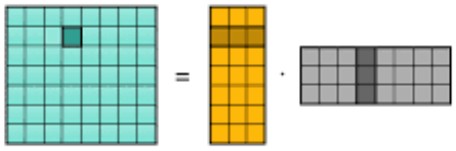


Figure 1.1: Matrix factorization as product of two latent factor matrices

The two matrices U and V can be solved by minimizing the *Frobenius* norm of the following cost function,

$$\min_{U,V} ||Y - M \circ (UV)||_F^2 \tag{1.3}$$

Since this is a bi-linear problem, one cannot guarantee global convergence. However, it usually works in practice. It has been used for solving recommender systems problems [2], where (1.3) was solved using stochastic gradient descent (SGD). SGD is not an efficient technique and requires tuning of several

parameters. In this work, we will solve (1.3) in a more elegant fashion using Majorization-Minimization (MM) [5].

For our given problem, the cost function to be minimized is given as $J(X) = ||Y - A(X)||_F^2$; the majorization step basically decouples the problem (from A), so that we can solve the optimization problem by solving the following,

$$\min_{U,V} ||B - UV||_F^2 \tag{1.4}$$

where $B_{k+1} = X_k + \frac{1}{a}A^T(Y - A(X_k))$ at each iteration k. Here, X_k is the matrix at iteration k and a is a scalar parameter in the MM algorithm.

This (1.4) is solved by alternating least squares [6], i.e. while updating U, V is assumed to be constant and while updating V, U is assumed to be constant,

$$U_k \leftarrow \min_{U} ||B - U_{k-1}V_{k-1}||_F^2 \tag{1.5}$$

$$V_k \leftarrow \min_{V} ||B - U_kV_{k-1}||_F^2 \tag{1.6}$$

Since the input would never be negative in case of our problem/s, we have imposed a non-negativity constraint on the recovered matrix X, so that it does not contain any negative values.

The matrix factorization algorithm has been summarized in **Algorithm 1**. The initialization of factor V is done by keeping r right singular vectors of X in V obtained by performing singular value decomposition (SVD) of X, where r is the approximate rank of the expression matrix to be recovered.

Algorithm 1 Matrix completion using matrix factorization

procedure MATRIX-FACTORIZATION(Y, A, r)
  Initialize: $X = random$, a, V (SVD initialization), k and l.
    For loop 1, iterate (k)
      $B_k = X_{k-1} + \frac{1}{a}A^T(Y - A \circ X_{k-1})$
      For loop 2, iterate (l)
        $U_l \leftarrow \min_U ||B_k - U_{l-1}V_{l-1}||_F^2$
        $V_l \leftarrow \min_V ||B_k - U_l V_{l-1}||_F^2$
      End loop 2
      $X_k = U_k V_k$
      $X_k \leftarrow X_k^+$
    End loop 1

1.1.2 Nuclear norm minimization

The problem depicted in (1.3) is non-convex. Hence, there is no guarantee for global convergence. Also one needs to know the approximate rank of the matrix X in order to solve it, which is unknown in this case. To combat this issue, researchers in applied mathematics and signal processing proposed an alternative solution. They would directly solve the original problem (1.1) with a constraint that the solution is of low-rank. This is mathematically expressed as,

$$\min_X \text{rank}(X) \text{ s.t. } Y = M \circ (X) \tag{1.7}$$

Here, rank(X) denotes the rank or the number of non zero singular values of X. The above problem turns out to be NP hard problem with doubly exponential complexity. Therefore, studies in matrix completion [7, 8] proposed relaxing the NP hard rank minimization problem to its closest convex surrogate: nuclear

norm minimization,

$$\min_X ||X||_* \text{ s.t. } Y = M \circ (X) \tag{1.8}$$

Here $||.||_*$ is the nuclear norm and is defined as the sum of singular values of data matrix X. It is the l_1 norm of the vector of singular values of X and is the tightest convex relaxation of the rank of matrix, and therefore its ideal replacement.

This is a semi-definite programming (SDP) problem. Usually its relaxed version (Quadratic Program) is solved [9] with the unconstrained Lagrangian version,

$$\min_X ||Y - M \circ (X)||_F^2 + \lambda ||X||_* \tag{1.9}$$

Here, $||.||_*$ is the nuclear norm and λ is called the Lagrange multiplier. The problem (1.9) does not have a closed form solution and needs to be solved iteratively.

To solve (1.9), we invoke MM once more. Here $J(X) = ||Y - M \circ (X)||_F^2 + \lambda ||X||_*$, we can express (1.9) in the following fashion in every iteration k,

$$\min_X ||B - X||_F^2 + \lambda ||X||_* \tag{1.10}$$

where $B_{k+1} = X_k + \frac{1}{a} M^T (Y - M(X_k))$.

Using the inequality $||Z_1 - Z_2||_F \geq ||s_1 - s_2||_2$, where s_1 and s_2 are singular values of the matrices Z_1 and Z_2 respective, we can solve the following instead

of solving the minimization problem (1.10),

$$\min_{s_x} ||s_B - s_X||_2^2 + \lambda ||s_X||_1 \tag{1.11}$$

Here s_B and s_X are the singular values of B and X, respectively and $||s_X||_1$ is the l_1 norm or the sum of absolute values of s_X. It has been shown that problem (1.10) is minimized by soft thresholding the singular values with threshold $\lambda/2$. The optimal update is given by,

$$s_X = \begin{cases} s_B + \lambda/2 \text{ when } s_B \leq -\lambda/2 \\ 0 \text{ when } |s_B| \leq \lambda/2 \\ s_B - \lambda/2 \text{ when } s_B \geq \lambda/2 \end{cases} \tag{1.12}$$

or more compactly by,

$$s_X = \text{soft}(s_B, \lambda/2) = \text{sign}(s_B)\text{max}(0, |s_B| - \lambda/2) \tag{1.13}$$

Algorithm 2 Matrix completion via nuclear norm minimization

procedure MATRIX-NNM(Y, M)
  Initialize: $X = random$, a
    For loop 1, iterate (k)
      $B_k = X_{k-1} + \frac{1}{a} M^T (Y - M \circ X_{k-1})$
      Compute SVD (singular value decomposition) of B : $B_k = USV^T$
      Soft threshold the singular values: $\Sigma = soft(S, \lambda/2)$ ▷ refer equation 6.2
      $X_k = U\Sigma V^T$
      $X_k \leftarrow X_k^+$
    End loop 1

We found that the algorithm is robust to values of the hyperparameter λ (a scalar) as long as as it is reasonably small (< 0.01).

Here, we have optionally imposed the non-negativity constraint on X pro-

vided the values to be recovered are not smaller than zero (a constraint in most applications).

1.1.3 Deep matrix factorization

In recent times, deep learning has permeated almost every aspect of computational science. Min et al. [10] give a comprehensive treatise into the early applications of deep learning in this area. The deep matrix completion frameworks of matrix completion proposed in this dissertation are motivated by the success of deep matrix factorization [11, 12], and deep dictionary learning [13]. The basic idea in there is to factor the data matrix into several layers of basis and a final layer of coefficients; shown here for three levels,

$$X = D_1 D_2 D_3 Z \tag{1.14}$$

Note that this is a feedbackward neural network, the connections are from the nodes towards the input. This is because matrix factorization is a synthesis formulation. Incorporating the deep matrix factorization formulation into (1.1) leads to,

$$Y = M \circ D_1 D_2 D_3 Z \tag{1.15}$$

Our task is to solve the different layers of basis (D1, D2, D3) and the coefficients

(Z) by solving the least squares objective function,

$$\min_{D_1,D_2,D_3,Z} ||Y - M \circ (D_1 D_2 D_3 Z)||_F^2 \tag{1.16}$$

Note that we cannot use techniques derived in [11, 12] for our purpose; this is because they operated on the full data where as we need to derive for partially observed data.

1.1.4 Graph regularization

Graph regularization assumes that points close to each other in the original space should also be close to each other in the learned manifold (*Local Invariance assumption*). So, Graph regularization would allow the algorithm to learn manifolds for the row and column spaces in which the data is assumed to lie.

For graph regularization of a matrix completion framework, we model the row and column entities as nodes of the corresponding row and column graphs. The distance/similarity information between the entities is interpreted as the edges of the graphs 1.2. This similarity information between each of the rows/-columns is encoded as a square symmetric matrix.

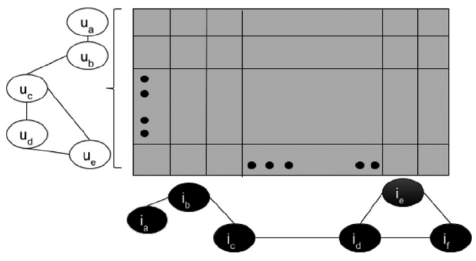


Figure 1.2: Graph regularization: Modeling rows and columns as nodes and their side information as edges of the graphs

The standard versions of both the matrix factorization and nuclear norm minimization techniques are unable to incorporate similarity information of the drugs and the targets. In recent studies [14, 15], it was shown that the best results are obtained when these techniques incorporate graph regularization penalties into them. The authors regularize the objective function by taking into account, the similarity for row and column entities (encoded as matrices S_r and S_c) by adding graph laplacian penalties (computed along rows and column) in the formulation [16]. The graph lalplacian [17] is defined as $L_r = D_r - S_r$ for rows and $L_c = D_c - S_c$ for columns, $D_r^{ii} = \Sigma_j S_r^{ij}$ and $D_c^{ii} = \Sigma_j S_c^{ij}$ being the diagonal matrices having degree of each node (a row or column here) on its diagonal. The current works have incorporated the standard similarity measures for drugs and targets in matrix factorization [14] and Matrix completion [15] frameworks.

The graph regularized version of Matrix facrorization is given by,

$$\min_{U,V} ||Y - M \circ (UV)||_F^2 + \mu_1 Tr(U^T L_d U) + \mu_2 Tr(V L_t V^T) \tag{1.17}$$

The graph regularized version of Nuclear Norminimization is given by,

$$\min_{X} ||Y - M \circ (X)||_F^2 + \lambda ||X||_* + \mu_1 Tr(X^T L_d X) + \mu_2 Tr(X L_t X^T) \tag{1.18}$$

1.2 Problems in brief

In this section, we briefly mention the core models proposed and contributions of this dissertation. The dissertation unfolds advancements along both biologi-

cal and algorithmic lines.

With the goal of modeling the crucial bioinformatics problem of handling dropouts in single-cell RNA seq (scRNA-seq) data [18, 19], we first model the imputation task as a matrix completion problem (Chapter 2). Ours was one of the first few initial works to handle this problem. Motivated by the success of deep learning in various fields, we model this using a novel framework developed, leveraging deep learning in matrix factorization [18] (Chapter 2).

Secondly, we propose to incorporate the various kinds of metadata associated with the rows and column entities of the matrix by targeting interaction/association prediction problems such as drug-target interaction (DTI) prediction and drug-disease association (DDA) prediction [20, 21] (Chapters 3 and 4). This primarily is a fast, efficient, and intelligent way of repurposing/repositioning drugs via computational means by pruning out the drug-space to be tested against the target proteins /diseases. We introduce multi-graph regularized nuclear norm minimization to solve the DTI prediction problem and use the standard graph regularized solution with a novel solution which takes into account the binary nature of the data to solve the DDA prediction task respectively.

Motivated by the above methodology of drug repositioning, we curate a drug-virus association database (Chapter 5) from the available sources and try our bit as data scientists to help clinicians in selecting anti-viral drugs [22]. For this, we model drug-virus association prediction (with the curated drug-virus association dataset, available publicly) as a matrix completion task. To enable the use

of graph-regularized methods, we have collected the drug and virus metadata using the chemical structure of drugs and the genomic sequences of viruses and incorporated this as the drug and virus similarity/Laplacian matrices. Using the best performing techniques, we computationally recommend six drugs (repositioned) that would be effective against SARS-CoV-2 (the virus which causes COVID-19) as per the predictions of our model, four of which are already under trial for COVID-19, suggesting that the computational results are in sync with the current state of practice..

Lastly, we render the benefits of single cell multiplexing where tools like REAP-seq [23] enable simultaneous measurement of gene as well as protein expression in single cells. Since this data has dropouts due to low starting material in single cells, we devise a formulation (refer Chapter 6) called collaborative matrix completion, to simultaneously co-complete the gene and protein expression matrices deriving the similarities between the cells from each of these expression data.

The chapter wise biological application along with the proposed model/methadology has been summarized in the Table 1.1 below.

Of note, all the matrix completion models proposed in this dissertation are devised for specific biological problem in hand, however, the usage is not limited and can be extended to other similar problems in each case.

Chapter	Application	Proposed Model	Rows	Columns
2	sc-gene expression	Deep matrix completion	cell	gene
3	Drug-target interaction	Multi graph regularized matrix completion	drugs	proteins
4	Drug-disease association	Graph-regularized one bit matrix completion	drugs	diseases
5	Drug-virus assocaition	All above methods	drugs	viruses
6	sc-transcriptomic proteomic expression	Collaborative matrix completion	cells	genes/ proteins

Table 1.1: Summary of the problems of interest along with the models deployed.

Chapter 2

scRNA-seq imputation using deep matrix completion

Single cell RNA-seq has inspired new discoveries and innovation in the field of developmental and cell biology over the past few years and is useful for studying cellular responses at individual cell resolution. But, due to the paucity of starting RNA, the data acquired has dropouts. To address this, we propose a deep matrix factorization based method, deepMc, to impute missing values in gene-expression data. For the deep architecture of our approach, we draw our motivation from great success of deep learning in solving various machine learning problems. In this work, we support our method with positive results on several evaluation metrics like clustering of cell populations, differential expression analysis and cell type separability.

2.1 Introduction

Bulk RNA sequencing has traditionally been used in transcriptome studies [24, 25, 26] for parallel screening of thousands of genes, revealing a global view of averaged expression levels. Single cell RNA sequencing (scRNA-seq), on the contrary, enables transcriptomic analysis and measurement of gene expressions at the single cell level, thus providing more perceptivity into functioning of individual cells. Over the past few years, scRNA-seq has transformed the field of functional biology and genomics [27] by enabling characterization of phenotypic diversity among seemingly similar cells [28, 29, 30]. This unique feature has been proved critical in characterizing cancer heterogeneity [31, 32], identification of new rare cell types and understanding the dynamics of transcriptional changes during development [33, 34, 35].

However, this powerful technology (like many other biological data) suffers from a number of sources of biological and technical noise and biases, the major one being lack of starting mRNA captured in individual cells. Due to small quantities transcripts are frequently missed during the reverse transcription step. This leads to 'dropout' events, where only a fraction of transcriptome of each cell is detected during the sequencing step [36], leading to a sparse gene-expression matrix. This is more prevalent in the lowly expressed genes. Excluding these genes from analysis may not be the most viable solution as many of the transcription factors and cell surface markers are sacrificed in this process [37]. Also, variability in dropout rate across individual cells or cell types, works as a

confounding factor for a number of downstream analyses [38, 39]. It has been shown for scRNA-seq datasets that the first principal components highly correlate with proportion of dropouts across individual transcriptomes. So, efficient imputation strategies need to be devised to recover the lost gene-expression for more accurate gene expression measurements in scRNA-seq datasets.

Recent efforts [40] to cater this problem include MAGIC [37], scImpute [41] drImpute [42], deepImpute [43] and SAVER [44]. MAGIC is based on the idea of heat diffusion and uses a neighborhood based affinity matrix to impute the dropouts. It works by sharing information across similar cells. scImpute, first learns each gene's dropout probability in each cell based on a mixture of Gamma and Normal distributions. It then imputes the dropout values in a cell by borrowing information of the same gene in other similar cells, which are selected based on the genes unlikely affected by dropout events. scImpute claims to have better performance than MAGIC. It should be noted that parametric modeling of single cell expression is a challenging task as the sources of technical noise and biases are not known [39]. Also, there is clear lack of consensus about the choice of probability density function. drimpute assumes that the clustering of cells is a true hidden cell classification and the expected value of a dropout event can be obtained by averaging the entries in the given cell cluster. It performs clustering multiple times to identify similar cells, and performs imputation by averaging the expression values from similar cells, followed by averaging multiple estimations for final imputation. DeepImpute is based on deep neural networks and uses dropout layers and loss functions to learn pat-

terns in the data for imputation. SAVER (single-cell analysis via expression recovery), borrows information across genes and cells to provide accurate expression estimates for all genes.

We propose deepMc, a deep Matrix Factorization based imputation technique for scRNA-seq data. Our technique does not assume any distribution for gene expression, outperforms other proposed imputation techniques in most experimental conditions. We believe that superior performancewill make deepMc the method of choice for imputing scRNA-seq data.

2.2 Dataset

We used scRNA-seq datasets from three different studies for performing various experiments.

- **Jurkat-293T data:** This dataset contains expression profiles of Jurkat and 293T cells, mixed *in vitro* at equal proportions (50:50). All $\sim$ 3,300 cells of this data are annotated based on the expressions of cell-type specific markers [45]. Cells expressing CD3D are assigned Jurkat, while those expressing XIST are assigned 293T.

  This dataset is also available at 10x Genomics website.

- **Preimplantation data:** This is an scRNA-seq data of mouse preimplantation embryos. It contains expression profiles of $\sim$ 300 cells from zygote, early 2-cell stage, middle 2-cell stage, late 2-cell stage, 4-cell stage, 8-cell

stage, 16-cell stage, early blastocyst, middle blastocyst and late blastocyst stages. The first generation of mouse strain crosses were used for studying monoallelic expression.

We downloaded the count data from Gene Expression Omnibus (GSE45719) [33].

- **Blakeley:** Single-cell RNA sequencing was performed on a human embryo to define three cell lineages of the human blastocyst [46]: pluripotent epiblast (EPI) cells that form the embryo proper, and extraembryonic trophectoderm (TE) cells and primitive endoderm (PE) cells that contribute to the placenta and yolk sac, respectively. This data with 30 cells, was shared by the authors of [42].

2.3 Methodology

2.3.1 Data preprocessing

Steps involved in preprocessing of raw scRNA-seq data are enumerated below.

- **Data filtering:** If a gene was detected with ≥ 3 reads in at least 3 cells we considered it expressed. We ignored the remaining genes. It should be noted that these are not the biologically silent genes, for which the expression is reduced but not 0.
- **Library-size Normalization:** Expression matrices were normalized by first dividing each read count by the total counts in each cell, and then

by multiplying with the median of the total read counts across cells.

- **Gene Selection:** For each expression data top 1000 high-dispersion (coefficient of variance) genes were kept for imputation and further analyses.
- **Log Normalization:** A copy of the matrices were $\log_2$ transformed following addition of 1 as pseudocount.
- **Imputation:** For various experiments, log transformed expression matrix was used as input for imputation.

2.3.2 Proposed framework

In recent times, deep learning has permeated almost every aspect of computational science. Bioinformatics is not an exception. Min et al. [10] give a comprehensive treatise into the early applications of deep learning in this area. Our current work is motivated by the success of deep matrix factorization [11, 12] and deep dictionary learning [13]. The is to factor the data matrix (which has cells on rows and genes on columns) into several layers of basis and a final layer of coefficients (corresponding to the multiple latent factors); shown here for three levels,

$$X = D_1 D_2 D_3 Z \tag{2.1}$$

Here, X here represents the complete scRNA-seq data matrix and the $D's$ and Z represent the basis and coefficient matrix of X. For visual understanding, please see deepMc architecture shown in Figure 2.1.

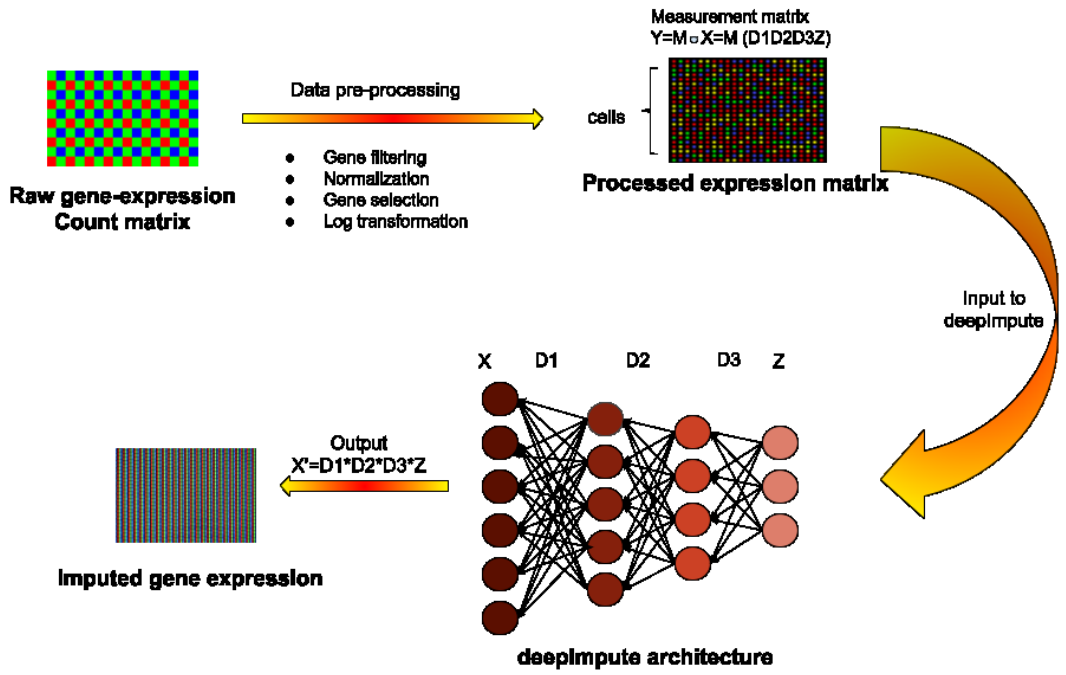


Figure 2.1: Overview of deepMc pipeline for imputing single cell RNA sequencing data.

Note that this is a feedbackward neural network, the connections are from the nodes towards the input. This is because matrix factorization is a synthesis formulation.

Our task is to solve the different layers of basis (D1, D2, D3) and the coefficients (Z) by solving the least squares objective function,

$$\min_{D_1,D_2,D_3,Z} ||Y - M \circ (D_1 D_2 D_3 Z)||_F^2 \tag{2.2}$$

Note that we cannot use techniques derived in [11, 12] for our purpose; this is because they operated on the full data where as we need to derive for partially observed data. The solution for (2.2) is shown below and algorithm is given in

Algorithm 3.

Mathematically we need to solve a problem of the form,

$$Y = M \circ X \tag{2.3}$$

where Y is the measurement, M is the sampling mask and X the matrix to be recovered; $\circ$ is the Hadamard product. Here we model the matrix X as a linear combination (Z) of several layers (3 in this case) of dictionaries (D_1, D_2 and D_3). This is expressed as,

$$X = D_1 D_2 D_3 Z \tag{2.4}$$

We do not explicitly show the activation functions along the layers, but we impose non-negativity constraints on the variables Z, $D_3 Z$ and $D_2 D_3 Z$. This is akin to the rectified linear units (ReLU) used in deep learning. A similar technique was used in [12] The activation function is needed in order to ensure that all the dictionaries do not collapse to a single one.

Incorporating the deep factorization model (2.4) into (2.3) leads to,

$$Y = M \circ (D_1 D_2 D_3 Z) \tag{2.5}$$

The task is to solve for the variables D_1, D_2, D_3 and Z by minimizing the Eu-

clidean norm,

$$\min_{D_1,D_2,D_3,Z} ||Y - M \circ (D_1 D_2 D_3 Z)||_F^2 \text{ s.t.} \\ D_2 D_3 Z > 0 \text{ and } D_3 Z > 0 \tag{2.6}$$

The constraints account for the ReLU type non-linearity between the layers.

For solving (2.6), we will follow the majorization minimization (MM) approach [47].

Let us start with $X = D_1 D_2 D_3 Z$. Therefore we have $J(X) = ||Y - M(X)||_F^2$. The majorizer in the k^{th} iteration will be,

$$G_x(x) = ||Y - M(X)||_F^2 + (X - X_k)^T (I - M^T M)(X - X_k) \\ = YY^T + X_k^T (aI - M^T M) X_k - 2(Y^T M + X_k^T (aI - M^T M))X + aX^T X \tag{2.7}$$

$$= a(-2B^T X + X^T X) + C \tag{2.8}$$

where $B = X_k + \frac{1}{a} M^T (Y - M \circ X_k)$, $C = YY^T + X_k^T (aI - M^T M) X_k$ and a is the maximum eigenvalue of $M^T M$.

Using the identity $||B - X||_F^2 = B^T B - 2B^T X + X^T X$, (2.8) can be expressed as follows,

$$G_x(x) = a||B - X||_F^2 - aB^T B + C \tag{2.9}$$

Instead of minimizing (2.9), one can simply minimize,

$$\begin{aligned} &{G_x}'(x) = ||B - X||_F^2 \\ &\implies X_{k+1} = B = X_k + \tfrac{1}{a} M^T (Y - M \circ X_k) \end{aligned} \tag{2.10}$$

This concludes the first step. In the next, we substitute $Z_1 = D_2 D_3 Z$. This leads to the following objective function,

$$J(Z_1) = ||X_k - D_1(Z_1)||_F^2 \tag{2.11}$$

As before, the majorizer is expressed as (for the l^{th} iteration),

$$G_l(Z_1) = ||X_k - D_1(Z_1)||_F^2 + (Z_1 - Z_{1,l})^T (aI - Z_1^T Z_1)(Z_1 - Z_{1,l}) \tag{2.12}$$

Using the same technique as before, the solution is given by,

$$Z_{1,l+1} = Z_{1,l} + \frac{1}{b} D_1^T (X_k - D_1 Z_{1,l}) \tag{2.13}$$

where b is the maximum eigenvalue of $D_1^T D_1$.

However, we have to ensure that all the coefficients of Z_1 are non-negative; this is ensured by putting the negative entries of (2.15) to zeroes. This non-negativity constraint acts as a relu activation function across the layers and makes sure that all the dictionaries do not collapse into one, hence ensuring non-linearity. For the second layer, i.e. $Z_2 = D_3 Z$, we will have for the m^{th} iteration,

$$Z_{2,m+1} = Z_{2,m} + \frac{1}{c} D_2^T (Z_{1,l} - D_2 Z_{2,m}) \tag{2.14}$$

where c is the maximum eigenvalue of $D_2^T D_2$.

As before, we have to ensure that all Z_2 is non-negative.

For the final layer, we will have the update for the n^{th} iteration,

$$Z_{n+1} = Z_n + \frac{1}{d} D_3^T (Z_{2,m} - D_3 Z_n) \tag{2.15}$$

This concludes the derivation of the algorithm. To prevent degenerate solutions where some of the D's are very high and others low, the columns of all the dictionaries are column normalized (such that each column sums 1) after every update.

This is an iterative solution; since the problem is non-convex, the solution is dependent on initialization. We initialize deterministically. The initial value of X is solved by $\min_X ||Y - M \circ (X)||_F^2$. For D_1, first the SVD of X is computed ($X = USV^T$); D_1 is initialized by the top left eigenvectors of X. For D_2, the SVD of SV^T is computed and the corresponding top eigenvectors are used to initialized D_2. The rest of the dictionaries are initialized in a similar fashion. In the last level, the coefficient (Z) is initialized by the product of the eigenvalues and the right eigenvectors of the last SVD. There can be other randomized techniques for initialization which may yield better results, but our deterministic initialization is repeatable and has shown to yield good results consistently.

Our proposed derivation results in a nested algorithm, i.e. for one update of D_1, the update for D_2 is in a loop; similarly for one update of D_2, the update for

D_3 is in a loop and so on. In a succinct fashion our algorithm can be expressed as Algorithm 3.

Algorithm 3 deepMc

procedure DEEPMC(a, b)
  Initialize: D_1, D_2 and D_3.
    For loop 1, iterate (k)
      $X_{k+1} = X_k + \frac{1}{a} M^T (Y - M \circ X_k)$
      For loop 2, iterate (l)
        $Z_{1,l+1} = Z_{1,l} + \frac{1}{b} D_1^T (X_k - D_1 Z_{1,l})$ ▷ ensure non-negativity
        For loop 3, iterate (m)
          $Z_{2,m+1} = Z_{2,m} + \frac{1}{c} D_2^T (Z_{1,l} - D_2 Z_{2,m})$ ▷ ensure non-negativity
          For loop 4, iterate (n)
            $Z_{n+1} = Z_n + \frac{1}{d} D_3^T (Z_{2,m} - D_3 Z_n)$
          End loop 4
        End loop 3
      End loop 2
    End loop 1

2.4 Results

2.4.1 Contribution

The main contribution of this work is a deep model for matrix completion. All previous techniques for addressing the said problem were shallow; based either on nuclear norm minimization or on matrix factorization. Here we extend the standard (shallow) matrix factorization approach to deeper levels. Instead of decomposing the data matrix into two factors, we decompose into multiple factors (matrices). This is the first work that solves the matrix completion problem by such a deep approach.

The ensuing optimization problem is solved using the majorization minimization approach. The advantage of this approach is that resulting algorithm does not introduce any hyper-parameter. This algorithm has only one parameter and that too can be theoreticaally estimated making the algorithm practically non-parametric. This is stark constrast to every other deep learning model where a significant volume of time needs to be expended in tuning a large number of parameters.

2.4.2 Improvement in clustering accuracy

Clustering single cell RNA-seq data for discovering distinct cell types from a heterogeneous cell population is one of the most important applications of scRNA-seq. But, an algorithm which aims to cluster cells of similar types might get tricked by a large number of dropouts in single cell RNA seq data which serve as biological noise in the input to clustering algorithm. This incorrect view of expression levels should be fixed by a reasonable imputation resulting in accurate delineation of cell types. Hence, we observe the K-means clustering results on all the log-transformed expression profiles for each dataset both without and with imputation.Adjusted Rand Index (ARI) was used as the performance metric to evaluate the correspondence between the original annotations and K-means assigned clusters. The cell type information for each dataset was treated as the clustering ground truth.

Figure 2.2 clearly shows not only that 1 layer matrix factorization based expression re-estimation is the most beneficial as compared to other methods, but also, as we go deeper to 2 and 3 layers, we observe better performance for all datasets.

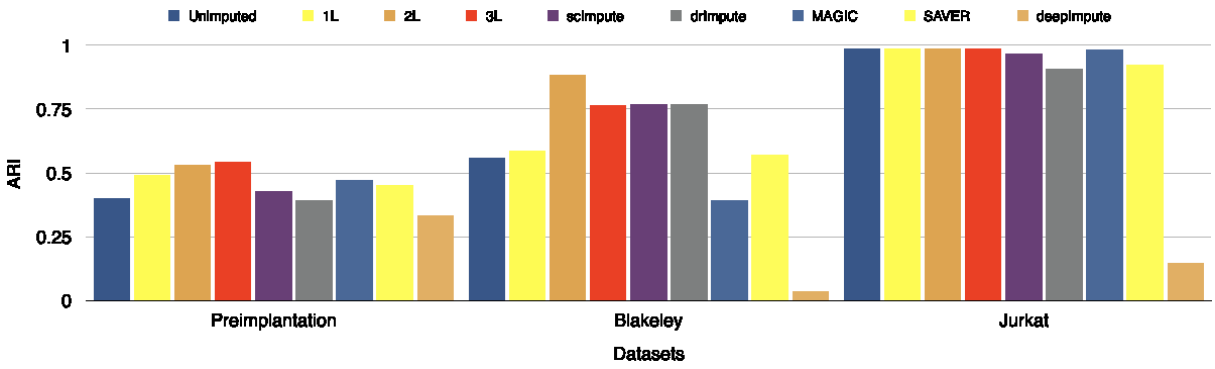


Figure 2.2: ARI values obtained after applying k-means clustering post various imputation techniques.

2.4.3 Improved differential genes prediction

RNA-Seq is widely used in the detection of differentially expressed genes (DEGs) [48]. A good imputation method should result in better congruence between scRNA-seq and bulk RNA-seq data of the same biological condition on differentially expressed genes [36].

To assess the accuracy of differential expression (DE) analysis, we used the standard Wilcoxon Rank-Sum test for identifying differentially expressed genes from matrices obtained from various imputation methods. The dataset of pluripotent stem cells [49] (having matching bulk RNA-Seq data) generated from different individuals was used. The authors call it Tung dataset. They identify DE and non-DE genes using three standard methods: limma-voom [50, 51],

edgeR [52] and DESeq2 [53] [1].

We show the agreement between bulk and single cell based DE calls using the Area Under the Curve (AUC) values obtained from the Receiver Operating Characteristic (ROC) curve (figure 2.3). The 2-layer (2L) deepMc imputation shows the best performance in predicting differentially expressed genes.

For each method, the AUC value was computed on the identical set of ground truth genes. We had to make an exception only for drImpute as it applies an additional filter to prune genes. Hence, the AUC value was computed based on a smaller set of ground truth genes for drImpute .

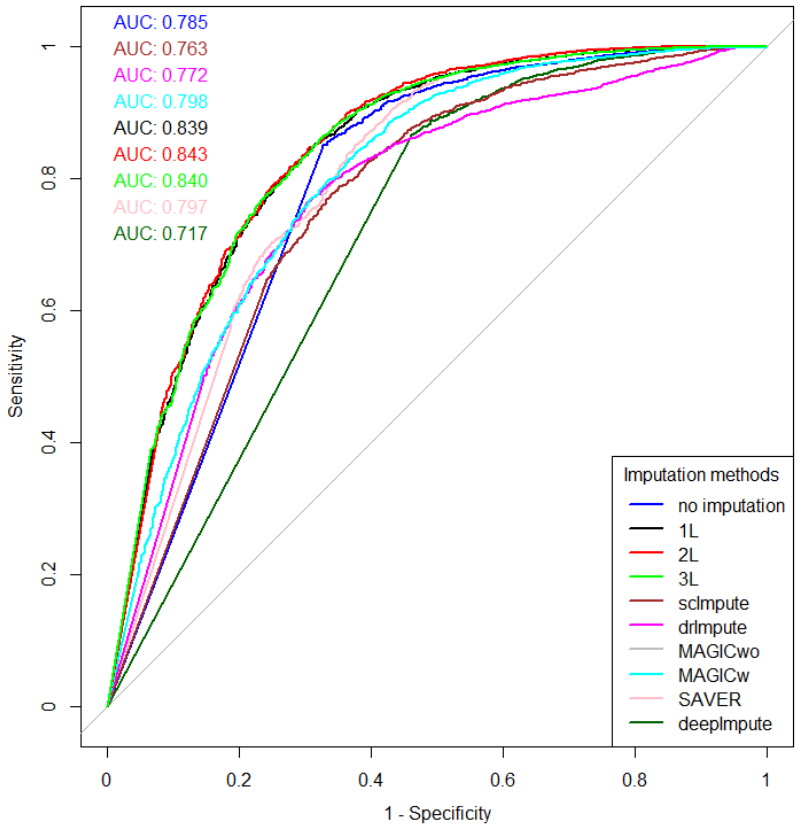


Figure 2.3: Plot showing ROC curve and AUC depicting how well do the DE genes predicted from scRNA and matching bulk RNA-Seq data agree. DE calls were made on expression matrix imputed using various methods.

[1] https://github.com/hemberg-lab/scRNA.seq.course

2.4.4 Improvement in cell type separability

Before explaining this analysis, we introduce the following terms:

1. **Intra-cell type scatter**: For any two cell groups, we first find the median of Pearson's correlation values computed for each possible pair of cells within their respective groups. We define the average of the median correlation values as the intra-cell type scatter.

2. **Inter-cell type scatter**: is defined as the median of Pearson's correlation values computed for pairs such that in each pair, cells belong to two different groups.

3. **Cell-type separability (CTS) score:** The difference between the intra-cell scatter and inter-cell type scatter is termed as the cell-type separability (CTS) score.

An effective imputation should lead to a higher CTS score, showing that expression similarities between cells of identical type are considerably higher than that of cells coming from different subpopulations. DeepImpute proves to be highly beneficial for all 3 datasets, with 2-layer (2L) and 3-layer (3L) deepMc algorithms giving the highest CTS score, validating our imputation strategy. We show the variation of CTS with various imputation strategies in figure 2.4 and observe that our algorithms best separates the cell types, giving the highest CTS.

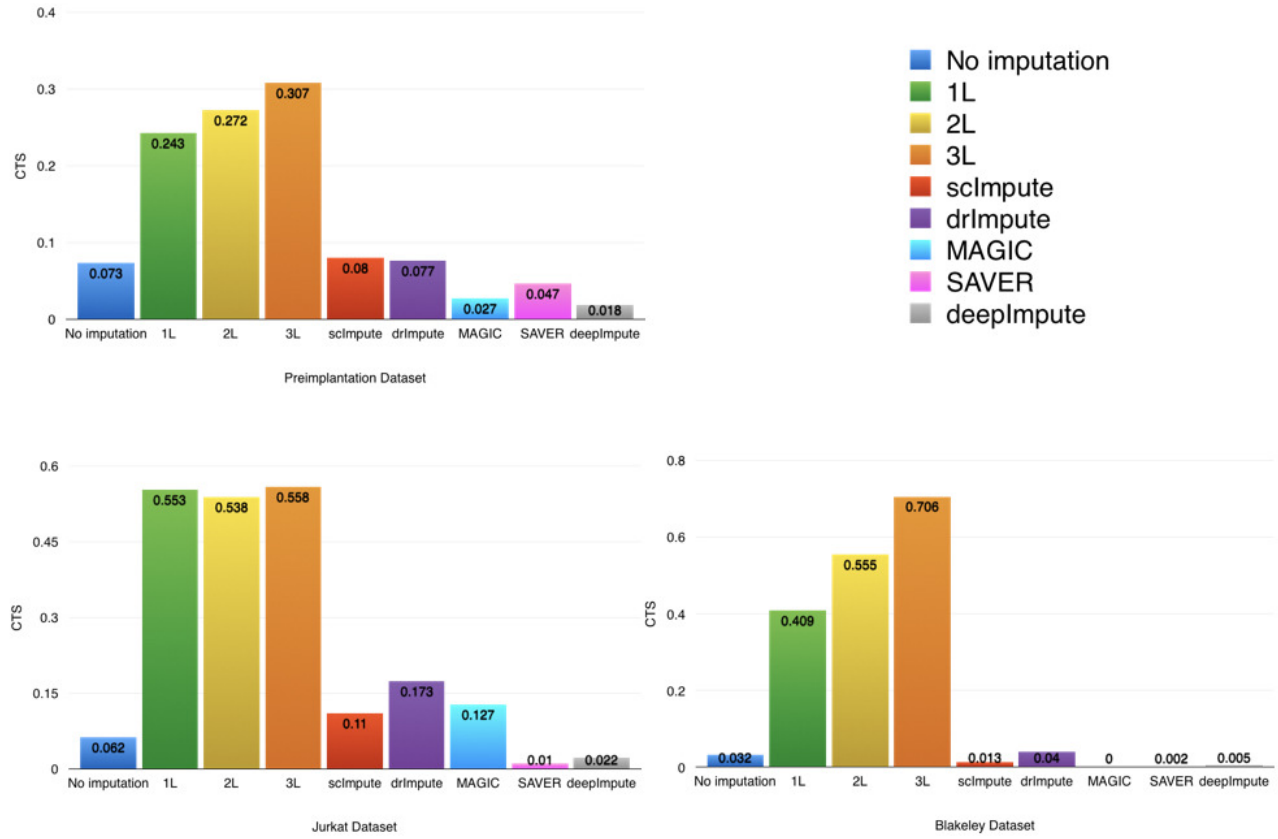


Figure 2.4: Plot showing the variation of CTS with various imputation strategies.

2.5 Conclusion

Single cell RNA-seq has inspired new discoveries and innovation in the field of developmental and cell biology over the past few years and is useful for studying cellular responses at individual cell resolution. But, due to the paucity of starting RNA, the data acquired has dropouts. To address this, we propose a deep matrix factorization based method, deepMc, to impute missing values in gene-expression data. For the deep architecture of our approach, we draw our motivation from great success of deep learning in solving various machine

learning problems. In this work, we support our method with positive results on several evaluation metrics like clustering of cell populations, differential expression analysis and cell type separability.

Chapter 3

Drug-target interaction prediction using multi graph regularized nuclear norm minimization

The identification of interactions between drugs and target proteins is crucial in pharmaceutical sciences. The experimental validation of interactions in genomic drug discovery is laborious and expensive; hence, there is a need for efficient and accurate in-silico techniques which can predict potential drug-target interactions to narrow down the search space for experimental verification.

In this work, we propose a new framework, namely, Multi Graph Regularized Nuclear Norm Minimization, which predicts the interactions between drugs and proteins from three inputs: known drug-target interaction network, similarities over drugs and those over targets. The proposed method focuses on finding a low-rank interaction matrix that is structured by the proximities of drugs and targets encoded by graphs. Previous works on Drug Target Interaction (DTI)

prediction have shown that incorporating drug and target similarities helps in learning the data manifold better by preserving the local geometries of the original data. But, there is no clear consensus on which kind and what combination of similarities would best assist the prediction task. Hence, we propose to use various multiple drug-drug similarities and target-target similarities as multiple graph Laplacian (over drugs/targets) regularization terms to capture the proximities exhaustively.

Extensive cross-validation experiments on four benchmark datasets using standard evaluation metrics (AUPR and AUC) show that the proposed algorithm improves the predictive performance and outperforms recent state-of-the-art computational methods by a large margin.

3.1 Introduction

The field of drug discovery in Pharmaceutical Sciences is plagued with the problem of high attrition rate. The task is to find effective interactions between chemical compounds (drugs) and amino-acid sequences/ proteins (targets). This is traditionally done through wet-lab experiments which are known to be costly and laborious. An effective and appropriate alternative to avoid costly failures is to computationally predict the interaction probability. A lot of algorithms have been proposed for DTI (Drug-target interaction) prediction in recent years [54, 55], which use small number of experimentally validated interactions in existing databases such as ChEMBL [56], DrugBank [57], KEGG DRUG [58],

STITCH [59] and SuperTarget [60]. Identification of drug-target pairs leads to improvements in different research areas such as drug discovery, drug repositioning, polypharmacology, drug resistance and side-effect prediction [61]. For instance, Drug repositioning [62, 63] (reuse of existing drugs for new indications) may contribute to its polypharmacology (i.e. having multiple therapeutic effects). One of the many successfully repositioned drugs is Gleevec (imatinib mesylate). It was originally thought to interact only with the Bcr-Abl fusion gene associated with leukemia but later, it was found to also interact with PDGF and KIT, eventually leading it to be repositioned to treat gastrointestinal stromal tumors as well [64, 65].

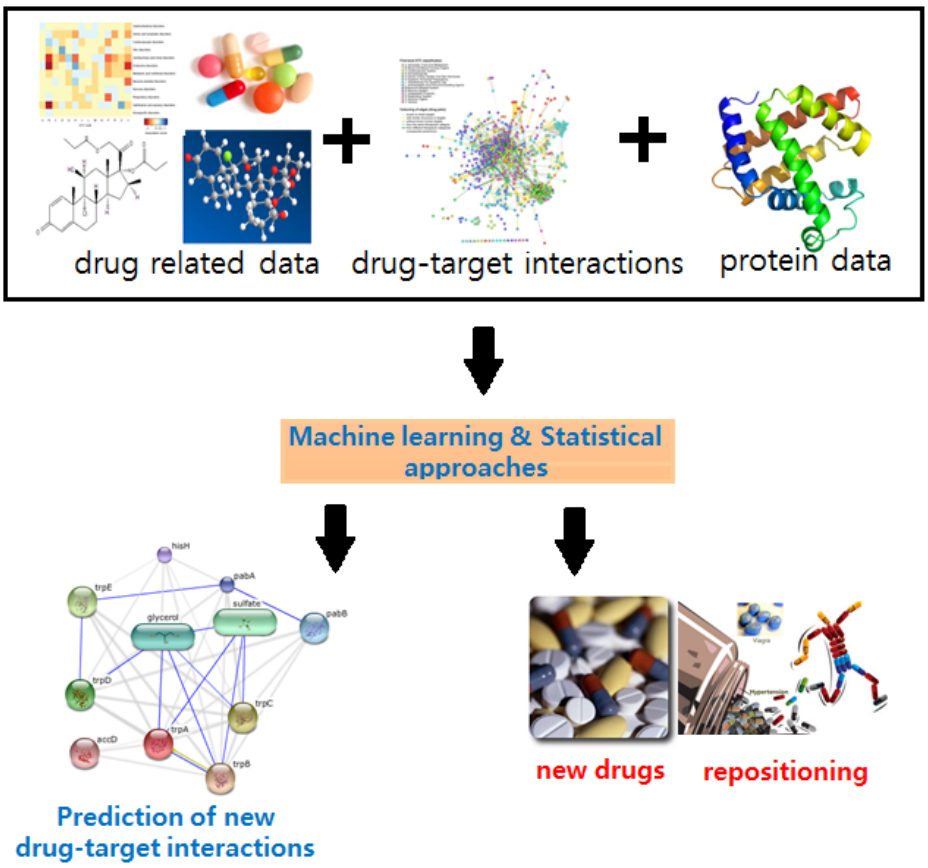


Figure 3.1: Drug repositioning using DTI

There are three major classes of computational methods for predicting DTI: Ligand-based approaches, Docking based approaches, and Chemogenomic approaches. Ligand-based approaches leverage the similarity between target proteins' ligands to predict interactions [66]. These approaches use the fact that similar molecules tend to share similar properties and usually bind similar proteins [67]. However, lack of known ligands per protein in some cases might compromise the reliability of results. Docking-based approaches are well-accepted and utilize the 3D structure information of a target protein and a drug; and then run a simulation to estimate the likelihood that they will interact or not [68, 69, 70]. But docking is heavily time-consuming and cannot be applied to protein families for which the 3D structure is difficult to predict or is unavailable [71] for example the G-protein coupled receptors (GPCRs).

Chemogenomic approaches overcome the challenges of traditional methods and thus, have recently gained much attention. The approaches under this category work with widely abundant biological data, publicly available in existing online databases and process information (chemical structure graphs and genomic sequences for the drugs and targets) from both the drug and target sides simultaneously for the prediction task. These approaches can further subclassified based on the representation of the input data: Feature-based methods and Similarity-based methods. Feature-based techniques are machine learning methods, which take their inputs in the form of feature vectors, representing a set of instances (i.e. drug-target pairs) along with their corresponding class labels (i.e. binary values indicating whether or not an interaction exists). Ex-

amples of typical feature based methods include Decision Tree (DT), Random Forest (RF) [25] and Support Vector Machines (SVM) to build classification models based on the labeled feature vectors [72]. Positive instances are the known interactions and negative instances, the non-interactions. It should be noted that negative instances here include both non-interactions and unknown drug-target interactions (false negatives). The other category of chemogenomic techniques, Similarity-based methods, use two similarity matrices corresponding to drug and target similarity, respectively, along with an interaction matrix which indicates which pairs of drugs and targets interact.

In a very recent review paper [55] it was empirically shown that matrix factorization based techniques yields by far the best results. The fundamental assumption behind matrix factorization to work is that there are very few (latent) factors that are responsible for drug target interactions. This is the reason, one can factor the DTI matrix into a tall (drug) latent factor matrix and a fat (target) latent factor matrix. Mathematically speaking, the assumption is that the DTI matrix is of low-rank. Matrix factorization is being used to model low-rank matrices for the past two decades since the publication of Lee and Seung's seminal paper [73]. However, matrix factorization is a bi-linear non-convex problem; there are no convergence guarantees. In order to ameliorate this problem, mathematicians proposed an alternate approach based on nuclear norm minimization [74, 75, 76]. The nuclear norm is the closest convex surrogate to the rank min-

imization (known to be NP-hard) problem and there are provable mathematical guarantees on its equivalence to rank minimization.

The standard versions of both the matrix factorization and nuclear norm minimization techniques are unable to incorporate similarity information of the drugs and the targets. In recent studies [14, 15], it was shown that the best results are obtained when these technique incorporate graph regularization penalties into them. But, these works regularize the objective function by taking into account, just the standard chemical structure similarity for drugs (S_d) and the genomic sequence similarity for targets (S_t). No study in literature gives a clear picture of which kind of similarities would be the best for DTI prediction. We, therefore, incorporate different other kinds of similarities and a combination of them as a multi graph Laplacian regularization with Nuclear Norm Minimization for DTI prediction. The algorithm uses four new similarity measures over the drugs and targets, apart from the standard similarities to construct the graph Laplacians. The four newly incorporated similarities are computed from the interaction matrix and take into account the Cosine similarity, Correlation, Hamming distance and Jaccard similarity between the drugs and targets. To the best of our knowledge, this is the first work on multiple graph laplacian regularized nuclear norm minimization for DTI prediction.

3.2 Dataset

We use the four benchmark datasets introduced in [77] concerning four different classes of target proteins, namely, enzymes (Es), ion channels (ICs), G protein-coupled receptors (GPCRs) and nuclear receptors (NRs). The data was simulated from public databases KEGG BRITE [78], BRENDA [79] SuperTarget [60] and DrugBank [57] and is publically available at http://web.kuicr.kyoto-u.ac.jp/supp/yoshi/drugtarget/..

The data gathered from these databases is formatted as an adjacency matrix, called interaction matrix between drugs and targets, encoding the interaction between n drugs and m targets as 1 if the drug d_i and target t_j are known to interact and 0, otherwise.

Along with the interaction matrix, drug similarity matrix S_d and a target similarity matrix S_t are also provided. In S_d, each entry represents the pairwise similarity between the drugs and is measured using SIMCOMP [80]. It represents the chemical structure similarity computed by the number of shared substructures in chemical structures between two drugs. In S_t, the similarity score between two proteins is the genomic sequence similarity. It is based on the amino acid sequences of the target protein and is computed using normalized Smith–Waterman [81].

The similarity matrices S_d and S_t constitute the most standard similarities that have been used in the DTI prediction task hitherto. We use these similarities along with the following four more similarities computationally derived from

the drug-target interaction matrix to form the graph laplacian terms:

- Cosine similarity: measures the cosine of the angle between two drug/target vectors projected in a multi-dimensional space. Its value ranges from -1 (exactly opposite) to 1 (exactly the same). Given two n-dimensional drug/target vectors, the cosine similarity is calculated as follows,

$$S^{\cos} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

- Correlation: computes the Pearson's linear correlation coefficient indicating the extent to which two variables are linearly related. It has a value between +1 and 1, where 1 is total positive linear correlation, 0 is no linear correlation, and 1 is total negative linear correlation. For a pair (say A and B) of drugs/targets with sample size n, it is given by,

$$S^{cor} = \frac{\sum_{i=1}^{n} (A_i - \overline{A})(B_i - \overline{B})}{\sqrt{\sum_{i=1}^{n} (A_i - \overline{A})^2}\sqrt{\sum_{i=1}^{n} (B_i - \overline{B})^2}}$$

  where

$$\overline{A} = \frac{1}{n}\sum_{i=1}^{n} A_i \text{ and } \overline{B} = \frac{1}{n}\sum_{i=1}^{n} B_i$$

- Hamming similarity: has been computed using hamming distance. For any two drugs/targets, the hamming distance is the percentage of interaction positions that differ. We calculate Hamming distance based similarity by

simply subtracting hamming distance from 1, giving us its complementary (the percentage of common interaction positions for a pair of drugs/targets). It can be calculated as follows,

$$S^{ham} = 1 - \frac{\#(A_i \neq B_i)}{n}$$

- Jaccard similarity: is defined as the percentage of common non-zero interaction positions for the two given sample sets of drugs/target,

$$S^{jac} = \frac{\#[(A_i = B_i) \cap ((A_i \neq 0) \cup (B_i \neq 0))]}{\#[(A_i \neq 0) \cup (B_i \neq 0)]}$$

Table 3.1 summarizes the features of all four datasets.

3.3 Methodology

Let us assume that X is the adjacency matrix where each entry denotes interaction between a drug and target (1 if they interact, 0 otherwise). Unfortunately, we only observe this matrix partially because all interactions are not known. If M denotes the partially observed adjacency matrix, the the problem of recovering X from its partially observed entries can be solved via Nuclear Norm minimization (section 1.1.2).

Table 3.1: Drugs, Targets and Interactions in each dataset used for validation.

Datasets	NR	GPCR	IC	E
Drugs	54	223	210	445
Targets	26	95	204	664
Interactions	90	635	1476	2926

3.3.1 Data preprocessing

Each of the drug and target similarity matrices were summed up to compute the combined similarity matrices S_d^{COM} and S_d^{COM} (equation (3.13)). We then follow the below steps for preprocessing the interaction data,

- SIMILARITY SPARSIFICATION: The combined similarity matrices were further sparsified by using p-nearest neighbor graph which is obtained by keeping only the similarity values of the nearest neighbors for each drug/-target in the similarity matrices. The usage of such a pre-processing, as shown by [14], helps learn a manifold on or near to which the data is assumed to lie which, in turn, is expected to preserve the local geometries of the original data and hence give more accurate results.

$$\forall i, j$$
$$N_{ij} = \begin{cases} 1, j \in N_p(i) \& i \in N_p(j) \\ 0, j \notin N_p(i) \& i \notin N_p(j) \\ 0.5, else \end{cases}$$

  where $Np(i)$ is the set of p nearest neighbors to drug d_i. Similarity matrix sparsification is done by element-wise multiplying it with N_{ij}. In the next step, the combined graph laplacian terms are computed.

- NORMALIZATION OF GRAPH LAPLACIANS: Instead of the graph laplacians ($L_{d/t}^{COM}$ and L_t^{COM}), we have used the normalized versions of them ($(D_d^{COM})^{-1/2} L_d^{COM} (D_d^{COM})^{-1/2}$ and $(D_t^{COM})^{-1/2} L_t^{COM} (D_t^{COM})^{-1/2}$) in-

stead as normalized graph Laplacians are known to perform better than their un-normalized counterparts [82].

3.3.2 Proposed framework

Nuclear Norm based Low-rank Matrix Completion has been around since the past decade. The problem with standard Nuclear norm minimization (NNM) is that it cannot accommodate associated information such as Similarity matrices for Drugs and Targets. But, it has been seen in recent studies that accommodating the similarity information is crucial for improving the DTI prediction results. The current works have incorporated the standard similarity measures for drugs and targets in matrix factorization [14] and Matrix completion [15] frameworks. It is imperative that NNM should be capable of taking into account more types and combinations of similarities. To achieve this, we have augmented four other types of similarities between drugs/targets and presented Multi Graph regularized Nuclear Norm Minimization (MGRNNM).

Graph regularization assumes that points close to each other in the original space should also be close to each other in the learned manifold (**Local Invariance assumption**). So, Graph regularization would allow the algorithm to learn manifolds for the drug and target spaces in which the data is assumed to lie.The multi graph regularized version of Nuclear norm minimization, aims to prevent over fitting and greatly enhance the generalizing capabilities. It is incorporated into the formulation/objective function as Laplacian weights corresponding to

drugs and targets as follows,

$$\min_{X} ||A - M \circ (X)||_F^2 + \lambda||X||_* + \mu_1 Tr(X^T \sum_{i=1}^{nos} L_d^i X) + \mu_2 Tr(X \sum_{i=1}^{nos} L_t^i X^T) \tag{3.1}$$

where $\lambda \geq 0$, $\mu_1 \geq 0$ and $\mu_2 \geq 0$ are parameters balancing the reconstruction error of NNM in the first two terms and graph regularization in the last two terms, $Tr(.)$ is the trace of a matrix, nos stands for number of similarity matrices ($nos = 5$ in our case) while recovering complete DTI matrix X (with drugs on rows and target proteins on columns) from its sampled version A.

If, say we consider a single similarity matrix for drugs (S_d) and that for targets (S_t), then $L_d = D_d - S_d$ and $L_t = D_t - S_t$ are the graph Laplacians [17] for S_d (drug similarity matrix) and S_t (target similarity matrix), respectively, and $D_d^{ii} = \Sigma_j S_d^{ij}$ and $D_t^{ii} = \Sigma_j S_t^{ij}$ are degree matrices.

We employ Problem (3.1) is solved using a variable splitting approach [83]. The augmented Lagrangian is expressed as (3.2). We introduce two new proxy variables Z and Y such that $Z^T = X$ and $Y = X$,

$$\min_{X,Y,Z} ||A - M \circ (X)||_F^2 + \lambda||X||_* + \mu_1 Tr(Z \sum_{i=1}^{nos} L_d^i Z^T) + \mu_2 Tr(Y \sum_{i=1}^{nos} L_t^i Y^T) + \nu_1||Z^T - X||_F^2 + \nu_2||Y - X||_F^2 \tag{3.2}$$

The variables are updated using ADMM (alternating direction method of multipliers) [84, 85] where we divide the problem into sub problems which are easier to solve. This leads to the following subproblems (3.3), (3.4) and (3.5),

$$X \leftarrow \min_X ||A - M \circ (X)||_F^2 + \nu_1 ||Z^T - X||_F^2 + \nu_2 ||Y - X||_F^2 + \lambda ||X||_* \quad (3.3)$$

$$Y \leftarrow \min_Y \mu_2 Tr(Y \sum_{i=1}^{nos} L_t^i Y^T) + \nu_2 ||Y - X||_F^2 \quad (3.4)$$

$$Z \leftarrow \min_Z \mu_1 Tr(Z \sum_{i=1}^{nos} L_d^i Z^T) + \nu_1 ||Z^T - X||_F^2 \quad (3.5)$$

Problem (3.3) can be expressed as a standard NNM probelm (by column stacking the variables),

$$\left\| \begin{pmatrix} A \\ \sqrt{\nu_1} Z^T \\ \sqrt{\nu_2} Y \end{pmatrix} - \begin{pmatrix} M \\ \sqrt{\nu_1} I \\ \sqrt{\nu_2} I \end{pmatrix} X \right\|_F^2 + \lambda ||X||_* \quad (3.6)$$

To solve for Y and Z, we differentiate (3.4) and (3.5) wrt Y and Z, respectiveley,

$$Y = {}_Y(F_1) \text{ where } F_1 = \mu_2 Tr(Y \sum_{i=1}^{nos} L_t^i Y^T) + \nu_2 ||Y - X||_F^2 \quad (3.7)$$

$$Z = \underset{Z}{}(F_2) \text{ where } F_2 = \mu_1 Tr(Z \sum_{i=1}^{nos} L_d^i Z^T) + \nu_1 ||Z^T - X||_F^2 \tag{3.8}$$

$$\frac{\partial F_1}{\partial Y} = \mu_2 (Y (\sum_{i=1}^{nos} L_t^i)^T + Y \sum_{i=1}^{nos} L_t^i) + 2\nu_2 (Y - X) \tag{3.9}$$

$$\frac{\partial F_1}{\partial Y} = \mu_2 Y [\sum_{i=1}^{nos} (L_t^{iT} + L_t^i)] + 2\nu_2 (Y - X) \tag{3.10}$$

Since L_t is a symmetric matrix, $L_t^T = L_t$. So,

$$\frac{\partial F_1}{\partial Y} = 2\mu_2 Y \sum_{i=1}^{nos} L_t^i + 2\nu_2 (Y - X)$$

Equating the derivative to zero, we get,

$$\nu_2 Y + \mu_2 Y \sum_{i=1}^{nos} L_t^i = \nu_2 X \tag{3.11}$$

The matrix equation of this form (AT+TB=C) cannot be solved directly for variable T and is called Sylvester equation. Such an equation has a unique solution when the eigenvalues of A and -B are distinct.

A similar Sylvester equation and update step for Z can be obtained by differentiating F_2 and equating to 0,

$$\nu_1 Z + \mu_1 Z \sum_{i=1}^{nos} L_d^i = \nu_1 X^T \tag{3.12}$$

It can be shown that computing the sum of the Graph Laplacians is equivalent to computing the Laplacian from the sum of various similarity matrices involved. For instance, consider the sum of drug Graph Laplacians,

$$\begin{aligned}
&\sum_{i=1}^{nos} L_d^i \\
&= \sum_{i=1}^{nos} (D_d^i - S_d^i) \\
&= \sum_{i=1}^{nos} D_d^i - \sum_{i=1}^{nos} S_d^i \\
&= \sum_{i=1}^{nos} diag(\sum_j S_d^j) - \sum_{i=1}^{nos} S_d^i \\
&= diag(\sum_j (\sum_{i=1}^{n} S_d^i)^j) - \sum_{i=1}^{nos} S_d^i
\end{aligned}$$

Let $\sum_{i=1}^{nos} S_d^i = S_d^{COM}$ where S_d^{COM} stands for combined similarity for drugs. Essentially,

$$S_d^{COM} = S_d + S_d^{cos} + S_d^{cor} + S_d^{ham} + S_d^{jac} \tag{3.13}$$

Then,

$$\sum_{i=1}^{nos} L_d^i = diag(\sum_j S_d^{COM}) - S_d^{COM} = D_d^{COM} - S_d^{COM} = L_d^{COM} \tag{3.14}$$

Here, D_d^{COM} and L_d^{COM} denote combined degree matrix and combined Laplacian matrix (sum of graph laplacians) for drugs. Of note, the individual Laplacians or the similarities can be weighted unequally to give more or less emphasis on a specific type of similarity. The pseudo-code for MGRNNM has been

given in Algorithm 4.

The standard NNM is a convex problem and the introduced graph regularization penalties are also convex, so entire formulation (5), being a sum of convex functions, is convex. Therefore it is bound to converge to a global minima. We chose the number of iterations such that the algorithm halts when the objective function does not change with iterations.

Algorithm 4 Multi Graph regularized Nuclear Norm Minimization

procedure MGRNNM($A, M, S_d^{COM}, S_t^{COM}$)
  Sparsify: S_d^{COM}, S_t^{COM}
  Initialize: $\lambda, \mu_1, \mu_2, \nu_1, \nu_2, L_d^{COM}, L_t^{COM}, Y = A, Z = A^T$
    $MM \leftarrow \begin{pmatrix} M \\ \sqrt{\nu_1} I \\ \sqrt{\nu_2} I \end{pmatrix}$
  For loop 1, iterate (k)
    $YY_k \leftarrow \begin{pmatrix} A \\ \sqrt{\nu_1} Z^T \\ \sqrt{\nu_2} Y \end{pmatrix}$
    $X_k \leftarrow MATRIX - SVS(YY_k, MM, \lambda)$
    $Y_k \leftarrow$ solve-sylvester$(\nu_1 I, \mu_1 L_d^{COM}, \nu_1 X_k')$
    $Z_k \leftarrow$ solve-sylvester$(\nu_2 I, \mu_2 L_t^{COM}, \nu_2 X_k)$
  End loop 1

3.4 Results

3.4.1 Experimental setup

We validated our proposed method by comparing it with recent and well-performing prediction methods proposed in the literature. Out of the 5 approaches with which we compare,

- Three are specifically designed for DTI task (WGRMF: Weighted Graph Regularized Matrix Factorization, CMF: Collaborative Matrix Factoriza-

tion, RLS_WNN: Regularized Least square Nearest neighbor profile) [14, 86, 87];

- One being traditional matrix completion (MC: matrix completion) [88] and
- Last one being a naive solution to our problem, available as an unpublished work (MCG: matrix completion on graphs). Of note, the Space complexity of MCG is $O(n^4)$ while that of MGRNNM is $O(n^2)$. [89])

All baselines designed for DTI problem are recent and are already compared against older methods.

We performed 5 repetitions of 10-fold cross-validation (CV) for each of the methods under three cross-validation setting (CVS) [55]:

- CVS1/Pair prediction: random drug–target pairs are left out as the test set for prediction. It is the conventional setting for validation and evaluation.
- CVS2/Drug prediction: entire drug profiles are left out to be used as test set. It tests the algorithm's ability to predict interactions for novel drugs i.e. drugs for which no interaction information is available.
- CVS3/Target prediction: entire target profiles are left out to be used as test set. It tests the algorithm's ability to predict interactions for novel targets.

κ-fold cross validation is an evaluation method where we divide the data into κ equal subsets (called folds). Out of all the subsets, 1 of them is treated as a testing set, while the remaining $\kappa - 1$ ones constitute the training set.

As the evaluation metrics, we used:

- AUC: AUC stands for *Area under the ROC Curve*. That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (a plot showing the true positive rate for a method as a function of the false positive rate). AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is the probability that the model ranks a random positive example more highly than a random negative example. The higher it is, the better the model is.

- AUPR: We also evaluated the performance by AUPR (Area Under the Precision-Recall curve), because AUPR punishes highly ranked false positives much more than AUC, this point being important practically since only highly ranked drug-target pairs in prediction will be biologically or chemically tested later in an usual drug discovery process, meaning that highly ranked false positives should be avoided [90, 14]. The precision-recall curve shows the tradeoff between precision and recall for different thresholds. A high area under the curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate. High scores for both show that the classifier is returning accurate results (high precision), as well as returning a majority of all positive results (high recall).

3.4.2 Parameter settings

For setting the parameters of our algorithm, we performed cross-validation on the training set on the parameters $p, \lambda, \mu_1, \mu_2, \nu_1, \nu_2$ to find the best parameter combination for each dataset, under each cross-validation setting. As mentioned earlier, the individual laplacians or the similarities can be weighted unequally to give more or less emphasis on a specific type of similarity, we weigh the Cosine, Correlation and Jaccard similarities heavily (4 times) relative to Hamming similarity. This was done because hamming similarity showed the least improvement in prediction accuracy as compared to the other three similarities when taken into account along with standard similarities (Refer Figure 3.2). For the other methods, we to set the parameters to their optimal (which were found to be already optimal) in [55].

,

3.4.3 Interaction prediction

Tables 3.2, 3.4 and 3.6 show the AUPR results and Tables 3.3, 3.5 and 3.7 show the AUC results from the above-mentioned cross validation settings. MGRNNM outperforms the state-of-the-art prediction methods. The second column in each table shows the results of our algorithm when only the standard similarity matrices (S_d: chemical structure similarity for drugs, S_t: Genomic sequence similarity for target proteins) were used for prediction.

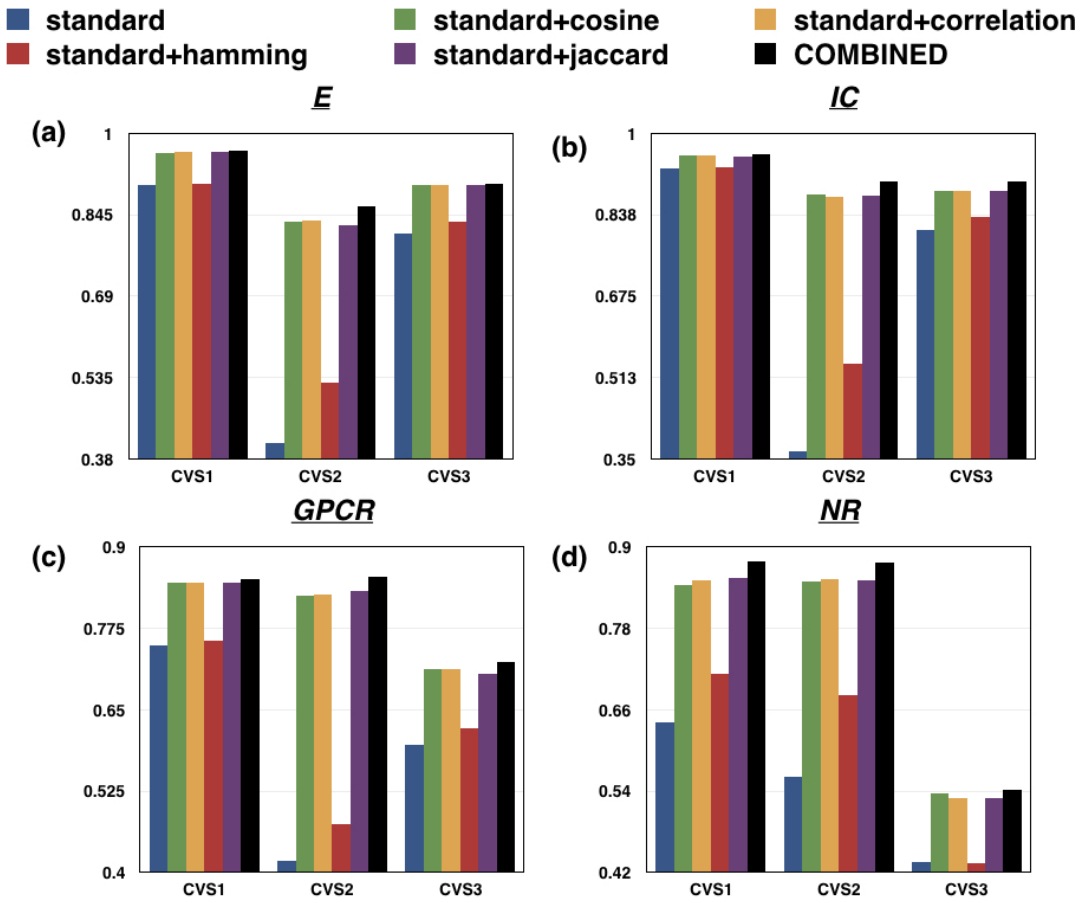


Figure 3.2: Bar plots depicting that incorporating all the similarities for drugs and targets for prediction task yields best results for every dataset **(a)** E **(b)** IC **(c)** GPCR and **(d)** NR under the three cross-validation settings in comparison to the cases where each type of similarity was considered separately. Here, *standard* represents the case when only the chemical structure similarity for drugs and genomic sequence similarity for targets were taken into account and *COMBINED* refers to the use case where all the similarity matrices (standard similarity, Cosine similarity, Correlation, Hamming similarity and Jaccard similarity) were considered.

3.4.4 Validation of multiple similarities

To precisely analyze the consequence of multiple similarities incorporation, we observed the mean AUPR for several cases:

- standard: When only the standard similarity matrices (S_d: chemical structure similarity for drugs, S_t: Genomic sequence similarity for target proteins) were used for prediction.
- standard+Cosine: When Cosine similarity between each pair of drugs/tar-

gets (S_d^{cos}, S_t^{cos}) was taken into account along with standard similarities.

- standard+Correlation: When Pearson's linear Correlation between each pair of drugs/targets (S_d^{cor}, S_t^{cor}) was taken into account along with standard similarities.
- standard+Hamming: When Hamming similarity between each pair of drugs/-targets (S_d^{ham}, S_t^{ham}) was taken into account along with standard similarities.
- standard+Jaccard: When Jaccard similarity between each pair of drugs/targets (S_d^{jac}, S_t^{jac}) was taken into account along with standard similarities.
- COMBINED: When all five similarity types between each pair of drugs/-targets (S_d^{COM}, S_t^{COM}) were taken into account.

The analysis was carried out for every dataset under all the three cross-validation settings. Figure 3.2 clearly depicts that incorporating all the similarities for drugs and targets for prediction task yields the best results.

3.5 Conclusion

Drug-target interaction prediction is a crucial task in genomic drug discovery. Many computational techniques have been proposed in the literature. In this work, we presented a novel chemogenomic approach for predicting the drug-target interactions, MGRNNM (Multi Graph regularized Nuclear Norm Minimization). It is a graph regularized version of the traditional Nuclear Norm

Table 3.2: AUPR results for interaction prediction under validation setting CVS1.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.9660 (0.0006)**	0.9014 (0.0018)	0.7882 (0.0022)	0.7621 (0.0025)	0.8768 (0.0020)	0.8093 (0.0045)	0.8837 (0.0026)
IC	**0.9585 (0.0013)**	0.9298 (0.0026)	0.8868 (0.0028)	0.8346 (0.0025)	0.9225 (0.0022)	0.8459 (0.0106)	0.9373 (0.0019)
GPCR	**0.8515 (0.0033)**	0.7483 (0.0039)	0.6481 (0.0116)	0.5956 (0.0102)	0.7370 (0.0024)	0.6933 (0.0226)	0.7543 (0.0017)
NR	**0.8791 (0.0019)**	0.6408 (0.0234)	0.3950 (0.0298)	0.4558 (0.0202)	0.6016 (0.0378)	0.7072 (0.0290)	0.6383 (0.0149)

Table 3.3: AUC results for interaction prediction under validation setting CVS1.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.9955 (0.0003)**	0.9798 (0.0004)	0.8753 (0.0023)	0.9596 (0.0015)	0.9647 (0.0013)	0.9635 (0.0014)	0.9705 (0.0013)
IC	**0.9947 (0.0004)**	0.9829 (0.0012)	0.9415 (0.0015)	0.9539 (0.0010)	0.9747 (0.0022)	0.9786 (0.0026)	0.9832 (0.0008)
GPCR	**0.9785 (0.0020)**	0.9531 (0.0028)	0.8110 (0.0055)	0.8977 (0.0047)	0.9432 (0.0010)	0.9458 (0.0044)	0.9493 (0.0031)
NR	**0.9660 (0.0056)**	0.9083 (0.0058)	0.5882 (0.0253)	0.8315 (0.0165)	0.8892 (0.0153)	0.9329 (0.0114)	0.8679 (0.0124)

Table 3.4: AUPR results for interaction prediction under validation setting CVS2.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.8603 (0.0095)**	0.4089 (0.0104)	0.0114 (0.0005)	0.0457 (0.0008)	0.4019 (0.0128)	0.2409 (00272)	0.3848 (0.0094)
IC	**0.9026 (0.0197)**	0.3650 (0.0178)	0.0473 (0.0035)	0.0925 (0.0013)	0.3666 (0.0169)	0.3090 (0.0200)	0.3538 (0.0137)
GPCR	**0.8538 (0.0112)**	0.4175 (0.0076)	0.0404 (0.0017)	0.1091 (0.0044)	0.4247 (0.0113)	0.3463 (0.0106)	0.4059 (0.0104)
NR	**0.8773 (0.0125)**	0.5620 (0.0262)	0.1120 (0.0206)	0.2404 (0.0337)	0.5695 (0.0136)	0.5373 (0.0216)	0.5203 (0.0250)

Table 3.5: AUC results for interaction prediction under validation setting CVS2.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.9460 (0.0033)**	0.8260 (0.0108)	0.5060 (0.0090)	0.7413 (0.0118)	0.7982 (0.0144)	0.7755 (0.0093)	0.7952 (0.0110)
IC	**0.9714 (0.0095)**	0.7913 (0.0090)	0.5512 (0.0034)	0.7196 (0.0071)	0.7902 (0.0149)	0.7669 (0.0140)	0.7576 (0.0125)
GPCR	**0.9567 (0.0084)**	0.8805 (0.0024)	0.5855 (0.0039)	0.7745 (0.0027)	0.8800 (0.0025)	0.8524 (0.0072)	0.8067 (0.0067)
NR	**0.9533 (0.0127)**	0.8452 (0.0215)	0.5294 (0.0200)	0.6992 (0.0244)	0.8615 (0.0244)	0.8390 (0.0261)	0.8124 (0.0228)

Minimization algorithm which incorporates multiple Graph Laplacians over the drugs and targets into the framework for an improved interaction prediction. The algorithm is generic and can be used for prediction in protein-protein interaction [91], RNA-RNA interaction [92], etc.

The evaluation was performed using three different cross-validation settings, namely CVS1 (random drug-target pairs left out), CVS2 (entire drug profile left out) and CVS3 (entire target profile left out) to compare our method with 5 other state-of-the-art methods (three specifically designed for DTI prediction). In almost all of the test cases, our algorithm shows the best performance, outperforming the baselines.

Table 3.6: AUPR results for interaction prediction under validation setting CVS3.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.9041 (0.0125)**	0.8087 (0.0156)	0.0124 (0.0005)	0.0691 (0.0009)	0.8070 (0.0185)	0.5465 (0.0144)	0.7808 (0.0131)
IC	**0.9029 (0.0024)**	0.8079 (0.0096)	0.0421 (0.0043)	0.2256 (0.0038)	0.8128 (0.0069)	0.7437 (0.0088)	0.7786 (0.0108)
GPCR	**0.7228 (0.0323)**	0.5963 (0.0336)	0.0549 (0.0105)	0.1061 (0.0027)	0.6093 (0.0314)	0.5397 (0.0193)	0.5989 (0.0323)
NR	**0.5418 (0.0309)**	0.4356 (0.0177)	0.0850 (0.0227)	0.2669 (0.0288)	0.4643 (0.0183)	0.4907 (0.0326)	0.4774 (0.0173)

Table 3.7: AUC results for interaction prediction under validation setting CVS3.

AUPR	MGRNNM	standard	MC	MCG	WGRMF	RLS_WNN	CMF
E	**0.9683 (0.0043)**	0.9246 (0.0091)	0.5234 (0.0057)	0.8065 (0.0012)	0.9338 (0.0071)	0.9067 (0.0105)	0.9272 (0.0050)
IC	**0.9541 (0.0019)**	0.9346 (0.0041)	0.4724 (0.0065)	0.7871 (0.0069)	0.9460 (0.0034)	0.9286 (0.0046)	0.9368 (0.0032)
GPCR	**0.8975 (0.0093)**	0.8798 (0.0134)	0.5683 (0.0310)	0.6289 (0.0151)	0.8892 (0.0110)	0.8694 (0.0146)	0.8966 (0.0073)
NR	0.7502 (0.0285)	0.7263 (0.0211)	0.3767 (0.0204)	0.6522 (0.0297)	0.7967 (0.0132)	0.8124 (0.0202)	**0.8373 (0.0083)**

Chapter 4

Drug-disease association prediction using graph-regularized one bit matrix completion

The importance of Drug repositioning has been discussed in the previous section. Just like drug-target interaction prediction, Drug-disease association prediction is another approach to predict the best disease indication for a drug given the open-source biological datasets. Owing to the fact that similar drugs tend to have common pathways and disease indications, the association matrix is assumed to be of low-rank structure. Hence, the problem of drug-disease association prediction can also been modelled as a low-rank matrix-completion problem.

In this work, we propose a novel matrix completion framework which makes use of the side-information associated with drugs/diseases for the prediction of drug-disease indications modelled as neighborhood graph: Graph regularized

1-bit matrix completion (GR1BMC). The algorithm is specially designed for binary data and uses parallel proximal algorithm to solve the aforesaid minimization problem taking into account all the constraints including the neighborhood graph incorporation and restricting predicted scores within the specified range. The results of the proposed algorithm have been validated on two standard drug-disease association databases (Fdataset and Cdataset) by evaluating the AUC across the 10-fold cross validation splits. The usage of the method is also evaluated through a case study where top 5 indications are predicted for novel drugs and diseases, which then are verified with the CTD database. The results of these experiments demonstrate the practical usage and superiority of the proposed approach over the benchmark methods.

4.1 Introduction

There have been some successfully re-positioned drugs through manual and rational investigations but this is not an efficient and scalable way given the huge space of drug interactions. Therefore, computational approaches have been used over the past years to systematically predict the indications, pruning down the massive search space for researchers and saving huge amounts of efforts, time and cost. This explains the immense importance of predicting new associations between drugs and diseases using statistical and machine learning based methods .

Early attempts to predict novel indications were based on gene expression

profiles [93, 94]. [93] proposed a database having ranked drug response gene expression which were queried with a gene signature specific to a disease. The drug response profiles which either correlate or anti-correlate were identified. This approach lacks validation on a large scale dataset and may not be precise enough owing to different conditions under which expression profiles are generated.

Drug-disease association prediction can also be modelled intuitively as a collaborative filtering problem. The objective of this class of approaches is to recover a complete matrix from its sampled entries by exploiting its low-rank structure. The low-rank assumption stems from the idea that similar drugs affect biological systems in a similar way and have common indications [95].

In this work, we formulate drug disease association prediction as a one-bit matrix completion problem. Furthermore, we introduce graph regularization to exploit the similarities between drugs and diseases. The objective function is minimized using parallel proximal algorithm (PPXA) [96]. PPXA is an iterative proximal splitting algorithm that paralelly solves for each of the non-necessarily smooth terms in the objective function, while benefiting from sounded convergence guarantees. The novelty of our approach lies in

- Modelling the drug-disease association prediction as graph-regularized matrix completion problem.
- Restricting the association scores in range [0,1] for obtaining meaningful biological scores.

- Solving the optimization problem using PPXA which has guaranteed convergence properties [96].

A schematic overview of GR1BMC is shown in Figure 4.1.

4.2 Dataset

We have used two gold standard databases to validate our approach. The first one, called *F dataset*, proposed by [97] has 313 diseases, 593 drugs and 1933 drug-disease associations from various sources. The second dataset, called *Cdataset* is a larger one with 663 drugs, 409 diseases and 2532 associations [98].

For the datasets the drug information is obtained from DrugBank [99], an exhaustive database containing comprehensive information about drugs and targets. The disease information was assembled from human phenotypes listed in public database, OMIM (Online Mendelian Inheritance in Man) database [100], which has information on human genes and diseases.

The similarity information of drugs, calculated as Tanimato score [101], is extracted using Chemical Development Kit (CDK) [102] based on the chemical structures of drugs in SMILES (Simplified Molecular-Input Line-Entry System) format, obtained from DrugBank. MimMiner [103] provides the similarities between diseases using the medical descriptors of diseases from OMIM database by measuring the number of MeSH (medical subject headings vocabu-

lary) terms. Both kinds of similarites are in range [0,1].

The information on number of drugs, diseases and the associations between them has been summarized in Table 4.1.

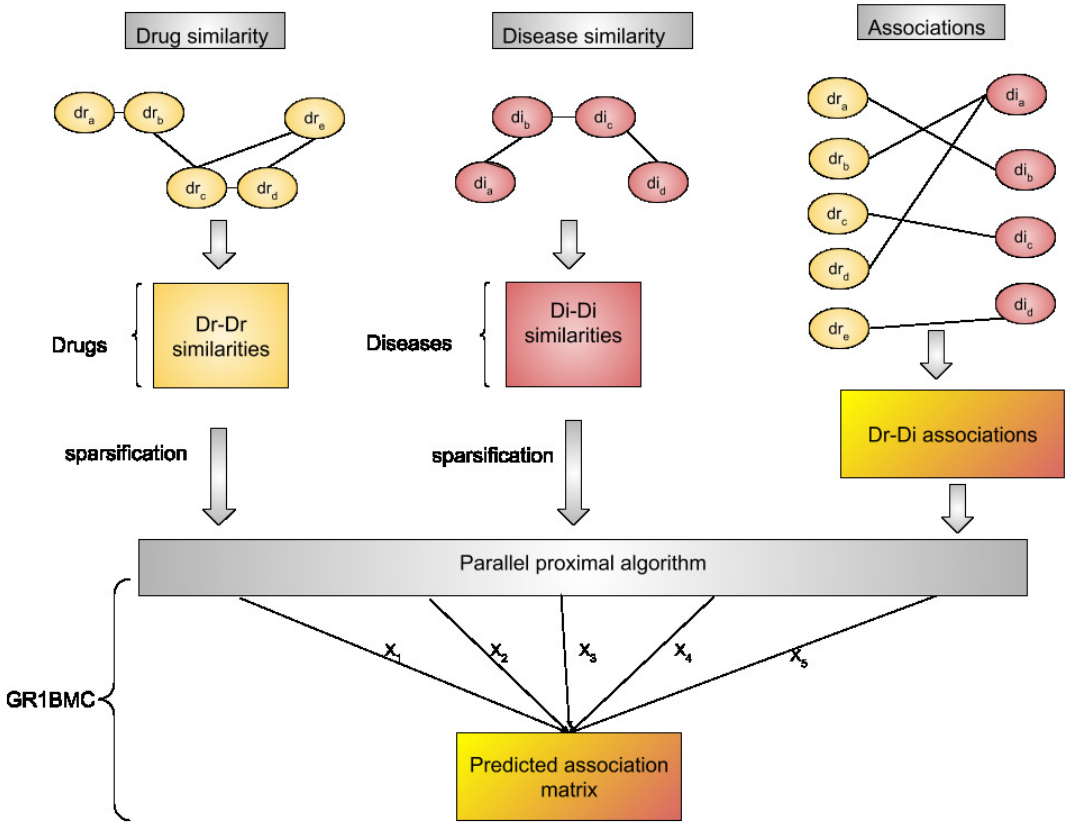


Figure 4.1: A schematic overview of GR1BMC for predicting drug-disease assocations

Table 4.1: A summary of the number of associations, drugs and diseases in each dataset used.

Datasets	# Associations	# Drugs	# Diseases
# Fdataset	1933	593	313
# Cdataset	2532	663	409

4.3 Methodology

4.3.1 Data preprocessing

We process the drug and disease similarity and graph laplacian matrices as per the two steps mentioned in subsection 3.3.1 of Chapter 3 to ensure better learning.

4.3.2 Proposed framework

To predict the drug-disease association matrix X (with drugs on rows and diseases on columns), we model it as a low-rank matrix and aim to complete its available version Y. Since low-rank approximation is an NP-hard problem, we solve its closest convex surrogate i.e. nuclear norm minimization. Nuclear norm is defined as the sum of absolute singular values of a matrix.

To incorporate the disease and drug similarities into this imputation framework, we introduce the Laplacian graph regularization terms,

$$\min_{X} ||Y - M \circ X||_F^2 + \lambda||X||_* + \mu_1 Tr(X^T L_{di} X) + \mu_2 Tr(X L_{dr} X^T)$$
$$\text{s.t. } X \in [0, 1] \tag{4.1}$$

Here, $||.||_*$ denotes the nuclear norm, Tr denotes the trace. M is the binary masking operator which is element-wise multiplied to complete data matrix X (having 1's at train indices and 0's at test indices) using hadamard product operator ($\circ$). L_{di} and L_{dr} denote the disease and drug Laplacian matrices.

Here, we propose to make use of the parallel proximal algorithm (PPXA) from [96] (see also [104] for its application in the context of biochemistry). In this algorithm, we solve for X, by taking a proxy variable for each of the terms in (4.1) [83] and an extra proxy variable X_3 to ensure that the predicted scores are in range [0,1]. For each iteration k, we need to compute the following proximity operators,

$$\widehat{X}_1^{(k)} = \arg\min_X \frac{\theta}{2}||Y - M \circ X||_F^2 + \frac{1}{2}||X_1^{(k-1)} - X||_F^2 \tag{4.2}$$

$$\widehat{X}_2^{(k)} = \arg\min_X \lambda\theta||X||_* + \frac{1}{2}||X_2^{(k-1)} - X||_F^2 \tag{4.3}$$

$$\widehat{X}_3^{(k)} = min(max(X_3^{(k-1)}, 0), 1) \tag{4.4}$$

$$\widehat{X}_4^{(k)} = \arg\min_X \theta\mu_1 Tr(X^T L_{di} X) + \frac{1}{2}||X_4^{(k-1)} - X||_F^2 \tag{4.5}$$

$$\widehat{X}_5^{(k)} = \arg\min_X \theta\mu_2 Tr(X L_{dr} X^T) + \frac{1}{2}||X_5^{(k-1)} - X||_F^2 \tag{4.6}$$

Hereabove, θ corresponds to the number of terms treated in parallel, that is $\theta = 5$. Below we provide the solution of each of the above sub-problems:

- Solving for $\widehat{X}_1^{(k)}$ involves taking the gradient of (4.2) and equating to 0,

$$\theta(-M^T)(Y - M\widehat{X}_1^{(k)}) + (\widehat{X}_1^{(k)} - X_1^{(k-1)}) = 0$$

$$\theta M^T M\widehat{X}_1^{(k)} - \theta M^T Y + \widehat{X}_1^{(k)} - X_1^{(k-1)} = 0$$

$$(\theta M^T M + I)\widehat{X}_1^{(k)} = X_1^{(k-1)} + \theta M^T Y$$

where I is the identity matrix. The above can now be easily solved by finding least squares solution.

- $\widehat{X}_2^{(k)}$ can be obtained by soft-thresholding the singular values of $X_2^{(k-1)}$ and multiplying the thresholded singular value matrix by the left and right singular vector matrices of $X_2^{(k-1)}$ i.e.,

$$X_2^{(k-1)} = US^{(k-1)}V^T$$
$$\widehat{S}^{(k-1)} = \text{soft}(S^{(k-1)}, \lambda\theta/2)$$
$$\widehat{X}_2^{(k)} = U\widehat{S}^{(k-1)}V^T$$

  where soft$(S^{(k-1)}, \lambda\theta/2)$ =sign$(S^{(k-1)})$max$(0, |S^{(k-1)}| - \lambda\theta/2)$. Here $S^{(k-1)}$ denotes the singular value matrix, U and V are the left and right singular matrices of $X_2^{(k-1)}$, obtained after SVD-decomposition.

- Solving for $\widehat{X}_3^{(k)}$ is done by applying max-thresholding followed by min-thresholding on $X_3^{(k-1)}$.

- To solve for $\widehat{X}_4^{(k)}$, we employ the same strategy as for $\widehat{X}_1^{(k)}$ and equate the

gradient of (4.5) to 0,

$$\theta\mu_1(L_{di}\widehat{X}_4^{(k)} + L_{di}^T\widehat{X}_4^{(k)}) + (\widehat{X}_4^{(k)} - X_4^{(k-1)}) = 0$$
$$2\theta\mu_1 L_{di}\widehat{X}_4^{(k)} + \widehat{X}_4^{(k)} = X_4^{(k-1)}$$
$$\widehat{X}_4^{(k)} = (2\theta\mu_1 L_{di} + I)^{\dagger} X_4^{(k-1)}$$

- SSimilarly, update step for $\widehat{X}_5^{(k)}$ can be obtained as follows,

$$\widehat{X}_5^{(k)} = X_5^{(k-1)}(2\theta\mu_2 L_{dr} + I)^{\dagger}$$

In the above two update steps, $A^{\dagger}$ denotes the Moore-Penrose pseudo inverse of A. The next iterate $X^{(k)}$ is finally obtained by averaging over the five proximal values, as follows,

$$\widehat{X}^{(k)} = \frac{1}{\theta}(\widehat{X}_1^{(k)} + \widehat{X}_2^{(k)} + \widehat{X}_3^{(k)} + \widehat{X}_4^{(k)} + \widehat{X}_5^{(k)}) \tag{4.7}$$

with $\theta = 5$. Furthermore, each of the proxy variables is updated via the following update rule,

$$X_i^{(k)} = X_i^{(k-1)} + 2\widehat{X}^{(k)} - \widehat{X}^{(k-1)} - \widehat{X}_i^{(k)} \tag{4.8}$$

The complete algorithm is given in Algorithm 5. [1]

We display in Figures 4.2 and 4.3 example of convergence plots (i.e. evo-

[1]The code of GR1BMC is available at https://github.com/aanchalMongia/GROBMC-PPXA-DDA

Algorithm 5 GR1BMC (Y, M, S_d, S_t)

Initialize: p, μ_1, μ_2, λ

$X_1^{(0)}, X_2^{(0)}, X_3^{(0)}, X_4^{(0)}, X_5^{(0)}$

Sparsify: Compute N_{di}^{ij}, N_{dr}^{ij}, $\widehat{S}_{di} = N_{di}^{ij} \odot S_{di}, \widehat{S}_{dr} = N_{dr}^{ij} \odot S_{dr}$

$D_{di} = \sum_j (\widehat{S}_{di})^{ij}, L_{di} = (D_{di})^{-1/2}(D_{di} - \widehat{S}_{di})(D_{di})^{-1/2}$

$D_{dr} = \sum_j (\widehat{S}_{dr})^{ij}, L_{dr} = (D_{dr})^{-1/2}(D_{dr} - \widehat{S}_{dr})(D_{dr})^{-1/2}$

For loop 1, iterate (k)

$\widehat{X}_1^{(k)} = (5M^TM + I)^{-1}(X_1^{(k-1)} + 5M^TY)$

$X_2^{(k-1)} = US^{(k-1)}V^T$

$\widehat{S}^{(k-1)} = \text{sign}(S^{(k-1)}) \max(0, |S_k| - 5\lambda/2)$

$\widehat{X}_2^{(k)} = U\widehat{S}^{(k-1)}V^T$

$\widehat{X}_3^{(k)} = \min(\max(X_3^{(k-1)}, 0), 1)$

$\widehat{X}_4^{(k)} = (10\mu_1 L_{di} + I)^{\dagger} X_4^{(k-1)}$

$\widehat{X}_5^{(k)} = X_5^{(k-1)}(10\mu_2 L_{dr} + I)^{\dagger}$

$\widehat{X}^{(k)} = \frac{1}{5}(\widehat{X}_1^{(k)} + \widehat{X}_2^{(k)} + \widehat{X}_3^{(k)} + \widehat{X}_4^{(k)} + \widehat{X}_5^{(k)})$

$X_i^{(k)} = X_i^{(k-1)} + 2\widehat{X}^{(k)} - \widehat{X}^{(k-1)} - \widehat{X}_i^{(k)}, i = 1, 2...5$

End loop 1

Return: $\widehat{X}^{(k)}$

lution of objective function along iterations) for Fdataset and Cdataset, respectively.

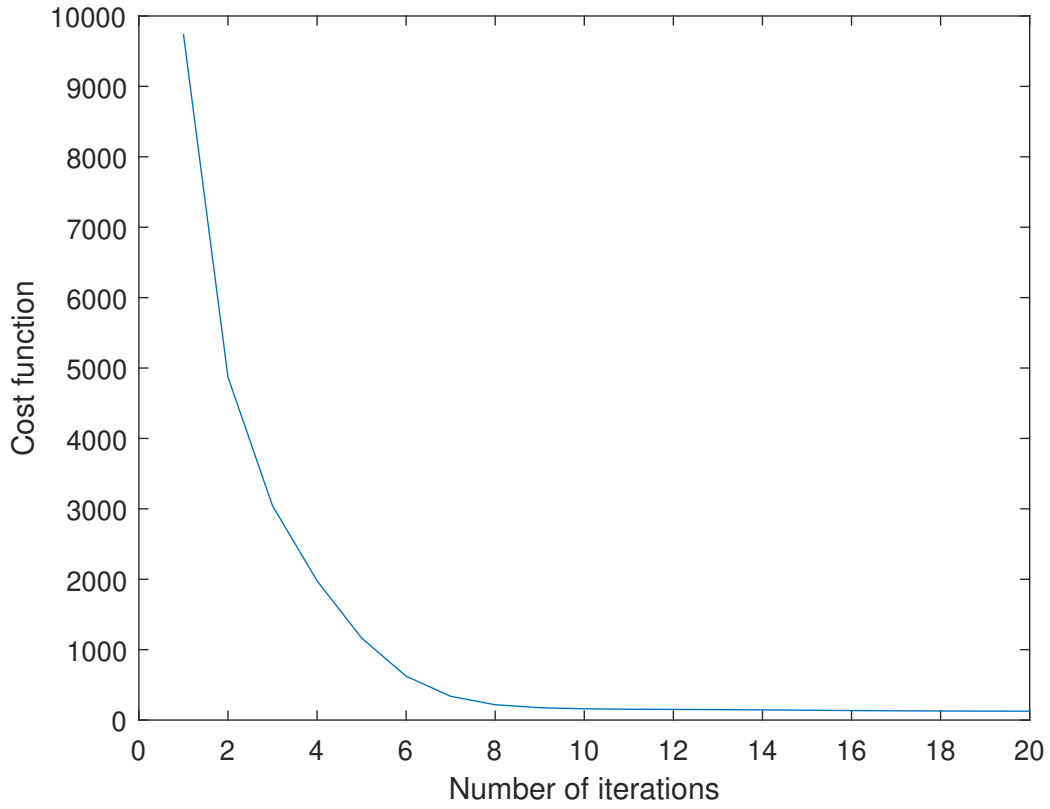


Figure 4.2: Convergence plot for GR1BMC on Fdataset

4.4 Results

4.4.1 Evaluation criteria

To experimentally evaluate the prediction performance of GR1BMC, we use κ-fold cross validation strategy ($\kappa = 10$) as used in subsection 3.4.1 of Chapter 3, called CVS1 (Cross validation setting 1) where we divide all the known associations into κ equal subsets and 1 of them is treated as a testing set, while the remaining ones constitute the training set. The associations in training set are

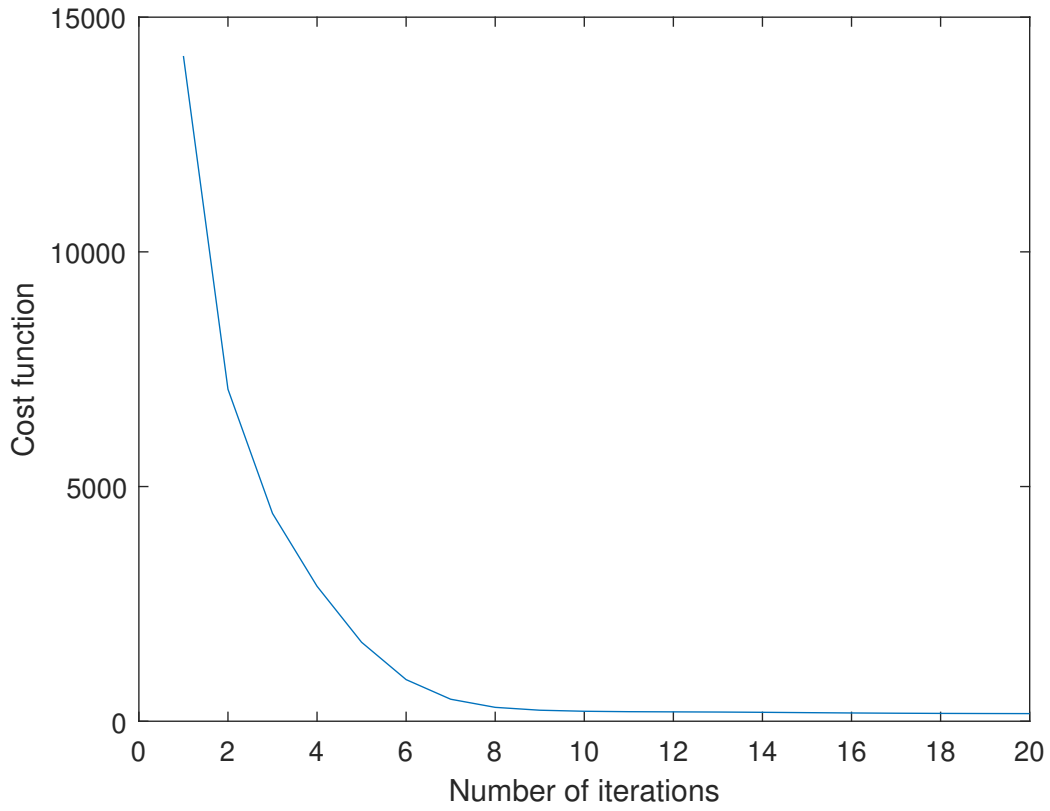


Figure 4.3: Converegence plot for GR1BMC on C dataset

given as input to the algorithm which then returns the fully imputed association matrix.

After this matrix completion, the predictions on testing set and other candidate associations for all drugs are ranked in descending order of scores and TPR (True Positive Rate)/Recall, FPR (False Positive Rate) and PPV (Positive predicted value)/Precision is calculated for every rank threshold. These values at every threshold are used to plot an ROC (Receiver Operating Characteristic) curve with FPR on x-axis and TPR on y-axis. In a similar way, a Precision-Recall curve is obtained by plotting Recall/TPR on x-axis and Precision on y-axis. The area under both these curves called Area under the ROC curve (AUC)

and Area under the precision-recall curve (AUPR) are used to asses the performance of the methods used to predict drug-disease associations, similar to the way we did in subsection **??** of chapter 3 in Drug-target interaction prediction.

Figures 4.4 and 4.5 show the ROC curves obtained on all the 10 folds after running GR1BMC on both the datasets. The average AUC and AUPR across all the folds has been shown in Tables 4.2 and 4.3. As can be observed from the table, GR1BMC performs better than the benchmarks techniques on both the datasets, especially in terms of AUPR. It should be noted that AUPR is a relatively more important metric in this problem since it heavily punishes highly ranked non-associations (false positives), which is crucial given the nature of application as false positive indications would lead to wastage of resources if the proposed indications are tested in clinical experiments.

4.4.2 Comparison with benchmark techniques

To evaluate the performance of GR1BMC, we compare the results of cross-validation experiments with those of the latest methods proposed for drug-disease association prediction: Bounded nuclear norm regularization (BNNR) [105], Heterogeneous Network for drug-Disease association prediction (HNRD) [106] and drug repositioning recommendation system (DRRS) [107]. BNNR and

Table 4.2: Average AUC across 10-fold cross-validation for various techniques while predicting drug-disease associations.

Datasets	GR1BMC	BNNR	HNRD	DRRS
Fdataset	**0.9773**	0.9330	0.9420	0.9300
Cdataset	**0.9807**	0.9480	0.9500	0.9470

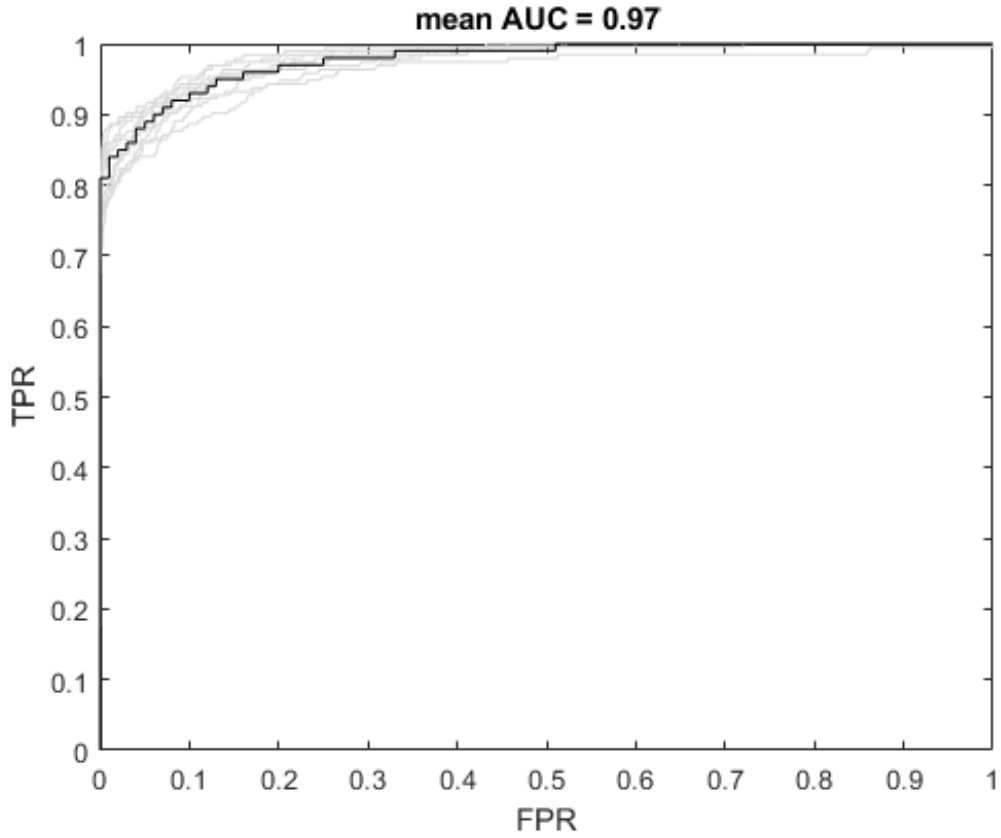


Figure 4.4: ROC curves obtained for all the 10 folds after applying GR1BMC on Fdataset

DRRS are the closest in terms of formulation used to model the problem. Both the methods deploy nuclear norm minimization on a heterogeneous network matrix obtained by integrating drug similarity, disease similarity, association matrix and its transpose; BNNR additionally handles the noise originating from similarities which violate the low-rankness and restrict the predicted values to be in range [0,1]. But, the low-rank property of the heterogeneous matrix is unexplained in both the works; which is a crucial assumption behind nuclear

Table 4.3: Average AUPR across 10-fold cross-validation for various techniques while predicting drug-disease associations.

Datasets	GR1BMC	BNNR	HNRD	DRRS
Fdataset	**0.7247**	0.4410	0.5720	0.3780
Cdataset	**0.7537**	0.4710	0.6700	0.4020

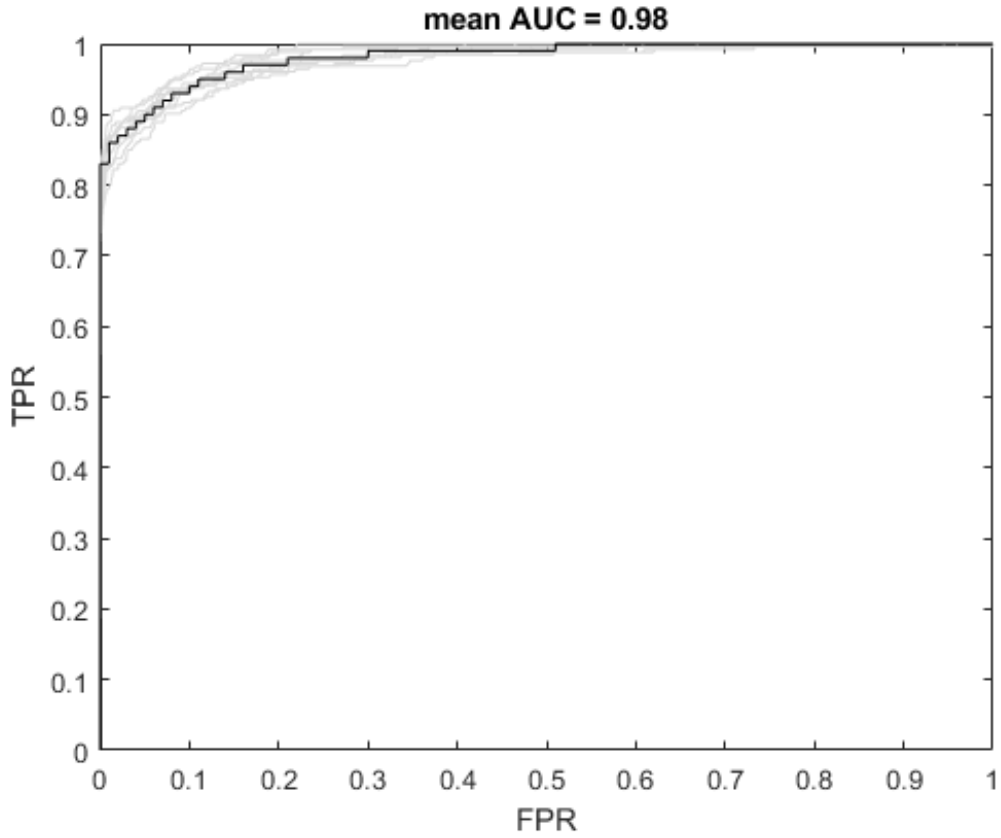


Figure 4.5: ROC curves obtained for all the 10 folds after applying GR1BMC on Cdataset

norm minimization. This heterogeneous matrix comprises of associations between drugs and diseases as well as drug-drug and disease-disease similarities. The authors clearly explain validity of the low-rank assumption in association matrix but not for the heterogeneous matrix.

The results of 10-fold cross-validation have been shown in Tables 4.2 and 4.3. It can be seen that our proposed approach shows competitive performance in terms of area under the ROC curve and is better than the benchmark techniques in terms of precision and recall also.

DRUG INFORMATION		DISEASE INFORMATION		
Drug name	DrugBank ID	Disease name	OMIM ID	Confirmation
Levodopa	DB01235	(PARKINSON DISEASE, LATE-ONSET; PD)	168600	CTD confirmed
		(DEMENTIA/PARKINSONISM WITH NON-ALZHEIMER AMYLOID PLAQUES)	D125320	CTD confirmed
		(DYSTONIA 9; DYT9	D601042	CTD confirmed
		(DEMENTIA, LEWY BODY; DLB)	D127750	
		(RENAL FAILURE, PROGRESSIVE, WITH HYPERTENSION; RFH1)	D161900	
Doxorubicin	DB00997	(COLORECTAL CANCER; CRC)	D114500	CTD confirmed
		(DOHLE BODIES AND LEUKEMIA)	D223350	
		(RETICULUM CELL SARCOMA)	D267730	CTD confirmed
		(RENAL CELL CARCINOMA, NONPAPILLARY; RCC)	D144700	CTD confirmed
		(LEUKEMIA, CHRONIC LYMPHOCYTIC, SUSCEPTIBILITY TO, 2)	D109543	CTD confirmed
Amantadine	DB00915	(PARKINSON DISEASE, LATE-ONSET; PD)	D168600	CTD confirmed
		(DEMENTIA/PARKINSONISM WITH NON-ALZHEIMER AMYLOID PLAQUES)	D125320	CTD confirmed
		(ALZHEIMER DISEASE, FAMILIAL EARLY-ONSET, WITH COEXISTING AMYLOID AND PRION PATHOLOGY)	D605055	CTD confirmed
		(DEMENTIA, LEWY BODY; DLB)	D127750	CTD confirmed
		(ALZHEIMER DISEASE; AD)	D104300	CTD confirmed
Flecainide	DB01195	(ATRIAL FIBRILLATION, FAMILIAL, 1; ATFB1)	D608583	CTD confirmed
		(HYPERTENSION, DIASTOLIC, RESISTANCE TO)	D608622	CTD confirmed
		(RENAL FAILURE, PROGRESSIVE, WITH HYPERTENSION; RFH1)	D161900	
		(INSENSITIVITY TO PAIN WITH HYPERPLASTIC MYELINOPATHY)	D147530	
		(STROKE, ISCHEMIC)	D601367	
Metformin	DB00331	(DIABETES MELLITUS, INSULIN-DEPENDENT, 2)	D125852	CTD confirmed
		(COLORECTAL CANCER; CRC)	D114500	CTD confirmed
		(HYPERLIPOPROTEINEMIA, TYPE V)	D144650	CTD confirmed
		(ENDOMETRIOSIS, SUSCEPTIBILITY TO, 1)	D131200	
		(UTERINE ANOMALIES)	D192000	

Table 4.4: Top 5 predicted diseases for Levodopa, Doxorubicin, Amantadine, Flecainide and Metformin with their evidence in CTD database

4.4.3 Parameter settings

The matrices $X_1^{(0)}, X_2^{(0)}, X_3^{(0)}, X_4^{(0)}$ and $X_5^{(0)}$ are initialized randomly and the algorithm is run for a fixed number of iterations k (k=20 here) that appears sufficient to reach practical stabilization of the objective function. The running time is in the order of seconds; PPXA takes approximately 4 and 6 seconds on Fdataset and Cdataset respectively on a single core machine with a clock speed of 2.8 GHz, 64 GB RAM (Intel(R) Xeon(R) CPU E5-1603 v3 processor) . To look for a feasible solution in the space of low-rank association matrices, we need to determine the values of the hyperparameters λ, μ_1 and μ_2. This is done to weigh the importance of nuclear norm term and the trace terms in our objective function for each of the two datasets. The values of μ_1 and μ_2 determine the weights given to each of the drug and disease laplacians, hence exhibiting the importance of neighborhood information of drugs and targets in our framework for a dataset. The optimal values of these parameters are

found by performing cross validation on the training set and taking the value of parameters from the set $\{0.01, 0.05, 0.1, 0.5, 1, 5, 10\}$. The values of λ, μ_1 and μ_2 are robust across the two datasets and are found to be $0.1, 0.05, 0.1$ for both the datasets.

,

4.4.4 Case study to predict novel associations

To asses the practical usage of the proposed algorithm, we perform a case study where we chose 5 candidate drugs to look for their novel indications (dummy drug re-positioning) after prediction the associations using our proposed approach.

We train our model on the known associations on Fdataset. After the matrix completion is done, we rank the remaining candidate diseases for each drug in descending order or predicted association scores.

These rankings or predictions of novel indications for drugs is verified by validating the top-5 indications for any 5 drugs with the public database comparative toxicogenomics database (CTD) [108]. We show the validation on the following 5 drugs: Levodopa, Doxorubicin, Amantadine, Flecainide and Metformin.

The indications predicted by GR1BMC and the evidence from CTD is shown in table 4.4. It can be seen that at least 3 indications are confirmed with the CTD databse for 4 out of 5 drugs and a total of 17 out of 25 predicted associations

have evidence in CTD database. Also, the indications which are not verified could be the potential candidates for drug-repositioning and could be explored by medical researchers.

4.5 Conclusion

The huge amount of time and efforts taken for the development drugs calls for the need for efficient and reliable computational methods to assist drug repositioning. In this thesis, we present a novel approach to predict drug-disease indications based on parallel proximal algorithm, which benefits from guaranteed convergence and great numerical performance. Cross validation and experiments on gold standard dataset demonstrate the superiority of the proposed approach over the benchmark techniques. The practical usage is also validated by the case study where novel indications for existing drugs are found and majority are validated with the CTD database. The proposed method is generic and can be applied to other association/interaction prediction problems such as protein-protein interaction prediction, human microbe-disease association (MDA) prediction, gene-disease association prediction, etc.

Chapter 5

Drug-virus association database: anti-viral drug prediction using matrix completion

COVID-19 has fast-paced drug re-positioning for its treatment. This work builds computational models based on matrix completion variants for the same. The aim is to assist clinicians with a tool for selecting prospective antiviral treatments. Since the virus is mutating fast [109], the tool is likely to help clinicians in selecting the right set of antivirals for the mutated isolate.

The most crucial contribution of this work is a manually curated database publicly shared, comprising of existing associations between viruses and their corresponding antivirals. The database gathers similarity information using the chemical structure of drugs and the genomic structure of viruses. Along with this database, we make available a set of state-of-the-art computational drug re-positioning tools based on matrix completion. The tools are first analysed on a standard set of experimental protocols for drug target interactions. The best performing ones are applied for the task of re-positioning antivirals for COVID-

19. These tools select six drugs out of which four are currently under various stages of trial, namely Remdesivir (as a cure), Ribavarin (in combination with others for cure), Umifenovir (as a prophylactic and cure) and Sofosbuvir (as a cure). Another unanimous prediction is Tenofovir alafenamide, which is a novel tenofovir prodrug developed in order to improve renal safety when compared to the counterpart tenofovir disoproxil. Both are under trail, the former as a cure and the latter as a prophylactic. These results establish that the computational methods are in sync with the state-of-practice. We also demonstrate how the selected drugs change as the SARS-Cov-2 mutates over time, suggesting the importance of such a tool in drug prediction.

5.1 Introduction

There has been an exponential rise in the total active cases and deaths due to COVID-19 (COrona VIrus Disease-2019) since the first case in Wuhan, China in December, 2019 [110]. The disease results in severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), which is known to be highly transmittable and has spread across more than 100 countries. This pandemic has wreaked havoc on people's social life, the global economy, and most importantly the health of the human race. The death numbers are frightening, confirming about 467K deaths worldwide till mid-June, 2020 [110].

As medical professionals are striving to save lives, research scientists specialized in drug development, are racing against time to develop a vaccine against

SARS-CoV-2 [111]. The investigation involved for developing a vaccine (or even a new drug) is time consuming, requiring several phases of extensive trials. Experts believe that it is highly unlikely that a vaccine will be ready before a year or more. In such circumstances a better way may be to re-position existing drugs for treating COVID-19. This is a well known approach where existing drugs (which have already been approved for release in the market) are investigated for new disease/s [62]. Drug re-positioning is usually cost effective and fast (compared to developing a new drug / vaccine) since its effects are well studied. One classic example for drug re-positioning is Chlorocyclizine , which was initially developed as an anti-allergic but later found to act against the hepatitis C virus [112]. Another example is Imatinib mesylate (sold under the trade name Gleevec), it was originally used as a treatment for leukemia but later was found to be effective against genes associated with gastrointestinal-stromal tumors [64, 65].

Given the relatively large drug-virus association space, manual investigation in wet-labs is not a scalable strategy. Putting all the anti-virals in trials for treating corona is not very feasible either; especially because time is of essence. In such a scenario, computational approaches can help; they can be used to prune down the search space for the drugs to be investigated [55]. Practically, such approaches could also assist the clinicians to come up with treatments for rapidly mutating viruses by pruning the anti-viral drug space. Specifically, a computational approach which takes into account the genomic structure of the latest viral isolate or its similarity with the previously occuring strains of viruses would be

helpful in deciding the treatment. With this objective, we have manually curated a comprehensive database called DVA (Drug Virus Association), having the approved (anti-viral) drug-virus associations in the literature along with the similarity measures associated with drugs (chemical structure similarity) and viruses (genome sequence similarity). To the best of our knowledge there is no existing database for drug virus association.

The DVA database we propose in this work lies the foundation for further computational studies on this topic. There can be various methodologies to predict drug virus association. The prediction problem can be approached via feature-based classification models, neighborhood models, matrix completion models, network diffusion models etc. A recent empirical study on well established drug-target interaction databases exhibit the best prediction performance by matrix completion models [55]. In computer science, matrix completion is used routinely in recommendation systems. The general problem of drug-disease association can actually be thought of as a recommendation system, where drugs are being recommended for treating a disease. Given the success of matrix completion techniques in drug target interaction, we deploy state-of-the-art matrix completion techniques on our curated DVA database. We perform a thorough comparative analysis of those for predicting assessed drug-disease associations. Then, we apply the methodology for pruning the search space of potential candidates for COVID-19 trial drugs. Finally, we show how the tool helps in selecting drugs as the virus mutates.

5.2 Dataset

The proposed DVA dataset aims at being exhaustive. It compiles various existing sources, housing together all the anti-viral drugs proved clinically to be effective against viruses infecting humans. The dataset has 121 drugs and 38 viruses. We believe such resource would be highly useful for analysing and proposing anti-virals not only for the novel coronaviruses but other viruses too. Along with that, it may also be used to computationally identify viruses that a newly discovered drug may target. The associated metadata (information about the drugs and viruses) may also help clinicians in manual analysis and having a deeper insight.

5.2.1 Drug-virus association compilation

All the associations corresponding to anti-viral drugs clinically shown to act against human host viruses have been assembled from standard DrugBank database [99] (https://www.drugbank.ca/categories/DBCAT000066). To ensure that the database is fully comprehensive, other literature works [113, 114, 115, 116, 117, 118, 119, 120, 121] and resources such as ViPR [122] were also scanned for any additional drug-viral associations. ViPR or NIAID Virus Pathogen Database and Analysis Resource (http://www.viprbrc.org/) is a repository of data and analysis tools for virology research [122] capturing various types of information derived from comparative genomics analysis and visualization tools. It has antiviral drug information (for 21 viral species) derived imported from Drug-

Bank (https://www.drugbank.ca/) [99].

The DrugBank Identifier (DrugBank ID) of the anti-viral drugs involved is considered as the unique key for the drugs, obtained from DrugBank vocabulary (https://www.drugbank.ca/releases/latestopen-data). Along with the viral association information, we also store the target pathway and mechanism of action of each drug for quick reference in any further investigation. Apart from this, each drug is mapped to its corresponding KEGG Identifier (KEGG ID) from the KEGG Compound/KEGG Drug database (https://www.genome.jp/kegg/drug/, https://www.genome.jp/kegg/ compound/) of the KEGG (Kyoto encyclopedia of genes and genomes) [78]. The KEGG IDs were taken from the linking file provided at https://www.drugbank.ca/releases/latestexternal-links [99] or manually added in the case of drugs missing in the linking file.

Each virus is identified by an acronym assigned to it (in case of no acronym, full virus name is used). The viral family, genome type, transmission route and incubation period is also available in the virus metadata file along with the accession number of the complete genomic sequence of the viruses fetched from NCBI (National Center for Biotechnology Information) Viral genome browser https://www.ncbi.nlm.nih.gov/genomes/GenomesGroup.cgi [123]).

5.2.2 Similarity computation

To integrate the similarity information to the drug-virus associations, we have computed similarities between the drugs based on their chemical structures and

between the viruses using their complete genomic sequences.

- DRUG SIMILARITY: All the DrugBank IDs were mapped to KEGG IDs of the corresponding drug/compound in the KEGG database [78]. The chemical structure similarity was measured between the drugs by computing the SIMCOMP score [80] based on the maximum common substructures between the chemical structure of the compounds using the KEGG API page at GenomeNet (https://www.genome.jp/tools /gn$_t ools_a pi.html$). The drugs for which the SIMCOMP score was less than the set cutoff (0.001) and the drugs with no KEGG IDs available were assigned a similarity score of 1 to themselves and 0 to other drugs in the dataset.

- VIRUS SIMILARITY: The $d2^*$ distance based on ONF (Oligonucleotide frequency) measure between the DNA sequences was shown to be the best amongst various other ONF metrics with several k-mers length in host prediction accuracy at the genus level [124]. Hence, we compute d2* dissimilarity/distance (at k=6) between the viral genome sequences obtained from NCBI [123]. The reference sequences of viruses were saved in FASTA format to be used by the distance computation software (https://github.com/jessieren/VirHo proposed by [124]. The $d2^*$ distance was subtracted from 1 to obtain the similarity measure.

  For the viruses with segmented structure (Influenza A virus, Influenza B virus, Influenza C virus, Lassa mammarenavirus), the coding sequence in

the nucleotide sequence of each genomic segment (taken in decreasing order of length was taken) was combined to form the complete viral sequence.

5.3 Methodology

5.3.1 Data preprocessing

The drug-virus indications have been stored and processed in a matrix form of size $m \times n$ (m being no of drugs in the database and n being the number of viruses) to be used as input for any of the 6 matrix completion algorithms we made available in our repository.

The similarity information has been represented as symmetric matrices of size $m \times m$ (drug similarity matrix) and $n \times n$ (virus similarity matrix). These matrices and the correspodning laplacian matrices were processed as per the two steps mentioned in subsection 3.3.1 of Chapter 3 to ensure better learning.

5.3.2 Proposed framework

The proposed computational approach is to use Matrix compeletion and its variants for drug-virus association prediction. In this subsection, we describe each of the matrix completion algorithms used (**www.github.com/AanchalMongia/DVA**), along with their mathematical formulations and resolution strategies.

Let $X_{m \times n}$ be the complete drug-virus association matrix (with m drugs on

rows and n viruses on columns) with binary entries (1 denoting that the drug is known to act against the virus and 0 denoting no association). Here X is the matrix to be recovered from its sampled (partially known) entries in Y. Let M denote the masking operator (elementwise multiplied to X) having 1's at positions where associations are known and 0 otherwise. Then, the matrix completion problem can be formulated as searching for X (as shown in equation (1.1)) satisfying,

$$Y = M \circ (X), \tag{5.1}$$

under specific constraints. In particular, it is typically assumed that similar drugs act in a similar manner, hence X to be recovered (from Y and M) is of low-rank.

5.3.2.1 Matrix factorization (MF)

The most straightforward technique of solving low-rank matrix completion is matrix factorization, where the data matrix $X_{m\times n}$ is decomposed into two latent factor matrices $U_{m\times k}$ and $V_{k\times n}$, where k denotes the number of latent (hidden) factors deciding if a drug is associated with a virus or not. X is recovered by solving for U and V in the following minimization problem,

$$\min_{U,V} ||Y - M \circ (UV)||_F^2. \tag{5.2}$$

The above problem is solved in an alternating manner, by first decoupling the mask using a majorization-minimization technique [5, 125] and then using al-

ternating least squares method [6] to obtain U and V. The complete algorithm is described in [126].

5.3.2.2 Deep matrix factorization (DMF)

An extension of matrix factorization has been proposed motivated by the success of deep dictionary learning [13], where the data matrix X is decomposed into multiple factor matrices (analog to multiple layers) to capture more complex hidden features in the data. The formulation of the minimization problem in the case of 2-layer matrix factorization is given below,

$$\min_{U_1,U_2,V} ||Y - M(U_1U_2V)||_F^2 \text{ s.t. } U_1 \geq 0, U_2 \geq 0. \tag{5.3}$$

The above problem is solve alternatively. The minimization with respect to variables U_1 and V, is done in a similar way to that of matrix factorization, while the update on U_2 can be obtained as shown in [18].

5.3.2.3 Graph regularized matrix factorization (GRMF)

Another variant of Matrix factorization has been proposed to incorporate metadata associated with the row and column entities (drug and virus similarities in this case) [14]. Here, the drug and virus entities form the nodes of two separate graphs and the similarity between them is assumed to be the weights between the nodes. Regularization is imposed by adding graph Laplacian penalties to

the cost function of matrix factorization as shown below,

$$\min_{U,V} ||Y - M \circ (UV)||_F^2 + \mu_1 \mathrm{tr}(\mathrm{U}^\top \mathrm{L_d U}) + \mu_2 \mathrm{tr}(\mathrm{V L_v V}^\top), \tag{5.4}$$

where $\mu_1 > 0$ and $\mu_2 > 0$ are coefficients penalizing the graph regularization Laplacian terms and tr denotes the trace of the matrix. $L_d = D_d - S_d$ and $L_v = D_v - S_v$ are the graph Laplacians [17] for S_d (row/drug similarity matrix) and S_v (column/virus similarity matrix), respectively, and $D_d^{ii} = \Sigma_j S_d^{ij}$ and $D_v^{ii} = \Sigma_j S_v^{ij}$ are the associated degree matrices. A resolution technique for the above formulation has been shown in [14].

5.3.2.4 Matrix completion (MC)

Matrix factorization based approach leads to a non-convex minimization problem and hence rarely benefits from global convergence guarantees. To limit the space of minimizers, it may be useful to impose a low-rank constraint on the solution X. Since rank minimization is still an NP-hard problem, it was proposed to relax the above constraint to its closest convex surrogate by making use of the nuclear norm penalty [7, 8]. The formulation for the resulting nuclear norm minimization problem (referred to as matrix completion by the authors) is,

$$\min_X ||X||_* \text{ s.t. } Y = M \circ (X). \tag{5.5}$$

The above problem can be solved alternatively, by invoking majorization-minimization arguments [5] to deal with the mask operator M and by applying thresholding

operations on the singular values to process the nuclear norm term [126].

5.3.2.5 Graph regularized matrix completion (GRMC)

Just like matrix factorization, nuclear norm minimization based matrix completion can also be graph regularized by incorporating graph Laplacian penalties to take metadata/similarity information into account. The formulation for the minimization problem is given by,

$$\min_{X} ||Y - M \circ (X)||_F^2 + \lambda ||X||_* + \mu_1 \mathrm{tr}(\mathrm{X}^\top \mathrm{L_d X}) + \mu_2 \mathrm{tr}(\mathrm{X L_v X}^\top). \quad (5.6)$$

The above formulation can either be solved using ADMM (alternating direction method of multipliers) [85, 127] as was done in [21] (referred as GRMC here) or by explicitly taking care of the constraint that the recovered values should be in the range $[0, 1]$. If the latter range constraint is taken into account, we obtain then a new variant called graph regularized binary matrix completion. The minimization with respect to X can be solved by making use of the PPXA (parallel proximal algorithm) [96]. Such approach allows to decouple the constraints by introducing proxy variables and then solving each subproblem in a parallel fashion as shown in [20] (referred as GRBMC here).

5.3.3 Setting of hyperparameters

The stepsize, regularization parameters and latent factor dimensions, for the above techniques have been tuned using cross-validation on training set (after

hiding 10 % of the data) in each of the three cross-validation settings (see Section 5.4.2). The parameters obtained after extensive cross-validation on the setting CV2 (randomly hiding the virus entities) have been further used in predicting drugs for SARS-Cov-2 and the corresponding isolates (see Sections 5.4.4 and 5.4.5). Similarly, the parameters selected for the setting CV3 (randomly hiding drug entities) have been used to evaluate the performance of the approaches in Section 5.4.3.

5.4 Results

We assess the performance of different matrix completion techniques in this section. The techniques have been described in the Methods section. Six matrix completion methods were used, which can be categorized into three families provided below.

- Basic frameworks (MF: Matrix factorization [126] and MC: Matrix completion or Nuclear norm minimization [126])
- Deep frameworks (DMF: Deep matrix factorization) [128],
- Graph regularized frameworks (GRMF: graph regularized matrix factorization [14], GRMC: graph regularized matrix completion [21], GRBMC: graph regularized binary matrix completion [20])

Matrix factorization (MF) is the traditional matrix completion method which factorizes the data matrix into two latent factor matrices (tall and short) and the

algorithm recovers these factor matrices to recover the original matrix. Since this problem is non-convex, it may not converge to a global minimum of the cost function. Nuclear-norm minimization based matrix completion (MC) was proposed as a (mathematically) better alternative; it directly recovers the matrix by penalising its nuclear norm (convex surrogate of rank). Deep matrix factorization (DMF) generalises MF to more than two factors. None of the techniques mentioned so far can take advantage of genomic structure of the viruses or chemical structure of the drugs. The said pieces of information can be incorporated into the graph regularized matrix completion techniques (GRMF, GRMC, GRBMC). These techniques have been explained in detail in the Methods section.

5.4.1 Overview: DVA prediction

The typical anti-viral drug discovery process involves genomic and biophysical understanding of the virus. It aims to target the enzymes or peptides involved in the viral replication cycle and takes years for successful clinical validation. Other approaches involve screening all the broad-spectrum anti-viral drugs or chemical libraries comprising large numbers of existing compounds/databases (having information on transcriptional signatures in different cell lines) to be further evaluated by standard anti-viral assays [129]. In view to assist acceleration of this process (by pruning down the search space), we create and share a publicly available DVA database, along with a number of matrix completion techniques (mentioned above) for drug-virus association prediction.

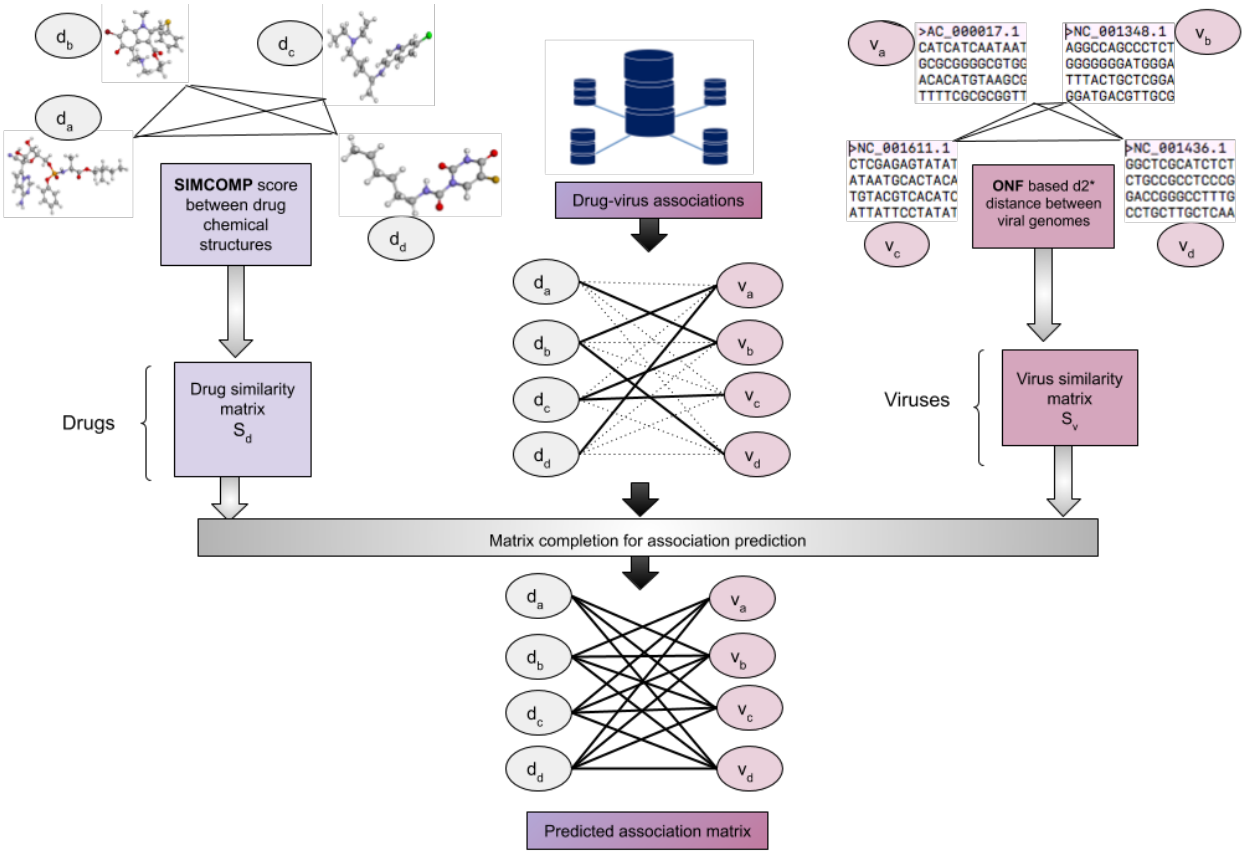


Figure 5.1: Schematic diagram depicting the DVA framework

The originality of the proposed work lies in the formalization of the drug-virus association prediction as a matrix completion problem, without the need for any anti-viral assays. Such a computational approach requires the chemical structure of the drugs and, in case of graph-regularized matrix completion techniques, the genome of the viruses, or existing associations otherwise. Figure 6.5 depicts the schematic flow of the proposed work involving data curation and implementation overview.

5.4.2 Empirical evaluation

In this sub-section, we carry out extensive experimental protocol to illustrate and compare the ability of the different methods to retrieve the drug-disease associations available in our curated dataset. The protocol dictates three variants of κ-fold (with κ=10 here) cross-validation setting (CV) as described in subsection 3.4.1 of Chapter 3. In the first setting CV1 (cross validation 1), 10 % of the associations selected at random are left out as testing set. This allows to assess each algorithm's ability to predict associations between existing drugs and viruses. To evaluate an algorithm for its ability to predict association for novel drugs and viruses i.e. those which have no association information, we use two other (more stringent) CV settings. In CV2 and CV3, 10 % of the complete virus and drug entities selected at random are left out as test set respectively.

The standard metrics for evaluation are the AUC (Area under the Receiver Operating Characteristic curve) and AUPR (Area under the precision-recall curve). AUC is more common in machine learning literature, it assumes that the classes are evenly balanced. Problems in drug-disease association have highly imbalanced classes, in such a scenario the AUPR is a more appropriate metric for evaluation [90, 14].

Table 5.1 shows how each of the 6 tested algorithms performs in retrieving the associations. A clear observation from the experiments is that the graph regularized-based matrix completion algorithms that incorporate the similarity information associated with the drugs and viruses, perform fairly well giving

an AUC greater or equal than 0.83 in CV1. The best performing algorithm (GRBMC) exhibits an AUC and AUPR of 0.88 and 0.54 respectively. Predicting the associations for novel drugs and viruses also have a reasonable performance with the best AUC/AUPR of 0.81/0.44 and 0.73/0.31 by GRBMC and GRMF. It can be noted that the standard matrix completion methods, which do not take into account the metadata, fail to learn from the association data giving a near-random performance as far as the prediction on novel viruses is concerned, depicting how very important the similarity information is.

	Metric	MC	MF	DMF	GRMF	GRMC	GRBMC
CV1	AUC	0.5959	0.6753	0.6974	0.8652	0.8279	0.8834
	AUPR	0.3238	0.2656	0.2615	0.4812	0.4445	0.5220
CV2	AUC	0.4909	0.5033	0.5704	0.7346	0.6705	0.6632
	AUPR	0.1106	0.0504	0.0855	0.3112	0.2951	0.2746
CV3	AUC	0.5438	0.5215	0.4529	0.7806	0.7507	0.8181
	AUPR	0.0538	0.0637	0.0824	0.4265	0.4333	0.4383

Table 5.1: Results for association prediction for all techniques under the 3 cross validation settings.

5.4.3 Association prediction for new drugs

DVA database and its associated computational tools can also be used on new drugs without any previously known virus association information. For evaluating this ability, we identified in our database all the drugs which are known to interact with only one virus (drugs associated with a single virus only) and hide that association to the methods. This allows us to assess the performance of the algorithms in predicting viruses associated with the new drugs in the database.

We hide the only virus corresponding to each of the 76 drugs (with only a

single virus associated with it) and run matrix completion to predict candidate viruses for these drugs. The drugs for which the test virus associated with it is the top-ranked virus predicted by the algorithm would have the maximum precision value (MPV) of 1. The number and percentage of drugs with a maximum precision value of 1 are reported in Table 5.2.

Nearly 34 % (26/76) of single association drugs with a maximum precision of 1 were predicted using GRMF. Other graph regularized frameworks show comparable performance in terms of predicting drugs with MPV of 1.

	MC	MF	DMF	GRMF	GRMC	GRBMC
# drugs with MPV=1	2	4	4	26	22	8
% drugs with MPV=1	2.6316	5.2632	5.2632	34.2105	28.9474	10.5263

Table 5.2: Number and percentage of drugs predicted with MPV=1 by the matrix completion methods.

5.4.4 SARS-CoV-2 prediction

In this experiment, we add the SARS-CoV-2 sample in our database by providing its ONF based d2* similarity **[124]** in the virus similarity matrix.

We then apply the matrix completion algorithms to predict the associations and rank prediction scores corresponding to SARS-CoV-2 to predict the top 10 recommended drugs.

As can be seen from the results of section 5.4.2 (Table 5.1), MC, MF and DMF often yield considerably worse results than their graph regularized counterparts (GRMF, GRMC and GRBMC). Such poor performance of non-graph regularized versions of matrix completion methods could be explained as they

Technique	SARS-Cov-2
GRMF	Remdesivir
	Ribavirin
	Sofosbuvir
	Umifenovir
	Taribavirin
	Tenofovir alafenamide
	Ibuprofen
	Pleconaril
	Geldanamycin
	Vidarabine
GRMC	Remdesivir
	Ribavirin
	Sofosbuvir
	Taribavirin
	Tenofovir alafenamide
	Vidarabine
	Telaprevir
	Boceprevir
	Simeprevir
	Palivizumab
GRBMC	Remdesivir
	Ribavirin
	Sofosbuvir
	Umifenovir
	Taribavirin
	Vidarabine
	Brivudine
	Tenofovir alafenamide
	Paritaprevir
	Peginterferon alfacon-1

Table 5.3: Top-10 drugs predicted for SARS-Cov-2 by the DVA computational methods.

do not incorporate any knowledge about the genomic structure of the viruses and the chemical structure of the drugs. Since the three graph-based methods perform reasonably well in the prediction task, we consider these techniques for the drug prediction on the novel coronavirus. The top-10 drugs they predicted have been reported in Table 5.3 (ranked by their predicted scores). Drugs highlighted with blue text are unanimously predicted drugs by the three considered matrix completion techniques and those in red text are predicted by two

methods. We also highlighted with yellow cells the drugs which are under trial/investigation as a potential cure/prophylactic against COVID-19.

It can be seen that the three techniques have consistently and unanimously selected six drugs, namely Remdesivir, Ribavarin, Sofosbuvir, Taribavirin, Tenofovir alafenamide and Vidarabine. Umifenovir has been recommended by two (GRMF and GRBMC) out of three techniques. Amongst these recommendations, Remdesivir [130], Ribavarin [131, 132], Sofosbuvir [133] and Umifenovir [134] are under clinical trials. Taribavirin is similar to Ribavirin but it is not approved by the FDA. Tenofovir alafemanide (an antiretroviral for HIV-1) is on undergoing trial [135]. GRMF has additionally selected Ibuprofen which is expected to be investigated in UK [136, 137]. The fact that three techniques unanimously select the aforementioned drugs make us confident about these recommendation results.

5.4.5 Predictions evolution with mutating novel coronavirus

In the previous sub-section, we have established that the results from our models are mostly in sync with clinical practice. In this sub-section, we will demonstrate how our proposed approach can be of help to clinicians.

All the results generated so far have been generated using the reference sequence of the SARS-Cov-2 strain (collected in December, 2019 in Wuhan). The novel coronavirus is rapidly mutating [109]. In such a scenario, it is necessary to select drugs that are effective against the mutated strain. While mutating,

the virus isolates may develop resistance to previous drugs used for its treatment. Our model may be of help to clinicians in this respect. Before proposing a treatment regime (trial, for e.g.) for COVID-19 treatment, the practitioner may use our approach to check the drugs selected for the particular isolate of novel coronavirus. In Table 5.4, we have experimented with three isolates of the novel coronavirus (collected over an interval of 2 months), in addition to the reference sequence (collected in December 2019). Those three isolates have been collected in February (from USA), April (from Australia) and June (from India).

One can note from the Table 5.4 that the selected drugs change with mutations. Baloxavir marboxil was not selected even once for the reference sequence from December 2019, but has been selected by two methods for the February isolate. A recent pre-print [138] reports the results of this antiviral on COVID-19 patients. The drug Ibuprofen, was selected by one of the methods for the December reference sequence, it was not selected for the February isolate, then it was selected by two methods for the April isolate and selected by all three for the June isolate. It may be worthy to note that lipid Ibuprofen is being considered in a trial in UK from starting June, 2020 [136]. Similarly, Pleconaril has been selected for by all three methods for the most recent (June) isolate, it was selected by only one of the techniques for the reference sequence (December) and was not selected for the February or June sequences. Pleconaril, although developed for treating enterovirus and rhinovirus, is not FDA approved. Rilpivirine and Etravirine are two antiretrovirals developed for treating HIV

positive subjects. Both of them have been predicted by all three methods in the latest isolate, but not in the previous isolates or in the reference sequence. To the best of our knowledge, this antiretroviral is not under study for COVID-19 trials. Note that Vidarabine, which was getting predicted for the reference sequence (albeit wrongly) has not been predicted from the later ones. Based on this discussion, we can see that how the mutations in genomic structure results in different predictions of drugs. Since the novel coronavirus is mutating, it may be judicious to account for the structure of the latest isolate while deciding the treatments to be put in trial. In such a case, our model may be of help to clinicians.

5.5 Conclusion

Computational techniques have the inherent advantage of learning from the data (which can be huge given a large number of drugs to be tested) and scale to a large number of drugs and viruses and hence be of immense importance to the clinicians by narrowing down the search space for the clinical trials to be carried out.

We would like to emphasize that the proposed DVA database and methods are not particular to the novel coronavirus. Such computational approaches have the general capability to help for identification of drugs which might be effective against a broad spectrum of viruses [139], or the viruses which can be targeted by multiple drugs (since many drugs could target specific elements of

Technique	SARS-Cov-2: february, 2020	SARS-Cov-2: april, 2020	SARS-Cov-2: june, 2020
GRMF	Remdesivir	Remdesivir	Remdesivir
	Ribavirin	Sofosbuvir	Umifenovir
	Umifenovir	Umifenovir	Pleconaril
	Taribavirin	Ribavirin	Ibuprofen
	Sofosbuvir	Tenofovir alafenamide	Sofosbuvir
	Baloxavir marboxil	Ibuprofen	Rilpivirine
	Geldanamycin	Pleconaril	Etravirine
	Tenofovir alafenamide	Hydroxychloroquine	Tenofovir alafenamide
	Tecovirimat	Valomaciclovir	Rimantadine
	Peramivir	Dexamethasone	Ribavirin
GRMC	Remdesivir	Remdesivir	Umifenovir
	Umifenovir	Sofosbuvir	Remdesivir
	Ribavirin	Tenofovir alafenamide	Ibuprofen
	Taribavirin	Boceprevir	Pleconaril
	Sofosbuvir	Telaprevir	Sofosbuvir
	Vidarabine	Palivizumab	Chloroquine
	Tenofovir alafenamide	Simeprevir	Etravirine
	Nelfinavir	Ribavirin	Rilpivirine
	Amprenavir	Umifenovir	Tenofovir alafenamide
	Boceprevir	Ibuprofen	Nelfinavir
GRBMC	Remdesivir	Remdesivir	Umifenovir
	Ribavirin	Umifenovir	Remdesivir
	Umifenovir	Sofosbuvir	Pleconaril
	Taribavirin	Ribavirin	Ibuprofen
	Sofosbuvir	Taribavirin	Sofosbuvir
	Paritaprevir	Paritaprevir	Rilpivirine
	Tenofovir alafenamide	Brivudine	Etravirine
	Atazanavir	Vidarabine	Ribavirin
	Baloxavir marboxil	Daclatasvir	Tenofovir alafenamide
	Favipiravir	Beclabuvir	Trifluridine

Table 5.4: Top-10 drugs predicted for three isolates of SARS-Cov-2 (collected at an interval of 2 months) by the DVA computational methods.

viral replication) [140]. We believe that the proposed work will pave the way for more scientific ideas for anti-viral drug re-positioning and assist clinicians in the process.

Chapter 6

Transcriptomic-proteomic expression completion using collaborative matrix completion

A very recent technology, REAP-seq (RNA expression and protein sequencing assay sequencing) allows simultaneous measurement of gene and protein expression levels in single cells. Both the expression profiles of genes and proteins are not complete (gene expression profiles being more sparse). This calls for an imputation framework that has the ability to impute both of these. In this work, we propose a collaborative matrix completion framework that performs matrix completion on both transcriptomic and proteomic data, using cell-information for the proteomic and transcriptomic counterparts. This work is still ongoing and is expected to yield meaningful biological results. The algorithmic framework proposed has been shown in this dissertation.

6.1 Introduction

The benefits of single-cell RNA sequencing enables the measurement of gene expressions in individual cells, assisting the discovery of novel or rare cell types. The measurement of gene expression also helps in giving an insight into the mechanisms of cellular development and cellular response to therapeutics. But the mRNA abundance does not infer the protein abundance, which is the primary target for drugs. To have a view of the proteins in cells for modeling the response to therapeutics, new technologies have been introduced [141, 23]. For instance, REAP-seq (RNA expression and protein sequencing assay) allows the simultaneous measurement of mRNA and proteins in single cells. But, due to low starting mRNA material in one cell in the reverse transcription step, the gene expression data obtained has dropouts. Not only this, the protein expression data is not fully complete because of the very few numbers of proteins profiled in a cell. Hence, there is a need for a collaborative matrix completion framework that simultaneously completes the transcriptomic and proteomic profiles. In this work, we propose such a collaborative framework, which uses the information in proteomic data to impute transcriptomic data and vice-versa.

The proposed algorithm employs graph regularized nuclear norm minimization introduced in chapter 3. The formulation has been modified to take care of the simultaneous imputation goal as shown in section 6.3.

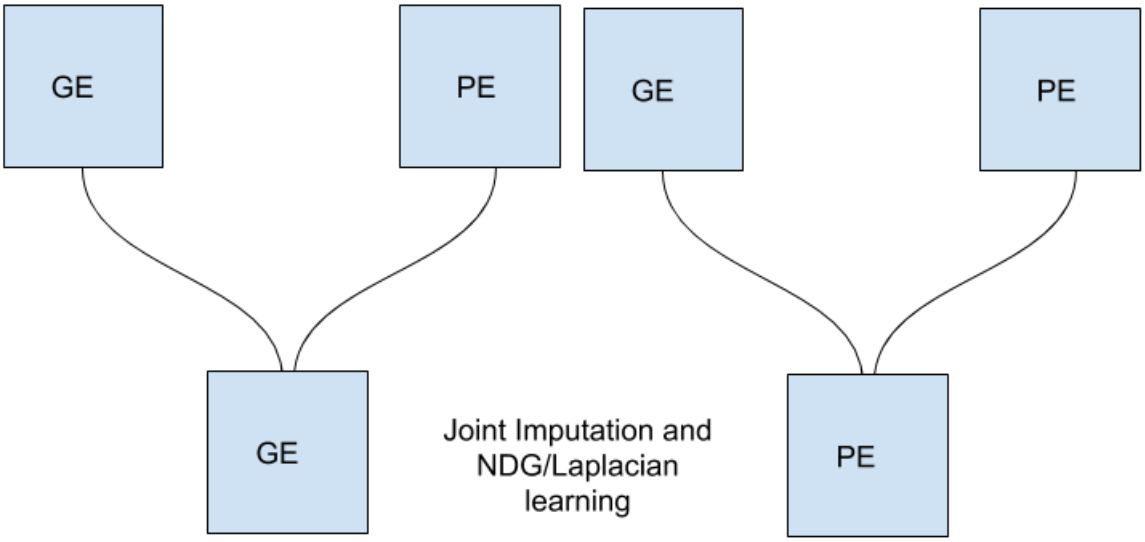


Figure 6.1: Schematic diagram depicting the collabrative imputation framework

6.2 Dataset

The gene and protein expression data has been generated using REAP-seq [23] and comprises of 1723 cells in gene expression and 1668 cells in protein expression data.

The imputation and entire analysis have been performed by taking 1453 common samples after cell filtering (removing zero count cells) from both.

6.3 Methodology

For the imputation of Gene expression matrix G (having cells on rows and genes on columns) and Protein-expression matrix P (with cells on rows and proteins on columns), we exploit the graph laplacians from both G and P.

6.3.1 Data preprocessing

We process the gene and protein expression matrices by performing the following steps:

- Cell filtering: To remove low-quality cells, the expression of all genes/proteins for every cell (c_j) is added to give a vector (v) of size n, where n is the total number of cells present in the expression data ($1jn$). This vector (v) is divided into four quartiles. The 1.5 ×IQR (inter-quartile region) rule was applied to detect the outlier cells in the genomic/proteomic space. The cells with total expression below $Q_1 - 1.5 * IQR$ and above $Q_3 + 1.5 * IQR$ are discarded (Q_1, Q_2, and Q_3 denote quantiles or cut-points used to partition the data into quartiles).
- Gene selection: The genes with read count ≥ 3 in at least 3 cells are considered as filtered genes (as done in Chapter 2). The rest of the genes which do not satisfy the above criteria are eliminated.

6.3.2 Proposed framework

To solve the simultaneous imputation of G and P, we use Collaborative Graph regularized Matrix completion. Both the matrices have cells on rows, and genes in G and proteins in P on column.

6.3.2.1 Formulation

The formulation to jointly co-complete both the gene expression and protein expression matrices is shown below:

$$\min_{G,P,L_G,L_P} ||Y_G - M_G \circ (G)||_F^2 + \lambda_G||G||_* + Tr(G^T(\alpha_G L_G + \beta_G L_P)G) + \\ ||Y_P - M_P \circ (P)||_F^2 + \lambda_P||P||_* + Tr(P^T(\alpha_P L_P + \beta_P L_G)P) \tag{6.1}$$

Motivated by [142], we replace the trace terms in the above objective function by weighted l_1 norm of W which enables us to alternatively solve the matrix completion and graph learning task jointly.

$$\begin{aligned} &\min_{G,P,W_G,W_P} ||Y_G - M_G \circ (G)||_F^2 + \lambda_G||G||_* + \\ &\alpha_G\{||W^G \circ Z^G||_{1,1} + \sigma_G^2 \Sigma_{i,j} W_{i,j}^G (log(W_{i,j}^G) - 1)\} + \\ &\beta_G\{||W^P \circ Z^G||_{1,1} + \sigma_P^2 \Sigma_{i,j} W_{i,j}^P (log(W_{i,j}^P) - 1)\} + \\ &||Y_P - M_P \circ (P)||_F^2 + \lambda_P||P||_* + \\ &\alpha_P\{||W^P \circ Z^P||_{1,1} + \sigma_P^2 \Sigma_{i,j} W_{i,j}^P (log(W_{i,j}^P) - 1)\} + \\ &\beta_P\{||W^G \circ Z^P||_{1,1} + \sigma_G^2 \Sigma_{i,j} W_{i,j}^G (log(W_{i,j}^G) - 1)\} \end{aligned} \tag{6.2}$$

6.3.2.2 Solution

This subsection shows how we solve the above proposed formulation for collaborative matrix completion.

We use ADMM to solve it and employ variable separation method to obtain the solution for each of the unknowns ($G, P, L_G or W_G, L_P or W_P$) as shown below,

$$G \leftarrow \min_{G,P,W_G,W_P} ||Y_G - M_G \circ (G)||_F^2 + \lambda_G ||G||_* + \alpha_G \{||W^G \circ Z^G||_{1,1}\} +$$
$$\beta_G \{||W^P \circ Z^G||_{1,1}\}$$
$$OR$$
$$G \leftarrow \min_{G,P,W_G,W_P} ||Y_G - M_G \circ (G)||_F^2 + \lambda_G ||G||_* + \alpha_G \{Tr(G^T L_G G)\} +$$
$$\beta_G \{Tr(G^T L_P G)\}$$
$$OR$$

$$\mathbf{G} \leftarrow \min_{G} ||Y_G - M_G \circ (G)||_F^2 + \lambda_G ||G||_* + Tr(G^T(\alpha_G L_G + \beta_G L_P)G) \tag{6.3}$$

Similarly,

$$\mathbf{P} \leftarrow \min_{P} ||Y_P - M_P \circ (P)||_F^2 + \lambda_P ||P||_* + Tr(P^T(\alpha_P L_P + \beta_P L_G)P) \tag{6.4}$$

Solution to equations (6.3) and (6.4) is shown in section 3.3.2.

$$\mathbf{W}^G_{i,j} \leftarrow \alpha_G\{||W^G \circ Z^G||_{1,1} + \sigma^2_G \Sigma_{i,j} W^G_{i,j}(log(W^G_{i,j}) - 1)\} + \\ \beta_P\{||W^G \circ Z^P||_{1,1} + \sigma^2_G \Sigma_{i,j} W^G_{i,j}(log(W^G_{i,j}) - 1)\} \tag{6.5}$$

Differentiating equation (6.5) wrt $W^G_{i,j}$ and equating to 0, we get:

$\alpha_G(Z^G + \sigma^2_G log(W^G_{i,j})) + \beta_P(Z^P + \sigma^2_G log(W^G_{i,j})) = 0$

$(\alpha_G + \beta_P)(\sigma^2_G log(W^G_{i,j})) = -(\alpha_G Z^G + \beta_P(Z^P)$

$W^G_{i,j} = exp(-\frac{\alpha_G||g_i - g_j||^2_2 + \beta_P||p_i - p_j||^2_2}{(\alpha_G + \beta_P)\sigma^2_G})$

Similarly,

$W^P_{i,j} = exp(-\frac{\alpha_P||p_i - p_j||^2_2 + \beta_G||g_i - g_j||^2_2}{(\alpha_P + \beta_G)\sigma^2_P})$

NOTE: The above expression can be written as:

$W^G_{i,j} = exp(-\frac{\alpha_G||g_i - g_j||^2_2}{(\alpha_G + \beta_P)\sigma^2_G}) \times exp(-\frac{\beta_P||p_i - p_j||^2_2}{(\alpha_G + \beta_P)\sigma^2_G})$

6.4 Results

This section shows the preliminary results of applying this framework to the gene and protein expression data. In the future, we would add more kinds of biological validation to evaluate the proposed method.

6.4.1 Cell visualization

We represent the transcriptomic and proteomic data visually by reducing it to a two-dimensional space and coloring each cell by its cell type. Since t-SNE is shown to be well-suited for the visualization tasks [143], we use t-SNE (with perplexity=30) on both the expression matrices.

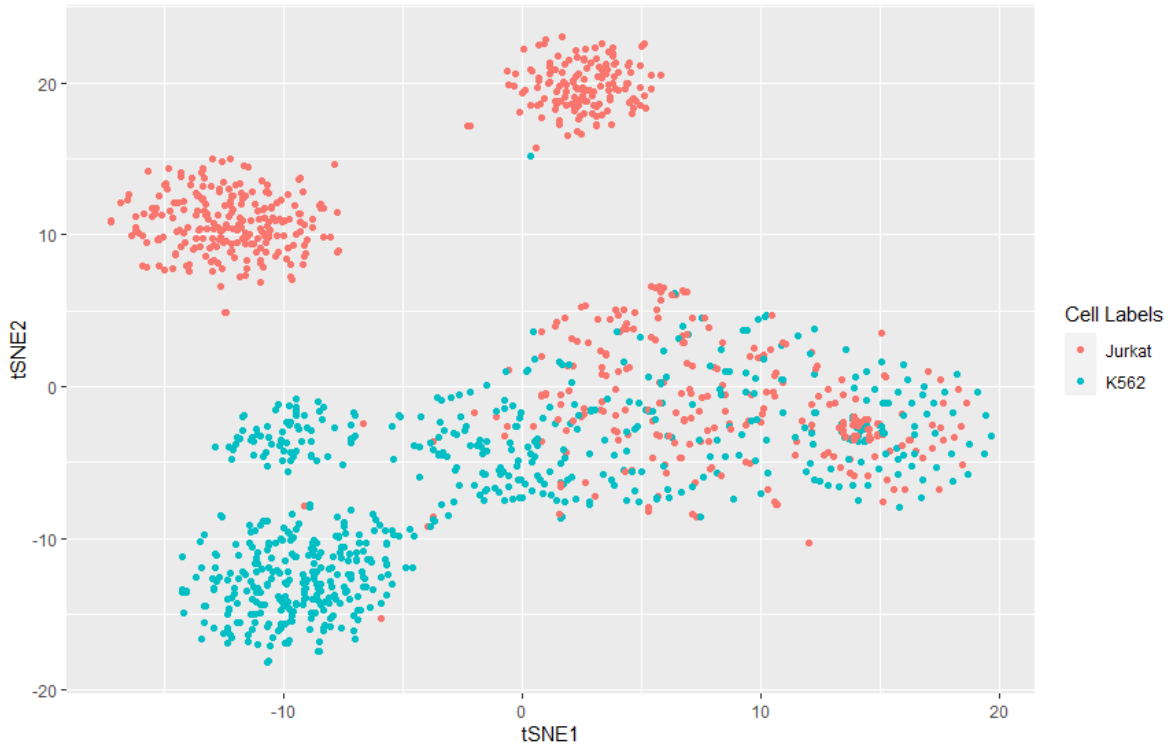


Figure 6.2: t-SNE representation of cells from unimputed gene expression data

As it can be observed, the imputation has helped the cells of the same type to come close together in the t-SNE space, when looking at the transcriptomic view. The same is not the case when looking at the proteomic view. This can be attributed to the fact that the information derived from gene-data is not as reliable (due to dropouts) for the completion of protein-data.

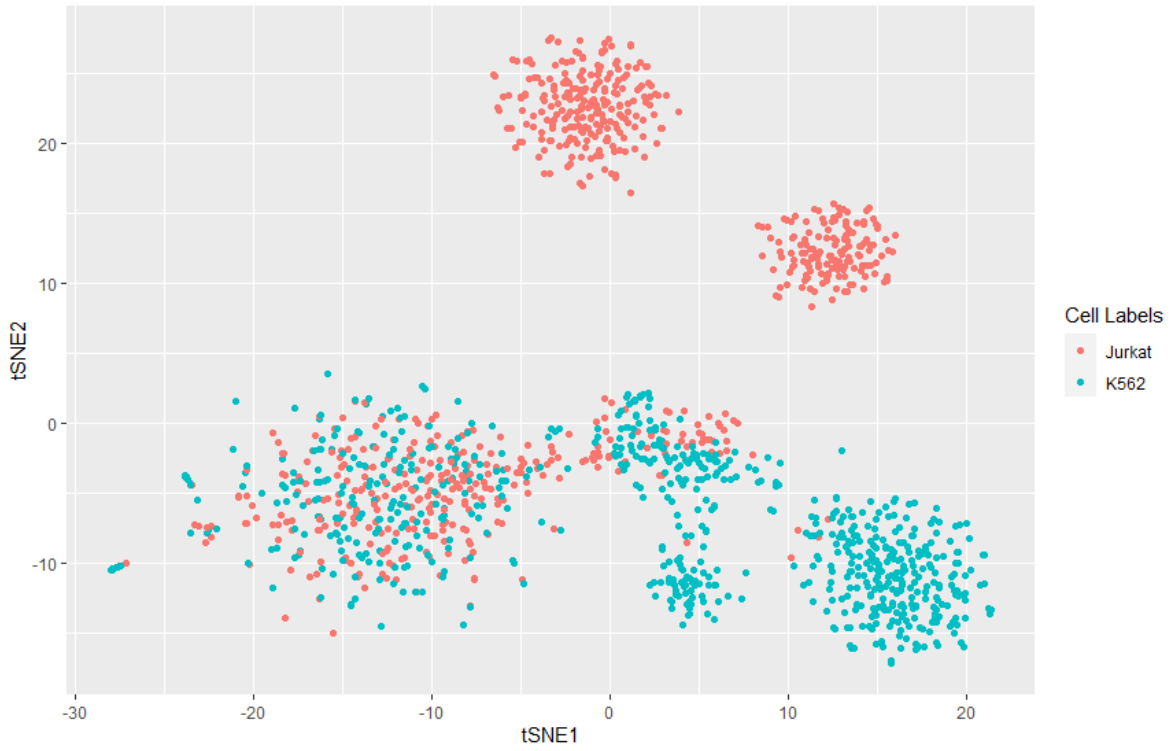


Figure 6.3: t-SNE representation of cells from imputed gene expression data

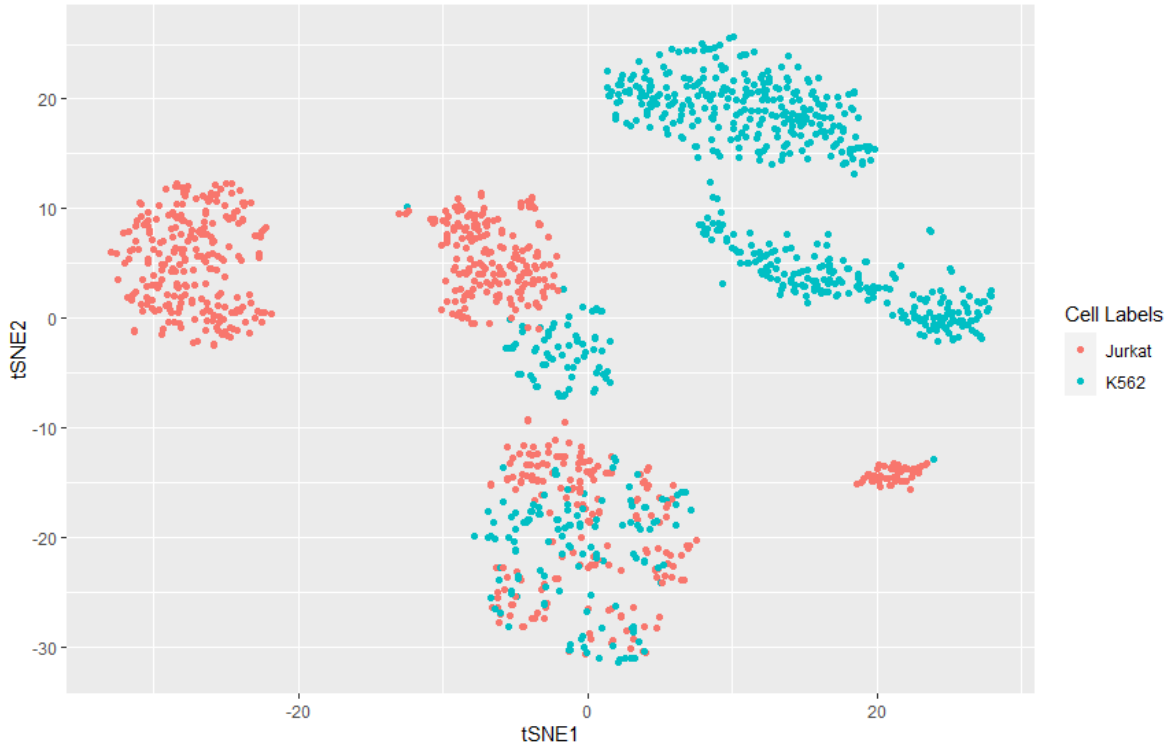


Figure 6.4: t-SNE representation of cells from unimputed protein expression data

Figure 6.5: t-SNE representation of cells from imputed protein expression data

6.5 Conclusion

In this work, we propose a collaborative matrix completion framework to impute gene and protein expression data obtained using a multiplexed quantification of genes and protein in single cells. The work is ongoing and is expected to improve the underlying downstream analysis after imputation. The method can also be used for other problems where simultaneous matrix completion is required, reaping benefits of matrix completion and side-information/graph-regularization.

Chapter 7

Conclusion

The proposed work in this thesis focuses on modeling various prediction or imputation problems in the field of Bioinformatics as Matrix completion problem, making use of the biological insights and the algorithmic techniques.

7.1 Summary of contribution

In this section, we briefly summarize the chapter-wise contribution giving a bird's eye view to the dissertation.

7.1.1 scRNA-seq imputation using matrix completion frameworks

In this part of the dissertation, we primarily model scRNA-seq data imputation for dropouts as a low-rank matrix completion problem. Although the model uses existing methods, ours was one of the first works in the field to have modeled the single-cell transcriptomic imputation using a novel approach exploiting

advantages of deep learning and matrix factorization based matrix completion: deepMc (deep matrix completion).

7.1.2 Drug-target Interaction prediction using multi graph regularized nuclear norm minimization

This chapter introduces a Drug-target imputation framework that can incorporate multiple kinds of metadata/side-information of the drugs and target entities involved. In particular, multiple types of similarity information between drugs and targets have been used to predict interactions between them. Although the contribution is largely incremental in terms of biological application, the novelty lies in the algorithm, which is the first framework to incorporate multiple types of similarity/graph-laplacians associated with drugs and targets.

7.1.3 Drug-disease association prediction using graph-regularized one bit matrix completion

This work is motivated by the fact that the values to be imputed in an association problem are binary. None of the existing matrix completion methods take care of this constraint. Hence, we ensure that the prediction is in the range [0,1] by solving the graph-regularized matrix completion in a different manner-PPXA (Parallel proximal algorithm). The results on drug-disease association prediction are at-par with the state-of-the-art and even better in some specific evaluation strategies.

7.1.4 Drug-virus association database: anti-viral drug prediction using Matrix completion

This part of the dissertation is overall the most crucial contribution. It puts-forward the first-ever drug-virus association database which can be explored, analyzed, and used for deploying other computational artificial intelligence/machine learning approaches apart from the one proposed in this dissertation to help clinicians in selecting a few antivirals that can be tried out for a particular virus. Apart from the database, we also propose to use the association and similarity information between drugs and viruses collected by deploying matrix completion frameworks. As can be seen, the dissertation already solves associated problems like- drug-target interaction and drug-disease association. Hence, it was only natural to address the most important pandemic (COVID-19) of our generation when the necessity arose by creating such a database and predicting drugs for the novel coronavirus (SARS-Cov-2).

7.1.5 Transcriptomic-proteomic expression completion using collaborative matrix completion

In the last ongoing work, we have created an imputation framework that has the capability to perform matrix completion simultaneously on transcriptomic and proteomic data using cell-information from each other. Such a framework can also be used for other problems where simultaneous matrix completion is required, reaping benefits of matrix completion and side-information/graph-regularization.

7.2 Future work

The algorithms proposed are generic and can not only be used in other bioinformatics problems like protein-protein interaction [91], RNA-RNA interaction [92], etc but also other research fields where one could capture the side-information associated with the data by finding the similarities amongst row entities and column entities and use them to learn deep representations.

Specifically, One may like to explore other similar interaction problems like

- **Cancer-drug response prediction:** As we move towards an era of precision medicine, the ability to predict patient-specific drug responses in cancer-based on molecular information such as gene expression data represent both an opportunity and a challenge. In particular, methods are needed that can accommodate the high-dimensionality of data to learn interpretable models capturing drug response mechanisms, as well as providing robust predictions across datasets. Prediction of cancer drug responses for unseen cell-lines/patients is a crucial problem.

- **Gene-disease association prediction:** Correctly identifying associations of genes with diseases has long been a goal in biology. With the emergence of large-scale gene-phenotype association datasets in biology, we can leverage statistical and machine learning methods to help us achieve this goal

- **Microbe-disease association prediction:** Accumulating clinical observa-

tions have indicated that microbes living in the human body are closely associated with a wide range of human non-infectious diseases, which provides promising insights into the complex disease mechanism understanding. Predicting microbe–disease associations could not only boost human disease diagnostic and prognostic but also improve the new drug development. However, little efforts have been attempted to understand and predict human-microbe disease associations on a large scale until now

References

[1] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[2] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009. [Online]. Available: http://dx.doi.org/10.1109/MC.2009.263

[3] A. Majumdar and R. Ward, "Some empirical advances in matrix completion," *Signal Process.*, vol. 91, no. 5, pp. 1334–1338, May 2011. [Online]. Available: http://dx.doi.org/10.1016/j.sigpro.2010.12.005

[4] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. MIT Press, 2001, pp. 556–562. [Online]. Available: http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf

[5] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning,"

Trans. Sig. Proc., vol. 65, no. 3, pp. 794–816, Feb. 2017. [Online]. Available: https://doi.org/10.1109/TSP.2016.2601299

[6] T. Hastie, R. Mazumder, J. D. Lee, and R. Zadeh, "Matrix completion and low-rank svd via fast alternating least squares," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 3367–3402, 2015.

[7] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inf. Theor.*, vol. 56, no. 5, pp. 2053–2080, May 2010. [Online]. Available: http://dx.doi.org/10.1109/TIT.2010.2044061

[8] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, Dec. 2009. [Online]. Available: http://dx.doi.org/10.1007/s10208-009-9045-5

[9] E. J. Candès and Y. Plan, "Matrix completion with noise," *CoRR*, vol. abs/0903.3131, 2009. [Online]. Available: http://arxiv.org/abs/0903.3131

[10] S. Min, B. Lee, and S. Yoon, "Deep learning in bioinformatics," *Briefings in bioinformatics*, vol. 18, no. 5, pp. 851–869, 2017.

[11] Z. Li and J. Tang, "Weakly supervised deep matrix factorization for social image understanding," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 276–288, 2017.

[12] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE*

transactions on pattern analysis and machine intelligence, vol. 39, no. 3, pp. 417–429, 2017.

[13] S. Tariyal, A. Majumdar, R. Singh, and M. Vatsa, "Deep dictionary learning," *IEEE Access*, vol. 4, pp. 10 096–10 109, 2016.

[14] A. Ezzat, P. Zhao, M. Wu, X.-L. Li, and C.-K. Kwoh, "Drug-target interaction prediction with graph regularized matrix factorization," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 14, no. 3, pp. 646–656, 2017.

[15] M. Wang, C. Tang, and J. Chen, "Drug-target interaction prediction via dual laplacian graph regularized matrix completion," *BioMed Research International*, vol. 2018, 2018.

[16] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1548–1560, 2010.

[17] F. R. Chung, "Spectral graph theory (cbms regional conference series in mathematics, no. 92)," 1996.

[18] A. Mongia, D. Sengupta, and A. Majumdar, "deepmc: Deep matrix completion for imputation of single-cell rna-seq data," *Journal of Computational Biology*, vol. 27, 2019.

[19] ——, "Mcimpute: Matrix completion based imputation for single cell rna-seq data," *bioRxiv*, p. 361980, 2018.

[20] A. Mongia, E. Chouzenoux, and A. Majumdar, "Computational prediction of drug-disease association based on graph-regularized one bit matrix completion," *bioRxiv*, 2020, https://www.biorxiv.org/content/10.1101/2020.04.02.020891v1.abstract.

[21] A. Mongia and A. Majumdar, "Drug-target interaction prediction using multi graph regularized nuclear norm minimization," *Plos One*, vol. 15, no. 1, p. e0226484, 2020.

[22] A. Mongia, S. K. Saha, E. Chouzenoux, and A. Majumdar, "A computational approach to aid clinicians in selecting anti-viral drugs for covid-19 trials," *arXiv preprint arXiv:2007.01902*, 2020.

[23] V. M. Peterson, K. X. Zhang, N. Kumar, J. Wong, L. Li, D. C. Wilson, R. Moore, T. K. McClanahan, S. Sadekova, and J. A. Klappenbach, "Multiplexed quantification of proteins and transcripts in single cells," *Nature biotechnology*, vol. 35, no. 10, pp. 936–939, 2017.

[24] K. Liu, J. Ye, Y. Yang, L. Shen, and H. Jiang, "A unified model for joint normalization and differential gene expression detection in rna-seq data," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2018.

[25] J. Li, J. Hu, M. Newman, K. Liu, and H. Ge, "Rna-seq analysis pipeline based on oshell environment," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 5, pp. 973–978, Sept 2014.

[26] J. M. Knight, I. Ivanov, K. Triff, R. S. Chapkin, and E. R. Dougherty, "Detecting multivariate gene interactions in rna-seq data using optimal bayesian classification," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 2, pp. 484–493, March 2018.

[27] A. Wagner, A. Regev, and N. Yosef, "Revealing the vectors of cellular identity with single-cell genomics," *Nature biotechnology*, vol. 34, no. 11, pp. 1145–1160, 2016.

[28] K. AleksandraA., K. K. Jong, S. Valentine, M. JohnC., and T. SarahA., "The technology and biology of single-cell rna sequencing," *Molecular Cell*, vol. 58, no. 4, pp. 610 – 620, 2015.

[29] Z. Wang, M. Gerstein, and M. Snyder, "Rna-seq: a revolutionary tool for transcriptomics," *Nature reviews genetics*, vol. 10, no. 1, pp. 57–63, 2009.

[30] S. Rizzetto, A. A. Eltahla, P. Lin, R. Bull, A. R. Lloyd, J. W. Ho, V. Venturi, and F. Luciani, "Impact of sequencing depth and read length on single cell rna sequencing data of t cells," *Scientific Reports*, vol. 7, no. 1, p. 12781, 2017.

[31] A. P. Patel, I. Tirosh, J. J. Trombetta, A. K. Shalek, S. M. Gillespie, H. Wakimoto, D. P. Cahill, B. V. Nahed, W. T. Curry, R. L. Martuza *et al.*, "Single-cell rna-seq highlights intratumoral heterogeneity in primary glioblastoma," *Science*, vol. 344, no. 6190, pp. 1396–1401, 2014.

[32] I. Tirosh, B. Izar, S. M. Prakadan, M. H. Wadsworth, D. Treacy, J. J. Trombetta, A. Rotem, C. Rodman, C. Lian, G. Murphy *et al.*, "Dissecting the multicellular ecosystem of metastatic melanoma by single-cell rna-seq," *Science*, vol. 352, no. 6282, pp. 189–196, 2016.

[33] L. Yan, M. Yang, H. Guo, L. Yang, J. Wu, R. Li, P. Liu, Y. Lian, X. Zheng, J. Yan *et al.*, "Single-cell rna-seq profiling of human preimplantation embryos and embryonic stem cells," *Nature structural & molecular biology*, vol. 20, no. 9, pp. 1131–1139, 2013.

[34] F. Tang, C. Barbacioru, S. Bao, C. Lee, E. Nordman, X. Wang, K. Lao, and M. A. Surani, "Tracing the derivation of embryonic stem cells from the inner cell mass by single-cell rna-seq analysis," *Cell stem cell*, vol. 6, no. 5, pp. 468–478, 2010.

[35] F. H. Biase, X. Cao, and S. Zhong, "Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell rna sequencing," *Genome research*, vol. 24, no. 11, pp. 1787–1796, 2014.

[36] P. V. Kharchenko, L. Silberstein, and D. T. Scadden, "Bayesian approach to single-cell differential expression analysis," *Nature methods*, vol. 11, no. 7, pp. 740–742, 2014.

[37] D. van Dijk, J. Nainys, R. Sharma, P. Kathail, A. J. Carr, K. R. Moon, L. Mazutis, G. Wolf, S. Krishnaswamy, and D. Pe'er, "Magic: A diffusion-based imputation method reveals gene-gene interactions in single-cell rna-sequencing data," *BioRxiv*, p. 111591, 2017.

[38] H. Li, E. T. Courtois, D. Sengupta, Y. Tan, K. H. Chen, J. J. L. Goh, S. L. Kong, C. Chua, L. K. Hon, W. S. Tan *et al.*, "Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors," *Nature Genetics*, 2017.

[39] D. Sengupta, N. A. Rayan, M. Lim, B. Lim, and S. Prabhakar, "Fast, scalable and accurate differential expression analysis for single cells," *bioRxiv*, p. 049734, 2016.

[40] L. Zhang and S. Zhang, "Comparison of computational methods for imputing single-cell rna-sequencing data," *IEEE/ACM transactions on computational biology and bioinformatics*, 2018.

[41] W. V. Li and J. J. Li, "scimpute: accurate and robust imputation for single cell rna-seq data," *bioRxiv*, p. 141598, 2017.

[42] I.-Y. Kwak, W. Gong, N. Koyano-Nakagawa, and D. Garry, "Drimpute: Imputing dropout events in single cell rna sequencing data," *bioRxiv*, p. 181479, 2017.

[43] C. Arisdakessian, O. Poirion, B. Yunits, X. Zhu, and L. Garmire, "Deepimpute: an accurate, fast and scalable deep neural network method to impute single-cell rna-seq data," *bioRxiv*, p. 353607, 2018.

[44] M. Huang, J. Wang, E. Torre, H. Dueck, S. Shaffer, R. Bonasio, J. I. Murray, A. Raj, M. Li, and N. R. Zhang, "Saver: gene expression recovery for single-cell rna sequencing," *Nature Methods*, vol. 15, no. 7, p. 539, 2018.

[45] G. X. Zheng, J. M. Terry, P. Belgrader, P. Ryvkin, Z. W. Bent, R. Wilson, S. B. Ziraldo, T. D. Wheeler, G. P. McDermott, J. Zhu *et al.*, "Massively parallel digital transcriptional profiling of single cells," *Nature communications*, vol. 8, p. 14049, 2017.

[46] P. Blakeley, N. M. Fogarty, I. Del Valle, S. E. Wamaitha, T. X. Hu, K. Elder, P. Snell, L. Christie, P. Robson, and K. K. Niakan, "Defining the three cell lineages of the human blastocyst by single-cell rna-seq," *Development*, vol. 142, no. 18, pp. 3151–3165, 2015.

[47] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2017.

[48] L. Wan and F. Sun, "Ceder: Accurate detection of differentially expressed genes by combining significance of exons using rna-seq," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 5, pp. 1281–1292, Sept 2012.

[49] P.-Y. Tung, J. D. Blischak, C. J. Hsiao, D. A. Knowles, J. E. Burnett, J. K. Pritchard, and Y. Gilad, "Batch effects and the effective design of single-cell gene expression studies," *Scientific reports*, vol. 7, p. 39921, 2017.

[50] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, "limma powers differential expression analyses for rna-

sequencing and microarray studies," *Nucleic acids research*, vol. 43, no. 7, pp. e47–e47, 2015.

[51] C. W. Law, Y. Chen, W. Shi, and G. K. Smyth, "voom: Precision weights unlock linear model analysis tools for rna-seq read counts," *Genome biology*, vol. 15, no. 2, p. R29, 2014.

[52] X. Zhou, H. Lindsay, and M. D. Robinson, "Robustly detecting differential expression in rna sequencing data using observation weights," *Nucleic acids research*, vol. 42, no. 11, pp. e91–e91, 2014.

[53] M. I. Love, W. Huber, and S. Anders, "Moderated estimation of fold change and dispersion for rna-seq data with deseq2," *Genome biology*, vol. 15, no. 12, p. 550, 2014.

[54] Y.-F. Dai and X.-M. Zhao, "A survey on the computational approaches to identify drug targets in the postgenomic era," *BioMed research international*, vol. 2015, 2015.

[55] A. Ezzat, M. Wu, X.-L. Li, and C.-K. Kwoh, "Computational prediction of drug-target interactions using chemogenomic approaches: an empirical survey," *Briefings in bioinformatics*, vol. 20, no. 4, pp. 1337–1357, 2019.

[56] A. Gaulton, L. J. Bellis, A. P. Bento, J. Chambers, M. Davies, A. Hersey, Y. Light, S. McGlinchey, D. Michalovich, B. Al-Lazikani *et al.*, "Chembl: a large-scale bioactivity database for drug discovery," *Nucleic acids research*, vol. 40, no. D1, pp. D1100–D1107, 2011.

[57] D. S. Wishart, C. Knox, A. C. Guo, D. Cheng, S. Shrivastava, D. Tzur, B. Gautam, and M. Hassanali, "Drugbank: a knowledgebase for drugs, drug actions and drug targets," *Nucleic acids research*, vol. 36, no. suppl_-1, pp. D901–D906, 2007.

[58] M. Kanehisa, S. Goto, Y. Sato, M. Furumichi, and M. Tanabe, "Kegg for integration and interpretation of large-scale molecular data sets," *Nucleic acids research*, vol. 40, no. D1, pp. D109–D114, 2011.

[59] M. Kuhn, D. Szklarczyk, S. Pletscher-Frankild, T. H. Blicher, C. Von Mering, L. J. Jensen, and P. Bork, "Stitch 4: integration of protein–chemical interactions with user data," *Nucleic acids research*, vol. 42, no. D1, pp. D401–D407, 2013.

[60] S. Günther, M. Kuhn, M. Dunkel, M. Campillos, C. Senger, E. Petsalaki, J. Ahmed, E. G. Urdiales, A. Gewiess, L. J. Jensen *et al.*, "Supertarget and matador: resources for exploring drug-target relationships," *Nucleic acids research*, vol. 36, no. suppl_1, pp. D919–D922, 2007.

[61] A. Masoudi-Nejad, Z. Mousavian, and J. H. Bozorgmehr, "Drug-target and disease networks: polypharmacology in the post-genomic era," *In silico pharmacology*, vol. 1, no. 1, p. 17, 2013.

[62] T. T. Ashburn and K. B. Thor, "Drug repositioning: identifying and developing new uses for existing drugs," *Nature reviews Drug discovery*, vol. 3, no. 8, p. 673, 2004.

[63] F. Cheng, C. Liu, J. Jiang, W. Lu, W. Li, G. Liu, W. Zhou, J. Huang, and Y. Tang, "Prediction of drug-target interactions and drug repositioning via network-based inference," *PLoS computational biology*, vol. 8, no. 5, p. e1002503, 2012.

[64] S. R. McLean, M. Gana-Weisz, B. Hartzoulakis, R. Frow, J. Whelan, D. Selwood, and C. Boshoff, "Imatinib binding and ckit inhibition is abrogated by the ckit kinase domain i missense mutation val654ala," *Molecular cancer therapeutics*, vol. 4, no. 12, pp. 2008–2015, 2005.

[65] S. Frantz, "Drug discovery: playing dirty," 2005.

[66] M. J. Keiser, B. L. Roth, B. N. Armbruster, P. Ernsberger, J. J. Irwin, and B. K. Shoichet, "Relating protein pharmacology by ligand chemistry," *Nature biotechnology*, vol. 25, no. 2, p. 197, 2007.

[67] A. Johnson and M. M. Wiley-Interscience, "Concepts and applications of molecular similarity. edited," 1991.

[68] L. Xie, T. Evangelidis, L. Xie, and P. E. Bourne, "Drug discovery using chemical systems biology: weak inhibition of multiple kinases may contribute to the anti-cancer effect of nelfinavir," *PLoS computational biology*, vol. 7, no. 4, p. e1002037, 2011.

[69] H. Li, Z. Gao, L. Kang, H. Zhang, K. Yang, K. Yu, X. Luo, W. Zhu, K. Chen, J. Shen *et al.*, "Tarfisdock: a web server for identifying drug targets with docking approach," *Nucleic acids research*, vol. 34, no. suppl_2, pp. W219–W224, 2006.

[70] G. Pujadas, M. Vaque, A. Ardevol, C. Blade, M. Salvado, M. Blay, J. Fernandez-Larrea, and L. Arola, "Protein-ligand docking: A review of recent advances and future perspectives," *Current Pharmaceutical Analysis*, vol. 4, no. 1, pp. 1–19, 2008.

[71] A. C. Cheng, R. G. Coleman, K. T. Smyth, Q. Cao, P. Soulard, D. R. Caffrey, A. C. Salzberg, and E. S. Huang, "Structure-based maximal affinity model predicts small-molecule druggability," *Nature biotechnology*, vol. 25, no. 1, p. 71, 2007.

[72] N. Nagamine, T. Shirakawa, Y. Minato, K. Torii, H. Kobayashi, M. Imoto, and Y. Sakakibara, "Integrating statistical predictions and experimental verifications for enhancing protein-chemical interaction predictions in virtual screening," *PLoS computational biology*, vol. 5, no. 6, p. e1000397, 2009.

[73] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[74] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2053–2080, 2010.

[75] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, p. 717, 2009.

[76] B. Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.

[77] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, "Prediction of drug–target interaction networks from the integration of chemical and genomic spaces," *Bioinformatics*, vol. 24, no. 13, pp. i232–i240, 2008.

[78] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, and M. Hirakawa, "From genomics to chemical genomics: new developments in kegg," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D354–D357, 2006.

[79] I. Schomburg, A. Chang, C. Ebeling, M. Gremse, C. Heldt, G. Huhn, and D. Schomburg, "Brenda, the enzyme database: updates and major new developments," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D431–D433, 2004.

[80] M. Hattori, N. Tanaka, M. Kanehisa, and S. Goto, "Simcomp/subcomp: chemical structure search servers for network analyses," *Nucleic acids research*, vol. 38, no. suppl_2, pp. W652–W656, 2010.

[81] J. AMoZ, "Identification of common molecular subsequences."

[82] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of machine learning research*, vol. 7, no. Nov, pp. 2399–2434, 2006.

[83] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[84] R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. I. Jordan, "A general analysis of the convergence of admm," *arXiv preprint arXiv:1502.02009*, 2015.

[85] S. Boyd, "Alternating direction method of multipliers," in *Talk at NIPS workshop on optimization and machine learning*, 2011.

[86] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu, "Collaborative matrix factorization with multiple similarities for predicting drug-target interactions," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 1025–1033.

[87] T. van Laarhoven and E. Marchiori, "Predicting drug-target interactions for new drug compounds using a weighted nearest neighbor profile," *PloS one*, vol. 8, no. 6, p. e66952, 2013.

[88] A. Majumdar and R. K. Ward, "Some empirical advances in matrix completion," *Signal Processing*, vol. 91, no. 5, pp. 1334–1338, 2011.

[89] V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Matrix completion on graphs," *arXiv preprint arXiv:1408.1717*, 2014.

[90] J. Burez and D. Van den Poel, "Handling class imbalance in customer churn prediction," *Expert Systems with Applications*, vol. 36, no. 3, pp. 4626–4636, 2009.

[91] I. Albert and R. Albert, "Conserved network motifs allow protein–protein interaction prediction," *Bioinformatics*, vol. 20, no. 18, pp. 3346–3352, 2004. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bth402

[92] C. Alkan, E. Karakoc, J. H. Nadeau, S. C. Sahinalp, and K. Zhang, "Rna–rna interaction prediction and antisense rna target search," *Journal of Computational Biology*, vol. 13, no. 2, pp. 267–282, 2006.

[93] J. Lamb, E. D. Crawford, D. Peck, J. W. Modell, I. C. Blat, M. J. Wrobel, J. Lerner, J.-P. Brunet, A. Subramanian, K. N. Ross *et al.*, "The connectivity map: using gene-expression signatures to connect small molecules, genes, and disease," *science*, vol. 313, no. 5795, pp. 1929–1935, 2006.

[94] G. Hu and P. Agarwal, "Human disease-drug network based on genomic expression profiles," *PloS one*, vol. 4, no. 8, 2009.

[95] E. Jadamba and M. Shin, "A systematic framework for drug repositioning from integrated omics and drug phenotype profiles using pathway-drug network," *BioMed research international*, vol. 2016, 2016.

[96] N. Pustelnik, C. Chaux, and J.-C. Pesquet, "Parallel proximal algorithm for image restoration using hybrid regularization," *IEEE transactions on Image Processing*, vol. 20, no. 9, pp. 2450–2462, 2011.

[97] A. Gottlieb, G. Y. Stein, E. Ruppin, and R. Sharan, "Predict: a method for inferring novel drug indications with application to personalized medicine," *Molecular systems biology*, vol. 7, no. 1, 2011.

[98] H. Luo, J. Wang, M. Li, J. Luo, X. Peng, F.-X. Wu, and Y. Pan, "Drug repositioning based on comprehensive similarity measures and bi-random walk algorithm," *Bioinformatics*, vol. 32, no. 17, pp. 2664–2671, 2016.

[99] D. S. Wishart, C. Knox, A. C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey, "Drugbank: a comprehensive resource for in silico drug discovery and exploration," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D668–D672, 2006.

[100] A. Hamosh, A. F. Scott, J. Amberger, C. Bocchini, D. Valle, and V. A. McKusick, "Online mendelian inheritance in man (omim), a knowledge-base of human genes and genetic disorders," *Nucleic acids research*, vol. 30, no. 1, pp. 52–55, 2002.

[101] T. T. Tanimoto, "An elementary mathematical theory of classification and prediction. 1958," *International Business Machines Corporation*, 1958.

[102] C. Steinbeck, Y. Han, S. Kuhn, O. Horlacher, E. Luttmann, and E. Willighagen, "The chemistry development kit (cdk): An open-source java library for chemo-and bioinformatics," *Journal of chemical information and computer sciences*, vol. 43, no. 2, pp. 493–500, 2003.

[103] M. A. Van Driel, J. Bruggeman, G. Vriend, H. G. Brunner, and J. A. Leunissen, "A text-mining analysis of the human phenome," *European journal of human genetics*, vol. 14, no. 5, pp. 535–542, 2006.

[104] A. Cherni, E. Chouzenoux, and M.-A. Delsuc, "Palma, an improved algorithm for dosy signal processing," *Analyst*, vol. 142, no. 5, pp. 772–779, 2017.

[105] M. Yang, H. Luo, Y. Li, and J. Wang, "Drug repositioning based on bounded nuclear norm regularization," *Bioinformatics*, vol. 35, no. 14, pp. i455–i463, 2019.

[106] Y. Wang, G. Deng, N. Zeng, X. Song, and Y. Zhuang, "Drug-disease association prediction based on neighborhood information aggregation in neural networks," *IEEE Access*, vol. 7, pp. 50 581–50 587, 2019.

[107] H. Luo, M. Li, S. Wang, Q. Liu, Y. Li, and J. Wang, "Computational drug repositioning using low-rank matrix approximation and randomized algorithms," *Bioinformatics*, vol. 34, no. 11, pp. 1904–1912, 2018.

[108] A. P. Davis, C. G. Murphy, R. Johnson, J. M. Lay, K. Lennon-Hopkins, C. Saraceni-Richards, D. Sciaky, B. L. King, M. C. Rosenstein, T. C. Wiegers *et al.*, "The comparative toxicogenomics database: update 2013," *Nucleic acids research*, vol. 41, no. D1, pp. D1104–D1114, 2013.

[109] S. Chatterjee, "An overview of mutations occurring within the coronavirus-2 genome: Mutations data reporting on sars-cov-2," *Available at SSRN 3632241*, 2020.

[110] *Coronavirus Update (Live)- Worldometer*, 2019 (accessed June 22, 2020), https://www.worldometers.info/coronavirus/.

[111] C. Harrison, "Coronavirus puts drug repurposing on the fast track." *Nature Biotechnology*, vol. 38, no. 4, pp. 379–381, 2020.

[112] S. He, B. Lin, V. Chu, Z. Hu, X. Hu, J. Xiao, A. Q. Wang, C. J. Schweitzer, Q. Li, M. Imamura *et al.*, "Repurposing of the antihistamine chlorcyclizine and related compounds for treatment of hepatitis c virus infection," *Science translational medicine*, vol. 7, no. 282, pp. 282ra49–282ra49, 2015.

[113] R. R. Razonable, "Antiviral drugs for viruses other than human immunodeficiency virus," in *Mayo Clinic Proceedings*, vol. 86, no. 10. Elsevier, 2011, pp. 1009–1026.

[114] E. De Clercq and G. Li, "Approved antiviral drugs over the past 50 years," *Clinical Microbiology Reviews*, vol. 29, no. 3, pp. 695–747, 2016.

[115] N. Sugaya and Y. Ohashi, "Long-acting neuraminidase inhibitor laninamivir octanoate (cs-8958) versus oseltamivir as treatment for children with influenza virus infection," *Antimicrobial Agents and Chemotherapy*, vol. 54, no. 6, pp. 2575–2582, 2010.

[116] A. Chopra, M. Saluja, and A. Venugopalan, "Effectiveness of chloroquine and inflammatory cytokine response in patients with early persistent musculoskeletal pain and arthritis following chikungunya virus infection," *Arthritis & Rhumatology*, vol. 66, no. 2, pp. 319–326, 2014.

[117] K. M. Gallegos, G. L. Drusano, D. Z. D Argenio, and A. N. Brown, "Chikungunya virus: in vitro response to combination therapy with ribavirin and interferon alfa 2a," *The Journal of Infectious Diseases*, vol. 214, no. 8, pp. 1192–1197, 2016.

[118] I. Das, I. Basantray, P. Mamidi, T. K. Nayak, B. Pratheek, S. Chattopadhyay, and S. Chattopadhyay, "Heat shock protein 90 positively regulates chikungunya virus replication by stabilizing viral non-structural protein nsp2 during infection," *PLoS One*, vol. 9, no. 6, p. e100531, 2014.

[119] S. A. Shiryaev, P. Mesci, A. Pinto, I. Fernandes, N. Sheets, S. Shresta, C. Farhy, C.-T. Huang, A. Y. Strongin, A. R. Muotri *et al.*, "Repurposing of the anti-malaria drug chloroquine for zika virus treatment and prophylaxis," *Scientific Reports*, vol. 7, no. 1, pp. 1–9, 2017.

[120] B. Winther and N. Mygind, "Potential benefits of ibuprofen in the treatment of viral respiratory infections," *Inflammopharmacology*, vol. 11, no. 4, p. 445, 2003.

[121] Z. Jin, Y. Zhao, Y. Sun, B. Zhang, H. Wang, Y. Wu, Y. Zhu, C. Zhu, T. Hu, X. Du *et al.*, "Structural basis for the inhibition of covid-19 virus main protease by carmofur, an antineoplastic drug," *bioRxiv*, 2020, https://www.biorxiv.org/content/10.1101/2020.04.09.033233v1.abstract.

[122] B. E. Pickett, E. L. Sadat, Y. Zhang, J. M. Noronha, R. B. Squires, V. Hunt, M. Liu, S. Kumar, S. Zaremba, Z. Gu *et al.*, "Vipr: an open

bioinformatics database and analysis resource for virology research," *Nucleic Acids Research*, vol. 40, no. D1, pp. D593–D598, 2012.

[123] *NCBI*, https://www.ncbi.nlm.nih.gov/.

[124] N. A. Ahlgren, J. Ren, Y. Y. Lu, J. A. Fuhrman, and F. Sun, "Alignment-free oligonucleotide frequency dissimilarity measure improves prediction of hosts from metagenomically-derived viral sequences," *Nucleic Acids Research*, vol. 45, no. 1, pp. 39–53, 2017.

[125] E. Chouzenoux, A. Jezierska, J. Pesquet, and H. Talbot, "A majorize-minimize subspace approach for l2-l0 image regularization," *SIAM Journal on Imaging Science*, vol. 6, no. 1, pp. 563–591, 2013.

[126] A. Mongia, D. Sengupta, and A. Majumdar, "Mcimpute: Matrix completion based imputation for single cell rna-seq data," *Frontiers in Genetics*, vol. 10, p. 9, 2019.

[127] N. Komodakis and J. Pesquet, "Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 31–54, 2015.

[128] A. Mongia and A. Majumdar, "Deep matrix completion on graphs: Application in drug target interaction prediction," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*. IEEE, 2020, pp. 1324–1328.

[129] A. Zumla, J. F. Chan, E. I. Azhar, D. S. Hui, and K.-Y. Yuen, "Coronaviruses—drug discovery and therapeutic options," *Nature Reviews Drug Discovery*, vol. 15, no. 5, pp. 327–347, 2016.

[130] J. H. Beigel, K. M. Tomashek, L. E. Dodd, A. K. Mehta, B. S. Zingman, A. C. Kalil, E. Hohmann, H. Y. Chu, A. Luetkemeyer, S. Kline *et al.*, "Remdesivir for the treatment of covid-19—preliminary report," *New England Journal of Medicine*, 2020.

[131] I. F.-N. Hung, K.-C. Lung, E. Y.-K. Tso, R. Liu, T. W.-H. Chung, M.-Y. Chu, Y.-Y. Ng, J. Lo, J. Chan, A. R. Tam *et al.*, "Triple combination of interferon beta-1b, lopinavir–ritonavir, and ribavirin in the treatment of patients admitted to hospital with covid-19: an open-label, randomised, phase 2 trial," *The Lancet*, vol. 395, no. 10238, pp. 1695–1704, 2020.

[132] Y.-M. Zeng, X.-L. Xu, X.-Q. He, S.-Q. Tang, Y. Li, Y.-Q. Huang, V. Harypursat, and Y.-K. Chen, "Comparative effectiveness and safety of ribavirin plus interferon-alpha, lopinavir/ritonavir plus interferon-alpha, and ribavirin plus lopinavir/ritonavir plus interferon-alpha in patients with mild to moderate novel coronavirus disease 2019: study protocol," *Chinese Medical Journal*, vol. 133, no. 9, pp. 1132–1134, 2020.

[133] J. Rodrigo, C.-B. J. Alberto, P. de León Samuel, A. Becerra, and A. Lazcano, "Sofosbuvir as a potential alternative to treat the sars-cov-2 epidemic," *Scientific Reports (Nature Publisher Group)*, vol. 10, no. 1, 2020.

[134] Z. Wang, X. Chen, Y. Lu, F. Chen, and W. Zhang, "Clinical characteristics and therapeutic procedure for four cases with 2019 novel coronavirus pneumonia receiving combined chinese and western medicine treatment," *Bioscience Trends*, vol. 14, no. 1, pp. 64–68, 2020.

[135] Y. Duan, H.-L. Zhu, and C. Zhou, "Advance of promising targets and agents against 2019-ncov in china," *Drug Discovery Today*, 2020.

[136] *New trial starts in UK to see if ibuprofen can help prevent severe breathing problems in Covid-19 patients*, https://www.thejournal.ie/ibuprofen-trial-coronavirus-5113390-Jun2020/.

[137] P. R. Martins-Filho, E. M. do Nascimento-Júnior, and V. Santana Santos, "No current evidence supporting risk of using ibuprofen in patients with covid-19," *International Journal of Clinical Practice*, p. e13576, 2020.

[138] Y. Lou, L. Liu, and Y. Qiu, "Clinical outcomes and plasma concentrations of baloxavir marboxil and favipiravir in covid-19 patients: an exploratory randomized, controlled trial," *medRxiv*, 2020.

[139] J. W. Huggins, "Prospects for treatment of viral hemorrhagic fevers with ribavirin, a broad-spectrum antiviral drug," *Reviews of Infectious Diseases*, vol. 11, no. Supplement_4, pp. S750–S761, 1989.

[140] E. A. Schaefer and R. T. Chung, "Anti- hepatitis c virus drugs in development," *Gastroenterology*, vol. 142, no. 6, pp. 1340–1350, 2012.

[141] M. Stoeckius, S. Zheng, B. Houck-Loomis, S. Hao, B. Z. Yeung, W. M. Mauck, P. Smibert, and R. Satija, "Cell hashing with barcoded antibod-

ies enables multiplexing and doublet detection for single cell genomics," *Genome biology*, vol. 19, no. 1, pp. 1–12, 2018.

[142] V. Kalofolias, "How to learn a graph from smooth signals," in *Artificial Intelligence and Statistics*, 2016, pp. 920–929.

[143] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

Appendices

.1 Plot of singular value decay

We plot of decay of singular values (against no of singular values) for one of the datasets for each of the problems in Introduction (Chapter 1) to support the low-rank assumption made in the dissertation for emploing the matrix completion tchniques.

The plots shown here (Figure 1) correspond to "Preimplanation" dataset for scRA-seq imputation problem, "E" (Enzymes) dataset for the drug-target interaction problem, "Catasets" for the drug-disease association problem. For the rest of the two chapters the only dataset used for each of the applications has been used.

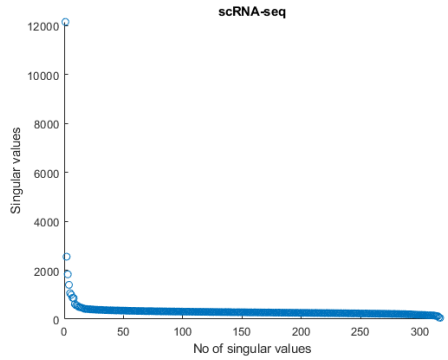
The plot shows that the singular values decreases rapidly (approximating exponential decay), so all but the k (rank of the matrix) largest singular values contain very little information anyway. This supports the low-rank assumption made on the biological datasets.

.2 Majorization minimization

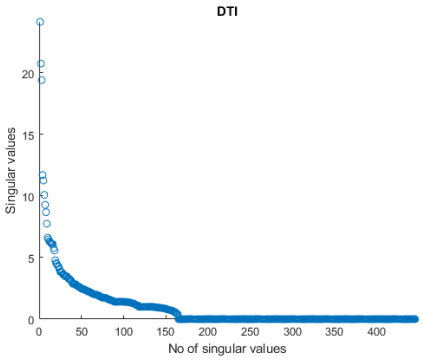
Majorization minimization (MM) is a concept from optimization theory where the goal is to replace a difficult minimization problem with a sequence of easier minimization problems by adding a majorizer term (which is easy to minimize) with some constraints to get a new function (to majorize the original cost function) and then minimizing this new function. This leads to a landweber update,

iterative application of which solves the cost function.

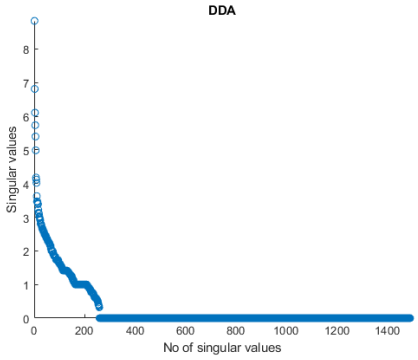
Figure 2 gives an idea of the majorization minimization algorithm. Let, $J(x)$ be the function to be minimized. Start with an initial point (at $k = 0$) x_k (**sub figure (a)**). A smooth function $G_k(x)$ is constructed through x_k which has a higher value than $J(x)$ for all values of x apart from x_k, at which the values are the same. This is the Majorization step. The function $G_k(x)$ is constructed such that it is smooth and easy to minimize. At each step, minimize $G_k(x)$ to obtain the next iterate x_{k+1} (**sub figure (b)**). A new $G_{k+1}(x)$ is constructed through x_{k+1} which is now minimized to obtain the next iterate x_{k+2}. As can be seen, the solution at every iteration gets closer to the actual solution.



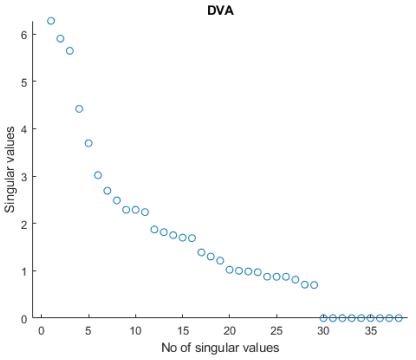
(a) Jurkat-293T



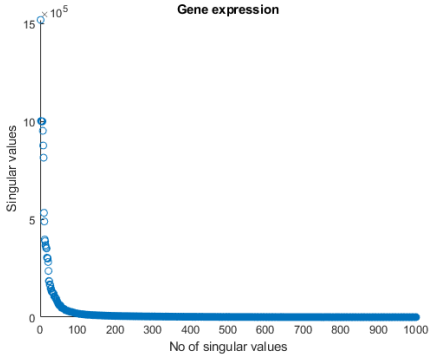
(b) PBMC



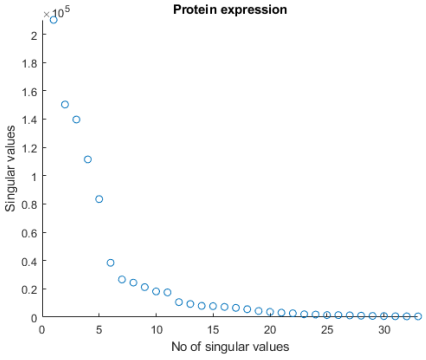
(a) Jurkat-293T



(b) PBMC

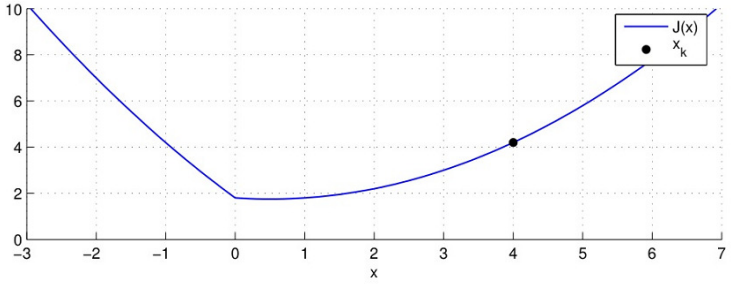


(a) Jurkat-293T

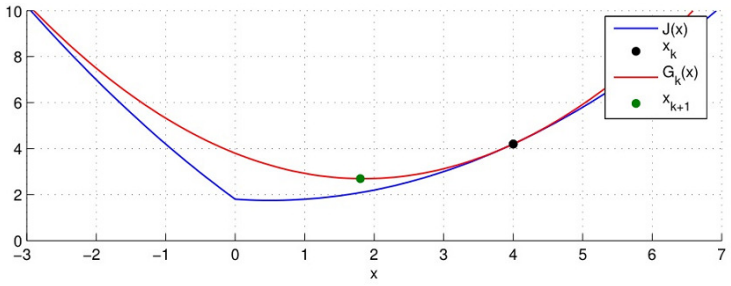


(b) PBMC

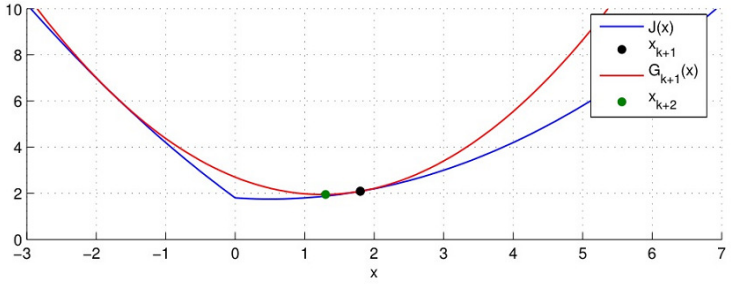
Figure 1: Singular value decay plot of each of the dataset taken from (a) scRNA-seq imputation (b) DTI prediction (c) DDA prediction (d) DVA prediction (e) gene expression matrix and (f) protein expression matrix in transcriptomic-proteomic prediction problem



(a) Function J(x) to be minimized



(b) One iteration of MM



(c) Subsequent iteration

Figure 2: Majorization Minimization - Schematic Diagram: