

Robust Discrete Matrix Completion

Jin Huang and Feiping Nie and Heng Huang*

Computer Science and Engineering Department

University of Texas at Arlington

Arlington, TX, 76019

huangjinsuzhou@gmail.com, feipingnie@gmail.com, heng@uta.edu

Abstract

Most existing matrix completion methods seek the matrix global structure in the real number domain and produce predictions that are inappropriate for applications retaining discrete structure, where an additional step is required to post-process prediction results with either heuristic threshold parameters or complicated mappings. Such an ad-hoc process is inefficient and impractical. In this paper, we propose a novel robust discrete matrix completion algorithm that produces the prediction from the collection of user specified discrete values by introducing a new discrete constraint to the matrix completion model. Our method achieves a high prediction accuracy, very close to the most optimal value of competitive methods with threshold values tuning. We solve the difficult integer programming problem via incorporating augmented Lagrangian method in an elegant way, which greatly accelerates the converge process of our method and provides the asymptotic convergence in theory. The proposed discrete matrix completion model is applied to solve three real-world applications, and all empirical results demonstrate the effectiveness of our method.

Missing data occur in many applications for different reasons. Some questions in a survey might be left blank due to individual user's negligence or reluctance to answer, certain parts of a gene microarray could fail to yield measurements due to noise and manufacturing defects, clinical studies often contain time sequential missing medical observations after participants drop out. In this paper, we focus on the prediction of random missing values.

In data mining community, a wide range of data sets are naturally organized in matrix form. Clearly recovering an arbitrary matrix M from partial entries is not a well-posed problem, therefore certain assumptions have to be imposed to the underlying matrix. The most common one is the low-rank of the matrix, which has been applied to factor-based models (Rennie and Srebro 2005; Srebro and Jaakkola 2003; Salakhutdinov and Mnih 2008). Training a linear factor model essentially seeks a low-rank matrix X that approximates M . There are other methods seeking low-rank approximation, including SVD (Billsus and Pazzani 1998),

Bayesian PCA (BPCA) (Bishop 1999), trace norm minimization (Candes and Recht 2009), and Schatten- p norm minimization (Nie, Huang, and Ding 2012; Nie et al. 2012).

So far all the prediction methods mentioned above work well on the data with continuous attributes, but there are also many applications only retaining discrete values, such as binary images and document-term associations. However, imposing the discrete constraints on the prediction outcome in the objective function could often turn the optimization into a computationally expensive NP-hard integer programming problem (Bertsimas and Weismantel 2005). Solution methods for integer programming problems can be categorized as optimal and heuristic. An optimal algorithm is one which guarantees to find the optimal solution mathematically. We are often not that interested in such optimal solution for efficiency consideration, and therefore researchers often take the heuristic approaches. In order to convert the predictions into discrete values, these methods usually resort to the thresholding methods. Since the ground truth is generally unknown in real applications, such post-process step is often heuristic and difficult to get the optimal prediction accuracy.

For example, given n users in a social network, we use a graph to describe the relationships between users (nodes): if one user is on the friends lists of the other users, then there is an edge between corresponding two nodes; no such edge otherwise. If we create a matrix $M^{n \times n}$ and use rows and columns to index these users, $M_{ij} = 1$ indicates user i is a friend of user j , 0 otherwise. It is expected M has a lot of missing values since individual users can only rate a few users, as a result the prediction is necessary to infer the relationship status between users. Binary discrete values are preferred here, because they give users clear and explicit information on trust or not. In contrast, a thresholding conversion or a mapping is necessary to transform the continuous prediction results to corresponding discrete values. It is clear that an appropriate threshold value or mapping function plays a crucial role in producing accurate final value. However, due to the severe data sparsity and the highly skewed link distribution in the real social network, such conversion is generally heuristic and impractical. Therefore, conventional matrix completion methods that seek global structure of the matrix are not applicable to practical discrete prediction problems.

*Corresponding Author

Copyright © 2013, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

There are also a few algorithms exploring the data structure in the discrete domain, such as (Hofmann 2001; Breese, Heckerman, and Kadie 1998; Marlin 2003; Collins, Dasgupta, and Schapire 2001). However, many of these algorithms rely on a significant portion of relatively complete rows (columns) and train the models based on the available information. There are two subtle drawbacks for these algorithms: first, the severe data sparsity in real applications often compromises the data integrity for the training observations, which limits the performance of these algorithms. second, these algorithms train the models from the individual rows and lose the global matrix structure information.

In this paper, we propose a robust matrix completion method via trace norm, which produces discrete prediction values from a global structure prospective. Our main contributions include:

- First, we explicitly impose the discrete constraint to the matrix completion objective, such that the new objective produces discrete outputs directly. What is more, there is no restriction on the number of discrete values, therefore our method is applicable to general discrete data sets.
- Second, we introduce an ancillary variable to reformulate the our integer programming problem into a problem that can be iteratively optimized. We utilize the Augmented Lagrange Multiplier (ALM) framework in the optimization. This not only significantly accelerates the convergence of our method, but also guarantees the asymptotic convergence in theory.
- Last, the combination of trace norm and ℓ_1 norm makes our low-rank approximation method robust to outliers, this is especially important when available entries are scarce and noisy. Trace norm of a matrix has been used as a convex relaxation of the rank minimization problem in (Candes and Recht 2009; Cai, Candes, and Shen 2010). However, classic trace norm minimization methods such as singular value thresholding (SVT) use ℓ_2 norm as discrepancy measure, therefore they are prone to the outlier influence.

Robust Discrete Matrix Completion

In this section, we would first introduce the necessary notations used in this paper and recent matrix completion methods that use trace norm. We will discuss the drawbacks in existing methods. To overcome the limitations, we will propose the new objective function.

Matrix Completion via Trace Norm

We use $M \in \mathbb{R}^{m \times n}$ ($n > m$) to represent the data matrix which contains missing values and M_Ω to represent the subset of M whose entries are known. The standard matrix completion solves the following optimization problem:

$$\begin{aligned} \min_X \quad & \text{rank}(X) \\ \text{s.t.} \quad & X_{ij} = M_{ij}, \quad (i, j) \in \Omega, \end{aligned} \quad (1)$$

to seek a low-rank matrix X that approximates M . However, even if there is a unique solution X to Eq. (1), this optimization problem is NP-hard and all known algorithms that pro-

vide exact solution have exponential time complexity (Ji and Ye 2009; Chistov and Grigoriev 1984).

To overcome this, Cai et al. (Candes and Recht 2009; Cai, Candes, and Shen 2010) propose to use trace norm (nuclear norm) of X as a convex approximation to its rank. Some researchers (Fazel 2002) further relax the constraints and use trace norm as a regularity term:

$$\min_X \|X_\Omega - M_\Omega\|_F^2 + \gamma \|X\|_* \quad (2)$$

where $\|X\|_*$ denotes the trace norm of X , which is the sum of singular values of X . $\gamma > 0$ is the regularity parameter balancing the observations fit and the rank of matrix X . X has the global optimal solution, but X is in the continuous domain and the discrete solution X^* via thresholding is no longer the optimal solution of Eq. (2).

Motivation and Our New Objective Function

It is well known that matrix completion via trace norm minimization can be applied to Netflix problem¹, where the vendor provides recommendations based on individual user's preferences given his/her pre-existing ratings on a subset of entries in a database. This practical problem shares some similarities with the link prediction problem we presented in the introduction: both data sets contain large number of users and suffer from severe data sparsity, the tasks are both predicting missing values in the matrix. Therefore it is beneficial to include trace norm regularity term to seek the low-rank matrix. On the other hand, as we mentioned in the introduction, it is much easier to interpret the prediction results for some applications if the discrete attributes of the data are retained, but current trace norm optimization algorithms only work on the continuous domain and it is often neither efficient nor practical to tune the threshold value, given the existence of large number of missing values.

To solve this problem, we explicitly impose the discrete constraints on the prediction values and propose the following optimization problem:

$$\begin{aligned} \min_X \quad & \|X_\Omega - M_\Omega\|_1 + \lambda \|X\|_* \\ \text{s.t.} \quad & X_{ij} \in \mathcal{D} \end{aligned} \quad (3)$$

where $\mathcal{D} = \{c_1, \dots, c_K\}$. Note that there is no restriction on K , in other words, this method can be applied to data sets with arbitrary number of distinct discrete values. We use the ℓ_1 norm in the first term to reduce the potential outlier influence and make our objective function more robust. The discrete constraints turn the optimization problem to an integer programming optimization problem, which is difficult to solve. We will present a novel algorithm in next section.

Optimization and Algorithm

In this section, we will derive the algorithm to optimize the objective function in Eq. (3). There have been a few optimization algorithms for trace norm regularity, such as SVT (Cai, Candes, and Shen 2010), extended gradient algorithm (Ji and Ye 2009). However, clearly these algorithms are not applicable here since the domain of X is now discrete.

¹<http://www.cs.uic.edu/~liub/KDD-cup-2007>

In this paper, we propose to incorporate the Augmented Lagrangian Multiplier (ALM) method (Bertsekas 2003) in our framework. The main idea is to eliminate equality constraints and instead add a penalty term to the cost function that assigns a very high cost to the infeasible points. ALM differs from other penalty-based approaches by simultaneously estimating the optimal solution and Lagrange multipliers in an iterative manner. The main advantages of ALM over other generic algorithms are the fast and accurate performance, and independence of problem schemes (Dewester et al. 1990).

We first introduce an ancillary variable Z that will be used to approximate X and write Eq. (3) into the following one

$$\begin{aligned} \min_{Z=X, Z} & \|X_\Omega - M_\Omega\|_1 + \lambda \|Z\|_* \\ \text{s.t. } & X_{ij} \in \mathcal{D} \end{aligned} \quad (4)$$

Then we write it in the following one that is suitable for ALM framework:

$$\begin{aligned} \min_{X, Z} & \|X_\Omega - M_\Omega\|_1 + \lambda \|Z\|_* \\ & + Tr(\Sigma^T(X - Z)) + \frac{\mu}{2} \|X - Z\|_F^2 \\ \text{s.t. } & X_{ij} \in \mathcal{D} \end{aligned} \quad (5)$$

where Tr is the trace operation for matrix, Σ is the parameter to adjust the discrepancy between X and Z , and μ is the penalty control parameter.

Since the objective function in Eq. (5) involves X and Z , we use an alternative optimization scheme, optimizing one variable while fixing the other one. After the initialization, the following steps are repeated until convergence.

Step 1: When X is fixed, optimizing with respect to Z is reduced to the following equation after the trace norm term is absorbed.

$$\min_Z \lambda \|Z\|_* + \frac{\mu}{2} \left\| X - Z + \frac{\Sigma}{\mu} \right\|_F^2 \quad (6)$$

It is easy to recognize the above equation can be written as the standard form

$$\min_Z \frac{1}{2} \left\| Z - \left(X + \frac{1}{\mu} \Sigma \right) \right\|_F^2 + \frac{\lambda}{\mu} \|Z\|_* \quad (7)$$

According to the singular value thresholding solution provided by (Cai, Candes, and Shen 2010), the solution of Z is

$$Z = U(\Sigma_1 - \frac{\lambda}{\mu} I)_+ V^T, \quad (8)$$

where U and V come from the SVD decomposition $X + \frac{1}{\mu} \Sigma = U \Sigma_1 V^T$ and $+$ is the thresholding operation for the singular values.

Step 2: When Z is fixed, optimizing with respect to X is reduced to the following one

$$\begin{aligned} \min_{x_{ij}} & \|X_\Omega - M_\Omega\|_1 + \frac{\mu}{2} \left\| X - Z + \frac{1}{\mu} \Sigma \right\|_F^2 \\ \text{s.t. } & X_{ij} \in \mathcal{D} \end{aligned} \quad (9)$$

Here we solve X based on whether $(i, j) \in \Omega$ or not.

To solve X_Ω^c , the complement of X_Ω , it is easy to see Eq. (9) becomes

$$\begin{aligned} \min_{x_{ij}} & \left\| X - Z + \frac{1}{\mu} \Sigma \right\|_F^2 \\ \text{s.t. } & X_{ij} \in \mathcal{D}, \quad (i, j) \notin \Omega \end{aligned} \quad (10)$$

Then Eq. (10) can be solved in an element-wise way and we get the following solution since each entry is chosen from the list of discrete values

$$X_{ij} = \arg \min_{c_k \in \mathcal{D}} (c_k - Z_{ij} + \frac{1}{\mu} \Sigma_{ij})^2, \quad (i, j) \notin \Omega. \quad (11)$$

Solving X_Ω comes to the following optimization problem

$$\begin{aligned} \min_X & \|X_\Omega - M_\Omega\|_1 + \frac{\mu}{2} \left\| X - Z + \frac{1}{\mu} \Sigma \right\|_F^2 \\ \text{s.t. } & X_{ij} \in \mathcal{D}, \quad (i, j) \in \Omega \end{aligned} \quad (12)$$

It can be solved in a similar manner as X_Ω^c .

$$X_{ij} = \arg \min_{c_k \in \mathcal{D}} |c_k - M_{ij}| + \frac{\mu}{2} (c_k - Z_{ij} + \frac{1}{\mu} \Sigma_{ij})^2, \quad (i, j) \in \Omega \quad (13)$$

The complete solution for X is given by combining Eq. (11) and Eq. (13).

Step 3: Re-calculate $\Sigma = \Sigma + \mu(X - Z)$ to update the discrepancy between X and Z .

Step 4: Update $\mu = \rho \mu$ via a fixed coefficient $\rho > 1$, as number of iterations increase, μ grows exponentially.

The main idea of this optimization framework is to consider the trace norm regularity norm and the discrete constraints separately. In Eq. (4), we introduce Z to approximate X . This removes the trace norm term of X and the original difficult problem Eq. (3) can now be solved in an easy way.

Due to space constraint, we provide an intuitive justification about algorithm convergence here. It is easy to notice that $\mu \rightarrow \infty$ as the number of iterations increase, X and Z have to be equal in order to keep objective function in Eq. (5) finite. In other words, Z asymptotically converges to X . More through proof regarding ALM convergence can be found in (Bertsekas 2003).

The complete steps of our algorithms are summarized in Algorithm (1). It is easy to observe the computation cost for the algorithm is dominated by the thresholding operation in step 1. This step is generally of order $O(m^2 n)$. Here we use PROPACK package (Larsen 2005) that has been used in (Cai, Candes, and Shen 2010). Note that as μ grows exponentially, usually it takes only a few iterations to converge. Therefore our method is faster than the conventional SVD method for matrix completion. The convergence criteria is the relative change of the objective function value falls within 10^{-4} . The value of ρ has a significant impact on the convergence speed of our algorithm, larger ρ value would reduce the required steps for convergence but meanwhile compromise the accuracy of final objective function value.

Experiments on Solving Three Applications

In this section, we apply our Robust Discrete Matrix Completion (RDMC) method to solve three real-world applica-

Algorithm 1 Robust Discrete Matrix Completion

Input: available entries M_Ω , ALM parameters μ, Σ, ρ .
Initialize M, X and Z .
repeat
 Update Z with Eq. (8).
 Update X with Eq. (11) and Eq. (13).
 $\Sigma = \Sigma + \mu(X - Z)$.
 $\mu = \rho\mu$.
until Convergence
Output: X

tions: social network link prediction, missing SNP imputation, and protein-protein interactions prediction. To evaluate our method, we compare the results of our method to multiple related methods, include SVD, Bayesian PCA (BPCA) (Oba et al. 2003)², Singular Value Thresholding (SVT) (Cai, Candes, and Shen 2010)³, Bayesian Probabilistic Matrix Factorization (BPMF) (Salakhutdinov and Mnih 2008)⁴, Maximum Margin Matrix Factorization (MMMF) (Rennie and Srebro 2005)⁵, Aspect Model (AM) (Hofmann 2001)⁶ and User Rating Profile (URP) (Hofmann 2001).

Experiment Setup

For RDMC, MMMF, and SVT, we tune the regularity parameter from the list of $\{0.01, 0.1, 1, 10, 100\}$. For BPCA and BPMF, we use the default parameters of the program. For ALM framework parameters, we initialize empirically as follows: $\rho = 1.05$, $\mu = 0.1$ and Σ zero matrix. Note that ALM framework parameters only control the convergence speed, and different settings have little performance differences given sufficient number of iterations. We initialize all methods with the same missing values of X , random values between 0 and 1 unless otherwise specified. For SVD, since the possible choices of rank are dependent upon the data set size, we specify the exact choice when we come to each data set. We first randomly hide different portions of the ground truth for prediction on different data sets given their characteristics, then we evaluate their performances using different measure metrics and record the optimal values. The reported results are the average of 20 times such runs.

Link Prediction on Social Network Graph

In social network, it is interesting to predict the future friendship links based on existing ones. Because the friendship between users is either Yes or No, such a prediction is a discrete matrix completion problem. We conduct experiments on a binary social graph data set called Wikivote⁷. Wikipedia users hold elections to promote some users to administrators. Here we consider a vote between two users as a directed link. Wikivote contains about 7,000 users and 103,000 edges.

Data set		MAE(%)	RMSE
Wikivote	SVD	1.87 ± 0.09	0.86 ± 0.08
	BPCA	1.77 ± 0.03	0.83 ± 0.18
	BPMF	1.72 ± 0.05	0.82 ± 0.15
	SVT	1.78 ± 0.21	0.80 ± 0.03
	RDMC	1.56 ± 0.14	0.81 ± 0.04
	MMMF	1.62 ± 0.17	0.87 ± 0.05
	AM	1.82 ± 0.24	0.82 ± 0.05
	URP	1.68 ± 0.19	0.89 ± 0.07

Table 1: Prediction Result for Wikivote

When creating links, we label these direct edges 1 and 0 otherwise. It can be observed that the distributions of these links is very skewed due to the domination of 0s. To alleviate the data skewness for fair comparison and keep the computation manageable, we select top 2,000 highest degree users from Wikivote. We randomly hide 90% of the true value of links and make prediction based on 10% available entries. We choose 90% missing here to simulate the severe data sparsity real social graphs generally suffer from. The reason is that most users from the online community only explicitly express trust and distrust to a *very small* fraction of peer users considering total number of users.

For SVD method, we tune the choice of SVD rank from list $\{5, 10, 15, 20, 25\}$. We initialize each missing value of M using the mean of its row and column available entries, which takes account of the individual user's vote pattern and others' evaluations for him.

From Table 1, we can observe that RDMC has a significant lower MAE value, but not the case for RMSE. The reason for such inconsistency is that since we impose the discrete constraints, we get much more penalization for a wrong prediction in terms of RMSE for binary data. For example, given some entry ground truth value 1 and the prediction in the continuous domain is 0.4, if we predict it 0, we got 1 more in RMSE numerator, much more than 0.36 in the continuous domain.

Now we evaluate the prediction accuracy in the discrete domain. We plot the prediction error curves for various methods using different thresholding values in Fig. (1). Given threshold value θ , if predicted value from competitive methods is less than equal to θ , we predict it 0, otherwise 1. Since these links are very sparse considering the data set size, we tune the θ from 0.01 to 0.1. From the figure, we can see that RDMC has the prediction error very close to the most optimal value of competitive methods after tuning the threshold. Since the result of RDMC is always restricted to the discrete domain during the optimization, there is no need to tune the heuristic threshold value. Thus, RDMC a very useful tool when discrete nature of the data is desired to retain.

Missing Data Inference on SNP

Single nucleotide polymorphism (SNP) is important for identifying gene-disease associations. SNP data sets usually contain a significant number of missing genotypes ranging from 5 to 20% (de Brevern, Hazout, and Malper-

²<http://hawaii.sys.i.kyoto-u.ac.jp/oba/tools/BPCAFill.html>

³<http://svt.stanford.edu/>

⁴<http://www.mit.edu/~rsalakhu/BPMF.html>

⁵<http://people.csail.mit.edu/jrennie/matlab/>

⁶<http://www.cs.cmu.edu/~lebanon/IR-lab.htm>

⁷<http://snap.stanford.edu/data>

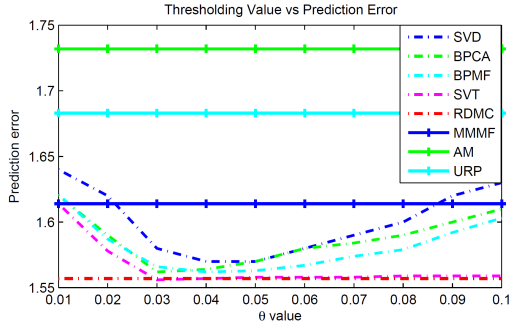


Figure 1: Plot for the Prediction Errors against Different Thresholding Values for All Methods on Wikivote.

tuy 2004). Here we use phased SNP dataset chromosomes 15 from NCBI⁸. It consists of four populations (CEU/YRI/JPT/HCB). A minor allele frequency, the lowest allele frequency at a locus that is observed in a particular population, is denoted by 1, and the majority alleles are denoted by 0. The ratios of major alleles and minor alleles from 4 populations are all roughly 2:1.

We randomly hide 20% entries for prediction purpose. We list the MAE and RMSE results on these 4 populations in Table 2. As the SNP matrix of CEU is of size 120×21138 , we tune the rank in the list of $\{5, 10, 15, 20\}$. From Table 2, we can observe the same trend as previous social graph data: our method achieves a significant lower value in MAE only.

Now we come to the accuracy evaluation. Since 1 and 0 are not that skewed distributed, we tune the threshold from 0.1 to 1. Here we include the accuracy curves plots for all four data sets in Fig. 2, because we find the optimal thresholds for competitive methods could be slightly different on these four SNP data sets. This observation justifies our motivation to maintain discrete structure as it is impossible to know the optimal threshold beforehand for practical application, meanwhile our method always maintains performance close to the optimal results.

Protein Interaction Prediction on Protein Network

In protein interaction network, the new protein-protein interactions prediction based on existing interactions is desired. This is a good discrete matrix completion application. We use the data set from BioGRID database (Stark et al. 2006). We evaluate all methods on the *Saccharomyces cerevisiae* genome, for which an undirected constructed, with vertices representing proteins and edges representing observed physical interactions (1 if two vertices have observed interaction and 0 otherwise). When constructing the graph, we only consider the vertices with no less than 100 degrees from BioGRID database of version 2.0.56, we denote such yielded binary graph G_1 , which is of size $1,654 \times 1,654$. More research into this genome revealed some interactions undiscovered in previous data set, we denote the corresponding vertices graph from version 3.1.69 as G_2 . Since the changes

Dataset		MAE(%)	RMSE
PIN	SVD	2.77 ± 0.12	0.89 ± 0.10
	BPCA	2.54 ± 0.04	0.85 ± 0.21
	BPMF	2.32 ± 0.08	0.85 ± 0.17
	SVT	2.24 ± 0.21	0.81 ± 0.06
	RDMC	1.98 ± 0.12	0.89 ± 0.04
	MMMF	2.12 ± 0.41	0.93 ± 0.07
	AM	2.34 ± 0.52	0.99 ± 0.10
	URP	2.21 ± 0.43	0.96 ± 0.08

Table 3: Protein-Protein Interaction Prediction Results

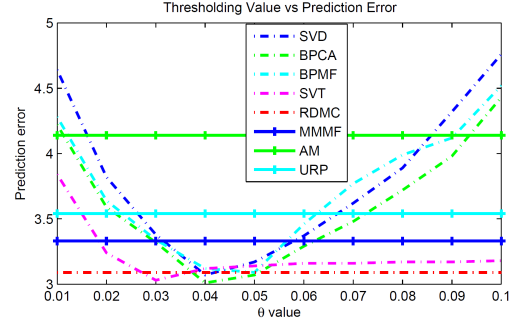


Figure 3: Plot for the Prediction Errors against Varying Thresholding Values for All Methods on Protein Interaction Network

from G_1 to G_2 would be only the new discovered links (more 1s in G_2 than G_1). To make the evaluation more comprehensive, we randomly hide 40% entries in G_1 and initialize these entries by random numbers between 0 and 1. Since this graph is undirected, we also mask $G_1(j, i)$ if $G_1(i, j)$ is missing. Now that there is no symmetry constraint imposed on all methods, we use the mean of predicted entries $X(i, j)$ and $X(j, i)$ to substitute original predicted values. Then we compare the link prediction results with G_2 .

We report the MAE and RMSE results in Table 3 and plot the accuracies of all methods in Fig. 3. Obviously our method can predict the new protein-protein interactions with quite high accuracy.

Conclusion

In this paper, we propose a robust discrete matrix completion (RDMC) prediction method that retains the discrete nature of the data sets. Different from conventional methods that tune the predictions using heuristic parameters, our method explicitly imposes the discrete constraints on the prediction and avoids the post-processing step. We solve the difficult integer programming problem via introducing an ancillary variable and decomposing the difficult problem into two manageable pieces. The asymptotic convergence of the ancillary variable to our prediction matrix justifies the soundness of our method.

Acknowledgements

This work was partially funded by NSF CCF-0830780, CCF-0917274, DMS-0915228, and IIS-1117965.

⁸<http://hapmap.ncbi.nlm.nih.gov/downloads/phasing/>

Dataset		MAE	RMSE	Dataset		MAE	RMSE
CEU	SVD	0.32 ± 0.02	0.91 ± 0.04	YRI	SVD	0.32 ± 0.03	0.95 ± 0.04
	BPCA	0.30 ± 0.02	0.83 ± 0.08		BPCA	0.31 ± 0.02	0.83 ± 0.07
	BPMF	0.31 ± 0.04	0.82 ± 0.06		BPMF	0.31 ± 0.03	0.84 ± 0.05
	SVT	0.30 ± 0.05	0.83 ± 0.04		SVT	0.31 ± 0.04	0.84 ± 0.03
	RDMC	0.23 ± 0.02	0.89 ± 0.08		RDMC	0.24 ± 0.02	0.90 ± 0.07
	MMMF	0.25 ± 0.02	0.91 ± 0.10		MMMF	0.25 ± 0.02	0.92 ± 0.11
	AM	0.32 ± 0.06	0.89 ± 0.07		AM	0.31 ± 0.05	0.88 ± 0.08
URP		0.25 ± 0.02	0.90 ± 0.07	URP		0.26 ± 0.03	0.90 ± 0.07

Dataset		MAE	RMSE	Dataset		MAE	RMSE
JPT	SVD	0.32 ± 0.02	0.91 ± 0.04	HCB	SVD	0.32 ± 0.02	0.91 ± 0.04
	BPCA	0.30 ± 0.02	0.83 ± 0.08		BPCA	0.30 ± 0.02	0.83 ± 0.08
	BPMF	0.31 ± 0.04	0.82 ± 0.06		BPMF	0.31 ± 0.04	0.84 ± 0.06
	SVT	0.30 ± 0.05	0.83 ± 0.04		SVT	0.30 ± 0.05	0.83 ± 0.04
	RDMC	0.23 ± 0.02	0.89 ± 0.08		RDMC	0.23 ± 0.02	0.89 ± 0.04
	MMMF	0.26 ± 0.03	0.93 ± 0.10		MMMF	0.25 ± 0.04	0.91 ± 0.11
	AM	0.32 ± 0.04	0.89 ± 0.09		AM	0.30 ± 0.05	0.90 ± 0.08
URP		0.25 ± 0.04	0.91 ± 0.07	URP		0.27 ± 0.05	0.91 ± 0.06

Table 2: Prediction Results on SNP Data

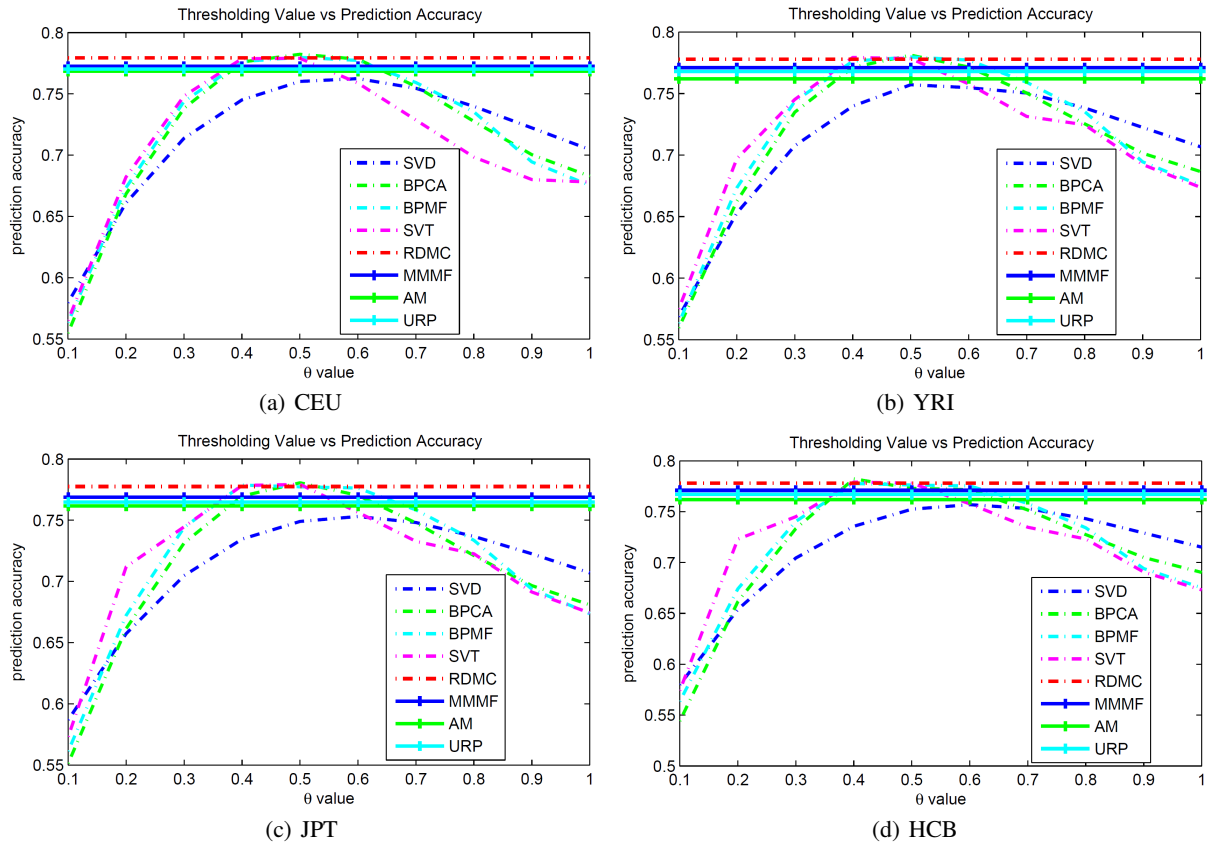


Figure 2: SNP Prediction Accuracy vs Threshold Values on 4 SNP Data Sets

References

- Bertsekas, D. P. 2003. *Nonlinear Programming*. Athena Scientific.
- Bertsimas, D. P., and Weismantel, R. 2005. *Optimization Over Integers*. Belmont, Massachusetts: Dynamic Ideas.
- Billsus, D., and Pazzani, M. J. 1998. Learning collaborative information filters. In *Proceedings of the Fifteenth International Conference on Machine Learning*, 46–54.
- Bishop, C. M. 1999. Bayesian pca. In *Proceedings of the 1998 conference on Advances in Neural Information Processing Systems*, 382–388.
- Breese, J. S.; Heckerman, D.; and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 43–52.
- Cai, J.; Candes, E. J.; and Shen, Z. 2010. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization* 20(4):1956–1982.
- Candes, E. J., and Recht, B. 2009. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics* 9(6):717–772.
- Chistov, A. L., and Grigoriev, D. Y. 1984. Complexity of quantifier elimination in the theory of algebraically closed fields. *Mathematical Foundations of Computer Science, volume 176 of Lecture Notes in Computer Science* 17–31.
- Collins, M.; Dasgupta, S.; and Schapire, R. E. 2001. A generalization of principal component analysis to the exponential family. In *Proceedings of Advances in Neural Information Processing Systems*, 617–624.
- de Brevern, A. G.; Hazout, S.; and Malpertuy, A. 2004. Influence of microarrays experiments missing values on the stability of gene groups by hierarchical clustering. *BMC Bioinformatics* 5(114).
- Dewester, S.; Dumains, S.; Landauer, T.; Furnas, G.; and Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the Society for Information Science* 41(6):391–407.
- Fazel, M. 2002. Matrix rank minimization with applications.
- Hofmann, T. 2001. Learning what people (don't) want. In *Proceedings of the 12th European Conference on Machine Learning*, 214–225.
- Ji, S., and Ye, J. 2009. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 457–464.
- Larsen, R. M. 2005. Propack software for large and sparse svd calculations. available online. url <http://sun.stanford.edu/rmunk/propack>.
- Marlin, B. 2003. Modeling user rating profiles for collaborative filtering. In *Proceedings of advances in Neural Information Processing Systems*, 627–634.
- Nie, F.; Wang, H.; Cai, X.; Huang, H.; and Ding, C. 2012. Robust matrix completion via joint Schatten p -norm and l_p -norm minimization. In *IEEE 12th International Conference on Data Mining (ICDM)*, 566–574.
- Nie, F.; Huang, H.; and Ding, C. 2012. Efficient Schatten- p norm minimization for low-rank matrix recovery. In *Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI)*, 655–661.
- Oba, S.; Sato, M.; Takemasa, I.; Monden, M.; Matsubara, K.; and Ishii, S. 2003. A bayesian missing value estimation method. *Bioinformatics* 19(16):2088–2096.
- Rennie, J. D., and Srebro, N. 2005. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, 713–719.
- Salakhutdinov, R., and Mnih, A. 2008. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th International Conference on Machine learning*, 880–887.
- Srebro, N., and Jaakkola, T. 2003. Weighted low-rank approximations. In *Proceeding of 20th International Conference on Machine Learning*, 720–727.
- Stark, C.; Breitkreutz, B. J.; Reguly, T.; Boucher, L.; Breitkreutz, A.; and Tyers, M. 2006. Biogrid: a general repository for interaction datasets. *Nucleic Acids Res.* 34(Database Issue).