UC Irvine
UC Irvine Electronic Theses and Dissertations

Title
Compact Factorization of Matrices Using Generalized Round-Rank

Permalink
https://escholarship.org/uc/item/9x58b95k

Author
Pezeshkpour, Pouya

Publication Date
2018

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,
IRVINE

Compact Factorization of Matrices Using Generalized Round-Rank

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Electrical Engineering

by

Pouya Pezeshkpour

Thesis Committee:
Professor Sameer Singh, Chair
Professor Syed A. Jafar
Professor Athina Markopoulou

2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Foremost, I would like to express my sincere gratitude to my advisor Prof. Sameer Singh for the continuous support of my research, for his patience, motivation, enthusiasm, and immense knowledge.

I would like to thank my family: my parents and my brothers for supporting me spiritually throughout my research studies and my life in general. At last but not least, I want to start this thesis with the following very beautiful poem:

گر نشان زندگی جنبندگی است   خار در صحرا سراسر زندگی است

هم جهل زنده است و هم پروانه لیک   فرق ها از زندگی تا زندگی است

ABSTRACT OF THE THESIS

Compact Factorization of Matrices Using Generalized Round-Rank

By

Pouya Pezeshkpour

Master of Science in Electrical Engineering

University of California, Irvine, 2018

Professor Sameer Singh, Chair

Matrix factorization is a popular machine learning technique, with applications in variety of domains, such as recommendation systems [16, 28], natural language processing [26], and computer vision [10]. Due to this widespread use of these models, there has been considerable theoretical analysis of the various properties of low-rank approximations of real-valued matrices, including approximation rank [1, 5] and sample complexity [2].

Rather than assume real-valued data, a number of studies (particularly ones on practical applications) focus on more specific data types, such as binary data [23], integer data [17], and ordinal data [12, 30]. For such matrices, existing approaches have used different *link* functions, applied in an element-wise manner to the low-rank representation [21], i.e. the output $\hat{Y}$ is $\psi(\mathbf{U}^T\mathbf{V})$ instead of the conventional $\mathbf{U}^T\mathbf{V}$. These link functions have been justified from a probabilistic point of view [4, 27], and have provided considerable success in empirical settings. However, theoretical results for linear factorization do not apply here, and thus the expressive power of the factorization models with non-linear link functions is not clear, and neither is the relation of the rank of a matrix to the link function used.

In this work, we first define a generalized notion of rank based on the link function ψ, as the rank of a latent matrix before the link function is applied. We focus on a link function that applies to factorization of integer-valued matrices: the generalized round function (GRF), and define the corresponding generalized round-rank (GRR). After providing background on GRR, we show that there are many low-GRR matrices that are full rank[1]. Moreover, we also study the approximation limitations of linear rank, by showing, for example, that low GRR matrices often cannot be approximated by low-rank linear matrices. We define uniqueness for GRR-based matrix completion, and derive its necessary and sufficient conditions. These properties demonstrate that many full linear-rank matrices can be factorized using low-rank matrices if an appropriate link function is used.

We also present an empirical evaluation of factorization with different link functions for matrix reconstruction and completion. We show that using link functions is efficient compared to linear rank, in that gradient-based optimization approach learns more accurate reconstructions using a lower rank representation and fewer training samples. We also perform experiments on matrix completion on two recommendation datasets, and demonstrate that appropriate link function outperform linear factorization, thus can play a crucial role in accurate matrix completion.

---

[1]We will refer to rank of a matrix as its *linear* rank, and refer to the introduced generalized rank as *link*-rank.

Chapter 1

Matrix Factorization and
Recommendation Systems

Matrix factorization is a commonly used method to represent data in a very compact form and simply defined as finding out two (or more) matrices such that when you multiply them together you will get back the original matrix. The intuition behind using matrix factorization to represent data in a compact form is that there should be some latent features that determine how rows and columns of the matrix related to each other. Accordingly, if we decompose a matrix $\mathbf{Y} \in \mathbb{R}^{n \times m}$ as:

$$\mathbf{Y} = \mathbf{U}\mathbf{V}^{T} \tag{1.1}$$

Where $\mathbf{U} \in \mathbb{R}^{n \times r}$ and $\mathbf{V} \in \mathbb{R}^{n \times r}$, we can represent all of the $n \times m$ entries of matrix $\mathbf{Y}$ with only $r \times (n + m)$ entries of matrices $\mathbf{U}$ and $\mathbf{V}$ which result in a very compact representation if we consider a small enough value for r. The only Achilles' heel of this approach is the fact that although choosing a smaller value for r will provide a

more compact representation, but will increase the error in the predictions as well. Accordingly, since real world matrices are very big and sparse representing them with a small r will produce a very erroneous prediction. As a result, providing a more compact representation, without increasing the error is of high importance. The goal of this work is to provide this very compact representation.

In this work, after studying the back ground of matrix factorization to provide a more compact representation, we define a generalized notion of rank based on the link function ψ, as the rank of a latent matrix before the link function is applied. We focus on a link function that applies to factorization of integer-valued matrices: the generalized round function (GRF), and define the corresponding generalized round-rank (GRR). Accordingly, the goal is instead of applying matrix factorization to the high rank original matrix, firstly map this matrix to a lower rank matrix (using GRF^{-1} function). Then, after applying matrix factorization on the new matrix, use GRF function to attain the original matrix. The idea of using GRR-based factorization is comming from the fact that we can show there are many low-GRR matrices that are full rank. Furthermore, to better understand the essence of our model, we define uniqueness for GRR-based matrix completion, and derive its necessary and sufficient conditions.

We also present an empirical evaluation of factorization for matrix reconstruction and completion. We show that using link functions is efficient compared to linear rank, in that gradient-based optimization approach learns more accurate reconstructions using a lower rank representation and fewer training samples. We also perform experiments on matrix completion on two recommendation datasets, and demonstrate that appropriate link function outperform linear factorization, thus can play a crucial role in accurate matrix completion.
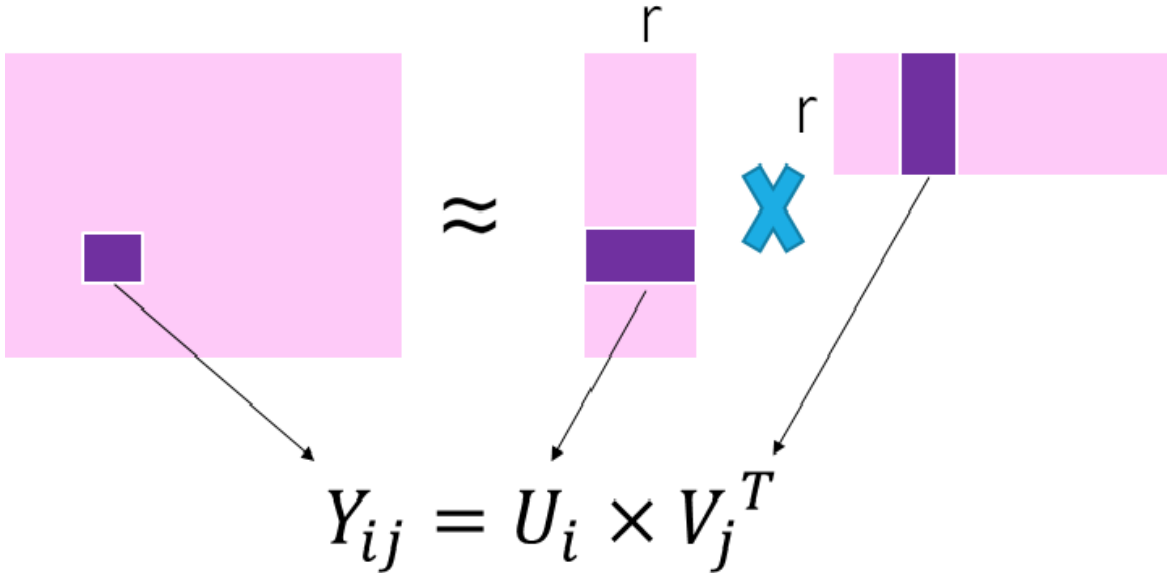
Figure 1.1: Representation of matrix factorization

1.1 Matrix Factorization: Linear Rank and Matrix Completion

Matrix factorization, broadly defined, is a decomposition of a matrix as a multiplication of two matrices. A simple representation of matrix factorization is depicted in figure 1.1. Accordingly, rank of a matrix $\mathbf{Y} \in \mathbb{R}^{n \times m}$ defined as the smallest natural number r such that:

$$\mathbf{Y} = \mathbf{U}\mathbf{V}^T, \text{or}, \mathbf{Y}_{ij} = \sum_k \mathbf{U}_{ik}\mathbf{V}_{jk} \tag{1.2}$$

where $\mathbf{U} \in \mathbb{R}^{n \times r}$ and $\mathbf{V} \in \mathbb{R}^{n \times r}$. We use r($\mathbf{Y}$) to indicate the rank of a matrix $\mathbf{Y}$. Accordingly, one of the most important applications of matrix factorization is matrix completion problem which is defined as follows.

				3.4
5.4	2		1	
		0	1	1.7
	2			
10.8				

➡️

10.8	4	0	2	3.4
5.4	2	0	1	1.7
5.4	2	0	1	1.7
5.4	2	0	1	1.7
10.8	4	0	2	3.4
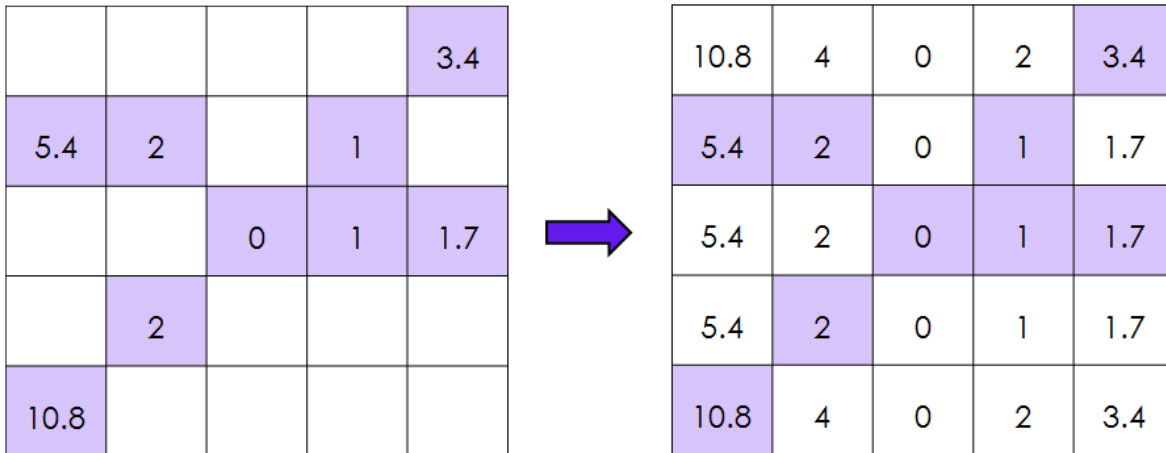
Figure 1.2: Example of matrix completion

1.1.1 Matrix Completion

Matrix completion is the task of filling in the missing entries of a partially observed matrix. Thus matrix completion often seeks to find the lowest rank matrix or, if the rank of the completed matrix is known, a matrix of rank r that matches the known entries. A simple example of matrix completion is presented in figure 1.2 with the assumption that the rank of the matrix is equal to one. Accordingly, to be able to complete the original matrix we consider a number of assumptions on the observed entries.

Uniform sampling of observed entries: To simplify the analysis of matrix completion problem, it is often assumed that the set of observed entries of the matrix is sampled using Bernoulli sampling, i.e. each entry being observed with the probability of p. Accordingly, by choosing $p = N/(nm)$ (where N is desired expected cardinality of observed entries) we can have a good approximation of uniform sampling.

Lower bound on number of observed entries: Along the same line, we can prove the following known information theoretic lower bound on the number of observed

entries so the matrix can be uniquely reconstructed.

Theorem 1.1.1. *For a given matrix* $\mathbf{Y} \in \mathbb{R}^{n \times m}$ *with rank* r *and and assuming that* $r \ll \{m, n\}$, *the lower bound on the number of observed entries to be able to uniquely reconstruct matrix* $\mathbf{Y}$ *is on the order of* $O(nr \log n)$.

Proof. We first need to find the number of degrees of freedom (the number of parameters of the matrix that may vary independently) of the matrix. To do so, let's fill the first r columns of the matrix with n degrees of freedom for each column, which result in linearly independent columns. We can now choose the remaining columns to be linear combinations of the first r columns. This gives r degrees of freedom for each column, namely the r coefficients of the linear combinations. As a result, the degrees of freedom will be equal to:

$$nr + r(m - r) \leq 2nr - r^2 \tag{1.3}$$

with assumption that $m \leq n$.

Secondly, there must be at least one observed entry per row and column of $\mathbf{Y}$. The Singular Value Decomposition of $\mathbf{Y}$ is given by $\mathbf{U\Sigma V}^T$. If row i does not have any observed entry, it is easy to see the i^{the} row of $\mathbf{V}$, V_i, can be changed to some arbitrary value and still yield a matrix matching Y over the set of observed entries. The same argument is applicable to any column of $\mathbf{Y}$ as well. Furthermore, if we assume Bernoulli sampling of the observed entries, the Coupon collector effect implies that entries on the order of $O(n \log n)$ must be observed to ensure that there is an observation from each row and column with high probability.

Combining these necessary conditions we can prove the theorem. $\qquad\square$

1.1.2 Gradient-Based Algorithm for Matrix Completion

Having discussed the intuition behind matrix factorization, we can now provide a gradient-based algorithm for matrix completion. Let's assume we want to complete a matrix $\mathbf{Y} \in \mathbb{R}^{n \times m}$ with a set of N observed entries E and rank r. Approximating $\mathbf{Y}$ as multiplication of two matrices $\mathbf{U} \in \mathbb{R}^{n \times r}$ and $\mathbf{V} \in \mathbb{R}^{m \times r}$ we will have:

$$\hat{Y}_{ij} = U_i \times \mathbf{V}_j^T = \sum_{k=1}^{r} U_{ik} V_{jk} \tag{1.4}$$

Now to approximate $\mathbf{Y}$ we just need to find $\mathbf{U}$ and $\mathbf{V}$. To do so, we first initialize the two matrices with some values, calculate how different their product is to $\mathbf{Y}$, and then try to minimize this difference iteratively using gradient descent method. Accordingly, The difference here can be calculated by the following equation:

$$e_{ij}^2 = (Y_{ij} - \hat{Y}ij)^2 = (Y_{ij} - \sum_{k=1}^{r} U_{ik} V_{jk})^2 \tag{1.5}$$

To minimize this error, we have to modify the values of U_{ik} and V_{kj} in a direction which has the most reduction. In other words, we need to know the gradient at the current values, and therefore we differentiate the above equation with respect to these two variables separately:

$$\frac{\sigma}{\sigma \mathbf{U}_{ik}} e_{ij}^2 = -2e_{ij} \mathbf{V}_{jk} \tag{1.6}$$

$$\frac{\sigma}{\sigma \mathbf{V}_{jk}} e_{ij}^2 = -2e_{ij} \mathbf{U}_{ik} \tag{1.7}$$

	Item 1	Item 2	Item 3	Item 4	Item 5
User 1	1			2	
User 2			3	5	
User 3		2	5		2
User 4	4			1	
User 5		3			4
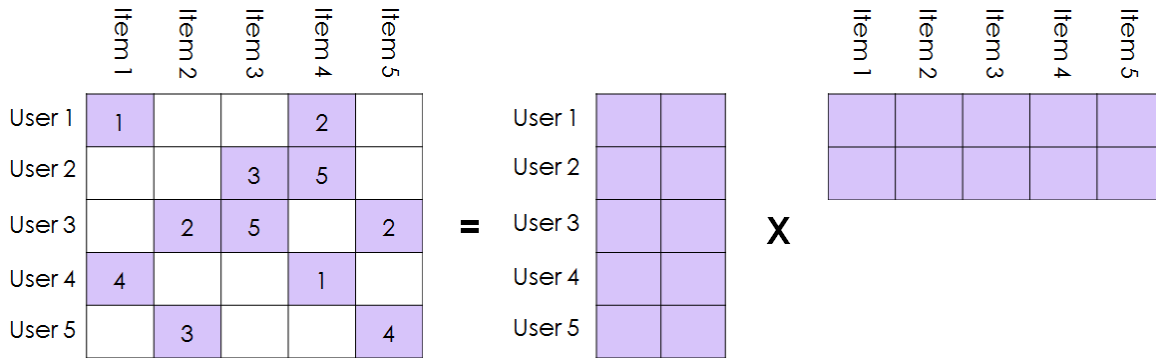
Figure 1.3: Representation of recommendation system model as matrix completion task.

Having obtained gradient, we can now formulate the update rules for $\mathbf{U}_{ik}$ and $\mathbf{V}_{jk}$:

$$\mathbf{U}_{ik}^{new} = \mathbf{U}_{ik} + 2\alpha e_{ij}\mathbf{V}_{jk} \tag{1.8}$$

$$\mathbf{V}_{jk}^{new} = \mathbf{V}_{jk} + 2\alpha e_{ij}\mathbf{U}_{ik} \tag{1.9}$$

Where α is a constant which determine the size of our steps in each iteration.

1.2 Recommendation Systems

The goal in the recommendation systems is to predict a rating that a user (customers, visitors, app users, readers) may give to an item (products, movies, events, articles) or vice versa. To do this prediction, our model needs to utilize all the information from the ratings that this user gave to other items and the ratings that this item achieved from other users. Accordingly, one method to solve this problem is treating it as a matrix completion task. As a result, we can demonstrate the ratings as a matrix $\mathbf{Y}$ with users representing the rows and items representing the columns. A simple demonstration of matrix factorization on recommendation systems is provided in figure 1.3.

There exist many recommendation systems tasks but here we mainly focus on two movie-user rating datasets. The first one is *smallnetflix* which is movie ratings data for 95526 users and 3561 movies, where the training dataset contains $3,298,163$ ratings and validation contains $545,177$ ratings, while each one of ratings is an integer in $\{1,2,3,4,5\}$. We also consider another movie recommendation dataset, *Movielens 100k*, with $100,000$ ratings from 1000 users on 1700 movies, with the same range as *smallnetflix*.

Chapter 2

Link Functions and Generalized Matrix Rank

In this chapter using our notation for matrix factorization which presented in previous chapter, we introduce link functions and *generalized link-rank*. We will focus on the round function and round-rank, introduce their generalized versions, and present their properties.

2.1   Generalized Link-rank

In this work we will focus on the round function and round-rank, introduce their generalized versions, and present their properties.

Link Functions and Link-Rank: Since the matrix $\mathbf{Y}$ may be from a domain $\mathbb{V}^{n \times m}$ different from real matrices, link functions can be used to define an alternate factor-

ization:

$$\mathbf{Y} = \psi_{\boldsymbol{\tau}}(\mathbf{X}), \mathbf{X} = \mathbf{U}\mathbf{V}^T, \tag{2.1}$$

where $\mathbf{Y} \in \mathbb{V}^{n \times m}$, $\psi : \mathbb{R} \to \mathbb{V}$ (applied element-wise), $\mathbf{X} \in \mathbb{R}^{n \times m}$, $\mathbf{U} \in \mathbb{R}^{n \times r}$, $\mathbf{V} \in \mathbb{R}^{n \times r}$, and $\boldsymbol{\tau}$ represent parameters of the link function, if any. Examples of link functions that we will study in this paper include the *round* function for binary matrices, and its generalization to ordinal-valued matrices. Link functions were introduced for matrix factorization by [29], consequently [30] presented their generalization to loss functions and regularization for *abstract data types*.

Definition 2.1.1. *Given a matrix* $\mathbf{Y}$ *and a link function* $\psi_{\boldsymbol{\tau}}$ *parameterized by* $\boldsymbol{\tau}$, *the* ***link-rank*** r_{ψ} *of* $\mathbf{Y}$ *is defined as the minimal rank of a real-matrix* $\mathbf{X}$ *such that,* $\mathbf{Y} = \psi_{\boldsymbol{\tau}}(\mathbf{X})$,

$$r_{\psi}(\mathbf{Y}) = \min_{\mathbf{X} \in \mathbb{R}^{n \times m}, \boldsymbol{\tau}} \{r(\mathbf{X}); \mathbf{Y} = \psi_{\boldsymbol{\tau}}(\mathbf{X})\} \tag{2.2}$$

Note that with $\psi \equiv I$, i.e. $\psi(x) = x$, $r_{\psi}(\mathbf{Y}) = r(\mathbf{Y})$.

Sign and Round Rank: If we consider the sign function as the link function, where $\text{sign}(x) = \{0 \text{ if } x < 0, 1 \text{ o.w.}\}$ (applied element-wise to the entries of the matrix), the link-rank defined above corresponds to the well-known sign-rank for binary matrices [20]:

$$\text{sign-rank}(\mathbf{Y}) = \min_{\mathbf{X} \in \mathbb{R}^{n \times m}} \{r(\mathbf{X}); \mathbf{Y} = \text{sign}(\mathbf{X})\}. \tag{2.3}$$

A variation of the sign function that uses a threshold τ, $\text{Round}_{\tau}(x) = \{0 \text{ if } x < \tau, 1 \text{ o.w.}\}$ when used as a link function results in the round-rank for binary matrices,

10

i.e.

$$\text{round-rank}_{\tau}(\mathbf{Y}) = \min_{\mathbf{X} \in \mathbb{R}^{n \times m}} \left\{ r(\mathbf{X}); \mathbf{Y} = \text{Round}_{\tau}(\mathbf{X}) \right\}, \tag{2.4}$$

as shown in [20]. Thus, our notion of *link-rank* not only unifies existing definitions of rank, but can be used for novel ones, as we will do next.

Generalized Round-Rank (GRR): Many matrix factorization applications use ordinal values, i.e $\mathbb{V} = \{0, 1, \ldots, N\}$. For these, we define generalized round function (GRF):

$$\text{GRF}_{\tau_1, \ldots, \tau_N}(x) = \begin{cases} 0 & x \leq \tau_1 \\ 1 & \tau_1 < x \leq \tau_2 \\ \vdots & \\ N-1 & \tau_{N-1} < x \leq \tau_N \\ N & \text{o.w.} \end{cases} \tag{2.5}$$

where its parameters $\boldsymbol{\tau} \equiv \{\tau_1, \ldots, \tau_N\}$ are thresholds (sorted in ascending order).A simple representation of GRF is depicted in figure 2.1. Accordingly, we define *generalized round-rank* (GRR) for any ordinal matrix $\mathbf{Y}$ as:

$$\text{GRR}_{\boldsymbol{\tau}}(\mathbf{Y}) = \min_{\mathbf{X} \in \mathbb{R}^{n \times m}} \left\{ r(\mathbf{X}); \mathbf{Y} = \text{GRF}_{\boldsymbol{\tau}}(\mathbf{X}) \right\}. \tag{2.6}$$

Here, we are primarily interested in exploring the utility of GRR and, in particular, compare the representation capabilities of low-GRR matrices to low-linear rank matrices. To this end, we present the following interesting property of GRR.
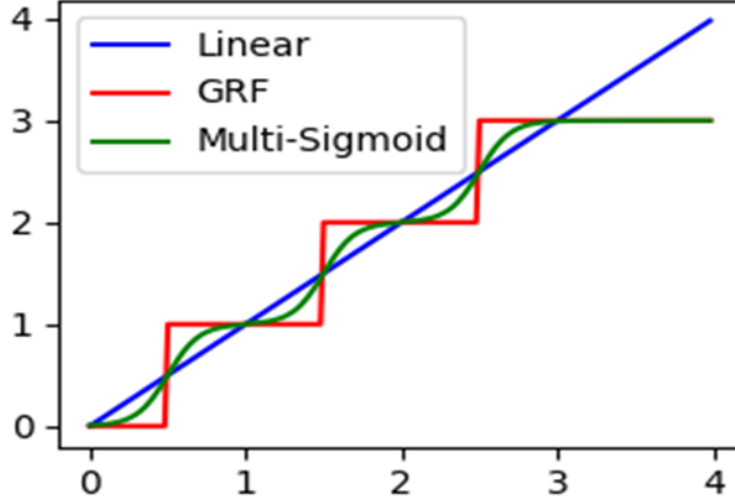
Figure 2.1: Generalized round function representation

Theorem 2.1.1. *For a given matrix* $\mathbf{Y} \in \{0, \ldots, N\}^{n \times m}$, *let's assume* $\boldsymbol{\tau}^*$ *is the set of optimal thresholds, i.e.* $GRR_{\boldsymbol{\tau}^\star}(Y) = argmin_{\boldsymbol{\tau}} GRR_{\boldsymbol{\tau}}(Y)$, *then for any other* $\boldsymbol{\tau}'$:

$$GRR_{\boldsymbol{\tau}'}(\mathbf{Y}) \leq N \times GRR_{\boldsymbol{\tau}^\star}(\mathbf{Y}) + 1 \tag{2.7}$$

Proof. To prove above inequality we first need two following lemmas:

Lemma 2.1.1. *We have the following inequality for GRR:*

$$GRR_{\tau_1 + c, \ldots, \tau_N + c}(\mathbf{Y}) \leq GRR_{\tau_1, \ldots, \tau_N}(\mathbf{Y}) + 1 \tag{2.8}$$

Where c is a real number.

Proof. We define $\mathcal{B}$ and $\mathcal{B}'$ as follows:

$$\mathcal{B} = \{B | \mathrm{GRF}_{\tau_1,...,\tau_N}(B) = \mathbf{Y}\} \tag{2.9}$$

$$\mathcal{B}' = \{B' | \mathrm{GRF}_{\tau_1+c,...,\tau_N+c}(B') = \mathbf{Y}\} \tag{2.10}$$

For an arbitrary $B \in \mathcal{B}$ let's assume we have matrix $\mathbf{U}$ and $\mathbf{V}$ in a way that, $B = \mathbf{U} \times \mathbf{V}^T$. If we add a column to the end of $\mathbf{U}$ and a row to the and of $\mathbf{V}$ and call them $\mathbf{U}'$ and $\mathbf{V}'$ as follows:

$$U' = \begin{bmatrix} & c \\ U & \vdots \\ & c \end{bmatrix}, \qquad V' = \begin{bmatrix} & 1 \\ V & \vdots \\ & 1 \end{bmatrix} \tag{2.11}$$

It is clear that $B' = \mathbf{U}' \times \mathbf{V}'^T \in \mathcal{B}'$. Furthermore, by using the fact that $\mathrm{r}(B') \leq \mathrm{r}(B)+1$ we can complete the proof. $\qquad \square$

Lemma 2.1.2. *For arbitrary $k \in \mathbb{R}$, the following equality holds:*

$$GRR_{k\tau_1,...,k\tau_N}(\mathbf{Y}) = GRR_{\tau_1,...\tau_N}(\mathbf{Y}) \tag{2.12}$$

Proof. Similar to previous Lemma, if we define $\mathcal{B}$ and $\mathcal{B}'$ as follows:

$$\mathcal{B} = \{B | \mathrm{GRF}_{\tau_1,...\tau_N}(B) = A\} \tag{2.13}$$

$$\mathcal{B}' = \{B' | \mathrm{GRF}_{k\tau_1,...,k\tau_N}(B') = A\} \tag{2.14}$$

For any $B \in \mathcal{B}$ it is clear that $k \times B \in \mathcal{B}'$. On the other hand, for any $B' \in \mathcal{B}'$ we know that $B'/k \in \mathcal{B}$.In result, by considering the fact that $\mathrm{r}(kB) = \mathrm{r}(B)$, we can complete the proof . $\qquad \square$

13

base on These lemmas and the fact that for any $i \in \{1, ..., N-1\}$, there exist an ϵ_i which will satisfies the following equality:

$$\mathrm{GRR}_{\tau_1,...,\tau_i-\epsilon_i,...,\tau_N}(\mathbf{Y}) = \mathrm{GRR}_{\tau_1,...\tau_N}(\mathbf{Y}) \tag{2.15}$$

We can show that there exists a set of ϵ_i ($i \in \{1, ..., N-1\}$), that transform $(\tau_1, ...\tau_N)$ in to $(\tau_1', ..., \tau_N')$ with a set of linear combinations. In another word, it means we have $k_0, ..., k_{N-1}$ in a way that:

$$T' = k_0 T_0 + ... + k_{N-1} T_{N-1} \tag{2.16}$$

Where $T' = (\tau_1', ...\tau_N')$, $T_0 = (\tau_1, ...\tau_N)$ and $T_i = (\tau_1, ..., \tau_i - \epsilon_i, ..., \tau_N)$ in vector format. Therefore, if we define $\mathcal{B}_i$ as follows:

$$\mathcal{B}_i = \{B_i | \mathrm{GRF}_{T_i}(B_i) = A\} \tag{2.17}$$

And considering the fact that:

$$\mathrm{r}(k_0 B + ... + k_{N-1} B_{N-1}) \leq \sum_{j=0}^{N-1} \mathrm{r}(k_j B_j) \tag{2.18}$$

$$= \sum_{j=0}^{N-1} \mathrm{r}(B_j) \tag{2.19}$$

Finally, with Lemma 2.1.1 equation 2.15 we can complete the theorem. $\square$

This theorem shows that even though using a fixed set of thresholds is not optimal, the rank is still bounded in terms of N, and does not depend on the size of the matrix (n or m). Other complementary lemmas are provided in appendix.

Remark 2.1.1. *The upper bound in the theorem 2.1.1 matches the upper bound found*

in [21] for the case where $N = 1$, $GRR_{\tau'}(\mathbf{Y}) \leq GRR_{\tau^*}(\mathbf{Y}) + 1$.

2.2 Comparing Generalized Round Rank to Linear Rank

Matrix factorization (MF) based on linear rank has been widely used in lots of machine learning problems like matrix completion, matrix recovery and recommendation systems. The primary advantage of matrix factorization is its ability to model data in a compact form. Being able to represent the same data accurately in an even more compact form, specially when we are dealing with high rank matrices, is thus quite important. Here, we study specific aspects of exact and approximate matrix reconstruction with GRR. In particular, we introduce matrices with high linear rank but low GRR, and demonstrate the inability of linear factorization in approximating many low-GRR matrices.

2.2.1 Exact Low-Rank Reconstruction

To compare linear and GRR matrix factorization, here we identify families of matrices that have high (or full) linear rank but low (or constant) GRR. Such matrices demonstrate the primary benefit of GRR over linear rank: factorizing matrices using GRR can be significantly beneficial.

As provided in [20] for round-rank (a special case of GRR), $GRR_{\tau}(\mathbf{Y}) \leq r(\mathbf{Y})$ for any matrix $\mathbf{Y} \in \mathbb{V}^{n \times m}$. More importantly, there are many structures that lower bound the linear rank of a matrix. For example, if we define the upper triangle number n_U for

matrix $\mathbf{Y} \in \mathbb{V}^{n \times n}$ as the size of the biggest square block which is in the form of an upper triangle matrix, then $r(\mathbf{Y}) \geq n_U$. If we define the identity number n_I similarly, then $r(\mathbf{Y}) \geq n_I$, and similarly for matrices with a band diagonal submatrix. None of these lower bounds that are based on identity, upper-triangle, and band-diagonal structures apply to GRR. In particular, as shown in [20], identity matrices (of any size) have a constant round-rank of 2, upper triangle matrices have round-rank of 1, and band diagonal matrices have round-rank of 2 (which also holds for GRR). Moreover, we provide another lower bound for linear rank of a matrix, which is again not applicable to GRR.

Theorem 2.2.1. *If a matrix $\mathbf{Y} \in \mathbb{R}^{n \times m}$ contains k rows, $k \leq n, k \leq m$, such that $R = \{Y_{R_1}, ..., Y_{R_k}\}$, two columns $C = \{j_0, j_1\}$, and:*

1. *rows in R are distinct from each other, i.e, $\forall i, i' \in R, \exists j, Y_{ij} \neq Y_{i'j}$,*
2. *columns in C are distinct from each other, i.e, $\exists i, Y_{ij_0} \neq Y_{ij_1}$, and*
3. *matrix spanning R and C are non-zero constants, w.l.o.g. $\forall i \in R, Y_{ij_0} = Y_{ij_1} = 1$,*

then $r(\mathbf{Y}) \geq k$.

Proof. Let us assume $r(\mathbf{Y}) < k$, i.e. $\exists k' < k, \mathbf{U} \in \mathbb{R}^{k' \times n}, \mathbf{V} \in \mathbb{R}^{k' \times m}$ such that $\mathbf{Y} = \mathbf{U}^T \times \mathbf{V}$. Since the rows R and the columns in C are distinct, their factorizations in $\mathbf{U}$ and $\mathbf{V}$ have to also be distinct, i.e. $\forall i, i' \in R, i \neq i', \mathbf{U}_i \neq \mathbf{U}_{i'}$ and $\mathbf{V}_{j_0} \neq \mathbf{V}_{j_1}$. Furthermore, $\forall i, i' \in R, i \neq i', \nexists a, \mathbf{U}_i = a\mathbf{U}_{i'}$ and $\nexists a, \mathbf{V}_{j_0} = a\mathbf{V}_{j_1}$ for $a \neq 0$, it is clear that $\mathbf{U}_i \cdot \mathbf{V}_{j_0} = \mathbf{U}_i \cdot \mathbf{V}_{j_1} = 1$ (and similarly for $i, i' \in R$).

Now consider a row $i \in R$. Since $\forall j \in C, \mathbf{Y}_{ij} = 1$, then $\mathbf{U}_i \cdot \mathbf{V}_j = 1$. As a result, $\mathbf{V}_j$ are distinct vectors that lie in the hyperplane spanned by $\mathbf{U}_i \cdot \mathbf{V}_j = 1$. In other words, the hyperplane $\mathbf{U}_i \cdot \mathbf{V}_j = 1$ defines a k'-dimensional hyperplane tangent to the unit

16

hyper-sphere.

Going over all the rows in R, we obtain constraints that $\mathbf{V}_j$ are distinct vectors that lie in the intersection of the hyperplanes spanned by $\mathbf{U}_i \cdot \mathbf{V}_j = 1$ for all $i \in R$. Since all $\mathbf{U}_i$s are distinct, there are k distinct k'-dimensional hyperplanes, all tangent to the unit sphere, that intersect at more than one point (since $\mathbf{V}_j$s are distinct).

Since k hyper-planes tangent to unit sphere can intersect at at most one point in $k' < k$ dimensional space, $\mathbf{V}_j$ cannot be distinct vectors. Hence, our original assumption $k' < k$ is wrong, therefore, $\mathrm{r}(\mathbf{Y}) \geq k$. $\qquad\square$

So far, we provide examples of high linear-rank structures that do not impose any constraints on GRR. We now provide the following lemma that, in conjunction with above results, indicates that lower bounds on the linear rank can be really high for matrices if they contain low-GRR structures (like identity and upper-triangle), while the lower bound on GRR is low.

Lemma 2.2.1. *For any matrix A, if there exists a submatrix A' in a way that $\mathrm{r}(A') = R$ and $GRR_\tau(A') = r$, then $GRR_\tau(A) \geq r$ and $\mathrm{r}(A) \geq R$.*

Proof. If we consider the linear rank as the number of independent row (column) of the matrix, consequently having a rank of R for submatrix A' means there exist at least R independent rows in matrix A. Using this argument we can simply prove above inequalities. $\qquad\square$

2.2.2   Approximate Low-Rank Reconstruction

Apart from examples of high linear-rank matrices that have low GRR, we can further show that many of these matrices cannot even be *approximated* by a linear factorization. In other words, we show that there exist many matrices for which not only their linear rank is high, but further, that the linear rank approximations are poor as well, while their low GRR reconstruction is perfect. In order to measure whether a matrix can be approximated well, we describe the notion of approximate rank (introduced by [1], we rephrase it here in our notation).

Definition 2.2.1. *Given ϵ, **approximate rank** of a matrix $\mathbf{X}$ is:*

$$\epsilon\text{-}rank(\mathbf{X}) = \min\{\mathrm{r}(\mathbf{X}') : \mathbf{X}' \in \mathbb{R}^{n \times m}, ||\mathbf{X} - \mathbf{X}'||_F^2 \leq \epsilon\} \tag{2.20}$$

We extend this definition to introduce the generalized form of approximate rank as follows:

Definition 2.2.2. *Given ϵ and a link function ψ (e.g. GRF), the **generalized approximate rank** of a matrix $\mathbf{Y}$ is defined as:*

$$\epsilon\text{-}rank_\psi(\mathbf{Y}) = \min\{\mathrm{r}_\psi(\mathbf{Y}') : \mathbf{Y}' \in \mathbb{V}^{n \times m}, ||\mathbf{Y} - \mathbf{Y}'||_F^2 \leq \epsilon\} \tag{2.21}$$

For an arbitrary matrix, we can evaluate how well a linear factorization can approximate it using SVD, i.e.:

Theorem 2.2.2. *For a matrix $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, where $\mathrm{diag}(\Sigma)$ are the singular values, and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices, then $\sum_{i=k+1}^{n} |\Sigma_{ii}|^2 = \min_{\mathbf{Y},\mathrm{r}(\mathbf{Y})=k} ||\mathbf{X} - \mathbf{Y}||_F^2$.*

18
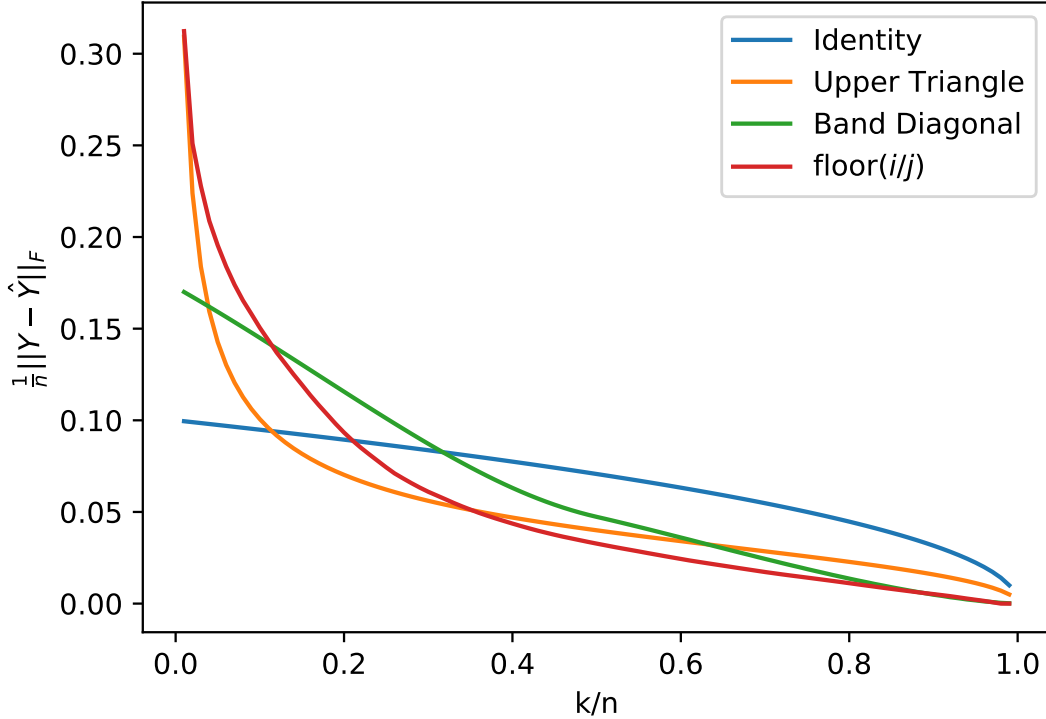
Figure 2.2: Comparison of the optimal linear factorization approximation as the rank k is varied for a number of matrices (of size $n \times n$), demonstrating that linear factorization is unable to approximate these matrices with low-rank. All of these matrices have a *constant* generalized round-rank (≤ 2).

Proof. This was first introduced in [7], and recently presented again in [30]. We omit the detailed proof, but the primary intuition is that the PCA decomposition minimizes the Frobenius norm, and $\mathbf{Y} = \mathbf{U}'\mathbf{V}'$, with $\mathbf{U}' = \mathbf{U}\Sigma^{\frac{1}{2}}$ and $\mathbf{V}' = \Sigma^{\frac{1}{2}}\mathbf{V}^T$. $\square$

For an arbitrary binary matrix $\mathbf{Y}$, recall that $\text{Round}_{\tau=0}(\mathbf{Y})$ is equal to sign-rank($\mathbf{Y}$). Using above theorem, we want to show that there are binary matrices that cannot be approximated by low linear-rank matrices (for non-trivial ϵ), but can be approximated well by low round-rank matrices. Clearly, these results extend to ordinal matrices and their GRR approximations, the generalized form of binary case.

Let us consider $\mathbf{Y}$, the identity binary matrix of size n, for which the singular values of $\mathbf{Y}$ are all 1s. By using Theorem 2.2.2, any linear factorization $\mathbf{Y}'$ of rank k will have $||\mathbf{Y} - \mathbf{Y}'||_F^2 \geq (n - k)$. As a result, the identity matrix cannot be approximated by *any* rank-k linear factorization for $\epsilon < n - k$. On the other hand, such a matrix can be reconstructed exactly with a rank 2 factorization if using the round-link function, since round-rank$(\mathbf{Y}) = 2$. In Figure 2.2, we illustrate a number of other such matrices, i.e. they can be exactly represented by a factorization with GRR of 2, but cannot be approximated by any compact linear factorization.

Chapter 3

Matrix Completion with Generalized Round-Rank Factorization

So far, we show that there are many matrices that cannot be represented compactly using conventional matrix factorization (linear), either approximately or exactly, whereas they can be reconstructed using compact matrices when using GRF as the link function. In this section, we study properties of *completion* of ordinal-valued matrices based on GRF (and the notion of rank from GRR). In particular, given a number of noise-free observations Ω from $\mathbf{Y} \in \{0, \dots, N\}^{n \times m}$ and its $\text{GRR}(\mathbf{Y}) = r, r \ll \min(n, m)$, the goal here is to identify $\mathbf{U} \in \mathbb{R}^{n \times r}, \mathbf{V} \in \mathbb{R}^{m \times r}$ such that $\text{GRF}(\mathbf{U}\mathbf{V}^T)$ completes the unobserved entries of $\mathbf{Y}$ accurately.

3.1 Theoretical Results for Uniqueness

Uniqueness in matrix completion is defined as the minimum number of entries required to recover the matrix $\mathbf{Y}$ with high probability, assuming that sampling of the set of observed entries is based on an specific distribution. To obtain uniqueness in GRR based factorization, we first need to introduce the interval matrix $\bar{\mathbf{X}}$. Based on definition of generalized round function (GRF) and a set of fixed thresholds, we define matrix $\bar{\mathbf{X}}$ to be a matrix with interval entries calculated based on entries of matrix $\mathbf{Y}$ and thresholds $(\tau_1, ... \tau_N)$. As an example, if an entry $\mathbf{Y}_{ij}$ is $k \in \{0, ..., N\}$, $\bar{\mathbf{X}}_{\mathbf{ij}}$ would be equal to the interval $[\tau_k, \tau_{k+1}]$. When entries of $\mathbf{Y}$ are equal to 0 or N, w.l.o.g. we assume the corresponding entries in matrix $\bar{\mathbf{X}}$ are bounded. Thus, each one of matrix $\bar{\mathbf{X}}$'s entries must be one of the $N+1$ possible intervals based on GRF's thresholds.

Definition 3.1.1. *A target matrix $\mathbf{Y} \in \{0, \ldots, N\}^{n \times m}$ with 1) observed set of entries $\Omega = \{(i,j), \mathbf{Y}_{ij} \text{is observed}\}$, 2) set of known thresholds $(\tau_1, ... \tau_N)$, and 3) $GRR_{\tau_1, ..., \tau_N}(\mathbf{Y}) = r$, is called uniquely recoverable, if we can recover its unique interval matrix $\bar{\mathbf{X}}$ with high probability.*

Similar to $\bar{\mathbf{X}}$, we introduce $\mathcal{X}^\star$ to be a set of all matrices that satisfy following two conditions: 1) For the observed entries Ω of $\mathbf{Y}$, $\mathbf{Y}_{ij} = \text{GRF}_{\tau_1, ..., \tau_N}(\mathbf{X}_{ij}^\star)$, and 2) linear rank of $\mathcal{X}^\star$ is r. If we consider a matrix $\mathbf{X} \in \mathcal{X}^\star$ then for an arbitrary entry $\mathbf{X}_{ij}$ we must have $\mathbf{X}_{ij} \in \bar{\mathbf{X}}_{ij}$, where $\bar{\mathbf{X}}_{ij}$ is an interval containing $\mathbf{X}_{ij}$. Given a matrix $\mathbf{X} \in \mathcal{X}^\star$, the uniqueness conditions ensure that we would be able to recover $\bar{\mathbf{X}}$, using which we can uniquely recover matrix $\mathbf{Y}$.

In the next theorems, we first find the necessary condition on the entries of matrix $\mathbf{X}$ for satisfying uniqueness of matrix $\mathbf{Y}$. Then, we derive the sufficient condition accordingly. In our calculations, we assume the thresholds to be fixed and our target matrix $\mathbf{Y}$ be

noiseless, and further, there is at least one observed entry in every column and row of matrix $\mathbf{Y}$.

Theorem 3.1.1. *(Necessary Condition) For a target matrix $\mathbf{Y} \in \mathbb{V}^{n \times m}$ with few observed entries and given $GRR(\mathbf{Y}) = r$, we consider set of $\{\mathbf{Y}_{i_1 j}, ..., \mathbf{Y}_{i_r j}\}$ to be the r observed entries in an arbitrary column j of Y. Given any matrix $\mathbf{X} \in \mathcal{X}^\star$, $\mathbf{X} = \mathbf{U} \times \mathbf{V}^T$, and taking an unobserved entry $\mathbf{Y}_{ij}$, we define $a_{i_k j}$ as: $\mathbf{U}_i = \sum_{k=1}^{r} a_{i_k j} \mathbf{U}_{i_k}$, where $\mathbf{U}_d$ ($d \in \{1, ..., n\}$) is the dth row of matrix $\mathbf{U}$ and i_k represents the index of observed entries in jth column. Then, the necessary condition of uniqueness of $\mathbf{Y}$ is:*

$$\sum_{k=1}^{r} |a_{i_k j}| \leq \epsilon \left(\frac{T_{min}}{T_{max}} \right) \tag{3.1}$$

Where $r = GRR(\mathbf{Y})$, T_{min} and T_{max} are the length of smallest and largest intervals and ϵ is a small constant.

Proof. To better understand the concept of uniqueness in GRR benchmark, let's first look at the uniqueness in fixed value matrix factorization (traditional definition(MF)).

In fixed value matrix factorization, it is proved that to achieve uniqueness, we need at least $r = r(\mathbf{X})$ observation in each column(other than the independent columns). Therefore, if we decompose $\mathbf{X}$ as $\mathbf{X} = \mathbf{U}\mathbf{V}^T$, and plan to changed only unobserved entries of $\mathbf{Y}$ in column j (in opposed to uniqueness), we need to change the *jth* row of matrix $\mathbf{V}$. To do so, let's assume we change the *jth* row to:

$$[\mathbf{V}_{j1} + c_1, ..., \mathbf{V}_{jr} + c_r] \tag{3.2}$$

Now since we know $r(U) = r$ and assume the respective rows of $\mathbf{U}$ to observed entries of column j in matrix $\mathbf{X}$ are independent (this is a required assumption for uniqueness),

23

we can show that only possible value for $c_1, ..., c_r$ which does not change the observed entries of $\mathbf{X}$ is 0, which confirm the uniqueness.

The biggest difference between MF based on GRR and traditional MF is the fact that the observed entries of matrix $\mathbf{X}$ are not fixed in GRR version, and can change through the respective interval. In result, to achieve uniqueness we need to find a condition which for any column of $\mathbf{X}$, by changing respective row of $\mathbf{V}$, while the value of observed entries stay in the respected intervals, the value of unobserved ones wouldn't change dramatically which result in moving to other intervals. To do so, we will calculate the maximum of the possible change for an arbitrary unobserved entry of column j in matrix $\mathbf{Y}$.

Let's call the r observed entries of column's j of matrix $\mathbf{Y}$, $\mathbf{Y}_{i_1 j}, ..., \mathbf{Y}_{i_r j}$. Similar to MF case, we assume that the respective rows of $\mathbf{U}$ to these entries are independent. In result, if we represent the change in entries of jth rows of $\mathbf{V}$ by c_i, we should have:

$$
\begin{bmatrix} \mathbf{U}_{i_1} \\ \vdots \\ \mathbf{U}_{i_r} \end{bmatrix} \times \begin{bmatrix} c_1 \\ \vdots \\ c_r \end{bmatrix} = \begin{bmatrix} \epsilon_{i_1 j} \\ \vdots \\ \epsilon_{i_r j} \end{bmatrix} \tag{3.3}
$$

Where $\mathbf{U}_{i_k}$ is the $i_k th$ row of $\mathbf{U}$, and $\epsilon_{i_k j}$ is the possible change for $\mathbf{X}_{i_k j}$, based on the observed interval. Therefore:

$$
\epsilon_{i_k j} \in \left(\tau_{i_k j} \downarrow -\mathbf{X}_{i_k j}, \tau_{i_k j} \uparrow -\mathbf{X}_{i_k j} \right) = \left(\epsilon_{i_k j}^-, \epsilon_{i_k j}^+ \right) \tag{3.4}
$$

Now let's assume we want to find the maximum possible change for $\mathbf{X}_{sj}$ considering that $\mathbf{Y}_{sj}$ is and unobserved entry. Since $\mathbf{U}_{i_k}$'s are independent, there exist $a_1, ..a_r$

24

which:

$$\mathbf{U}_s = \sum_{k=1}^{r} a_{i_k j} \mathbf{U}_{i_k} \tag{3.5}$$

Therefore, we can show the change in entry $\mathbf{X}_{sj}$ as:

$$A = \sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k j} \tag{3.6}$$

In result, for the maximum possible change we have:

$$max|A| = max(\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k j}^{sign(a_{i_k j})}, |\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k j}^{-sign(a_{i_k j})}|) \tag{3.7}$$

Where $sign(.)$ is the sign function. On the other hand we know:

$$\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k j}^{sign(a_{i_k j})} + |\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k j}^{-sign(a_{i_k j})}| = \sum_{k=1}^{r} |a_{i_k j}| T_{i_k j} \tag{3.8}$$

$$\Rightarrow max|A| \geqslant \frac{1}{2} \sum_{k=1}^{r} |a_{i_k j}| T_{i_k j} \tag{3.9}$$

Where $T_{i_k j}$ is the length of the interval entry of $\bar{\mathbf{X}}_{i_k j}$. Clearly, to achieve the uniqueness we need $max|A| \leq T_{sj}$. But, since the entry $\mathbf{X}_{sj}$ is unobserved we don't know the value of T_{sj}. In result, for sake of uniqueness in the worst case we need:

$$\sum_{k=1}^{r} |a_{i_k j}| T_{max} \leq \epsilon T_{min} \tag{3.10}$$

$$\Rightarrow \sum_{k=1}^{r} |a_{i_k j}| \leq \epsilon \frac{T_{min}}{T_{max}} \tag{3.11}$$

Where T_{min} and T_{max} are the smallest and the biggest interval, and ϵ is a small real

constant. $\qquad$ □

The same condition is necessary for matrix $\mathbf{V}$ as well. The necessary condition must be satisfied for all columns of matrix $\mathbf{X}$. Moreover, if the necessary condition is not satisfied, we cannot find a unique matrix $\mathbf{X}$, and hence a unique completion, i.e. $\mathbf{Y} = \mathrm{GRF}_{\tau_1,\ldots,\tau_N}(\mathbf{X})$ where $\mathbf{X} \in \mathcal{X}^\star$.

Theorem 3.1.2. *(Sufficient Condition) Using above necessary condition, for any un-observed entry $\mathbf{Y}_{ij}$ of matrix $\mathbf{Y}$ we define $\bar{\epsilon}$ as minimum distance of $\mathbf{X}_{ij}$ with its respected interval's boundaries. Then, we will have the following inequality as sufficient condition of uniqueness:*

$$\bar{\epsilon} \geq \max\left(\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k}^{sign(a_{i_k j})}, \left|\sum_{k=1}^{r} a_{i_k j} \epsilon_{i_k}^{-sign(a_{i_k j})}\right|\right) \qquad (3.12)$$

where r and $a_{i_k j}$ are defined as before, $\epsilon_{i_k j}^{+}$ is defined as the distance of $\mathbf{X}_{i_k j}$ to its upper bound, and $\epsilon_{i_k j}^{-}$ is defined as negative of the distance of $\mathbf{X}_{i_k j}$ to its lower bound.

Above sufficient condition is a direct result of necessary condition proof. Although not tight, it guarantees the existence of unique $\bar{\mathbf{X}}$, and thus the complete matrix $\mathbf{Y}$.

3.2 Gradient-Based Algorithm for GRR Factorization

Although previous studies have used many different paradigms for matrix factorization, such as alternating minimization [9, 11] and adaptive sampling [13], stochastic gradient descent-based (SGD) approaches have gained widespread adoption, in part due to their

flexibility, scalability, and theoretical properties [6]. For linear matrix factorization, a loss function that minimizes the squared error is used, i.e. $L_{\text{linear}} = \sum (Y_{ij} - U_i V_j)^2$, where the summation is over the observed entries. In order to prevent over-fitting, L_2 regularization is often incorporated.

Round: We extend this framework to support GRR-based factorization by defining an alternate loss function. In particular, with each observed entry Y_{ij} and the current estimate of $\boldsymbol{\tau}$, we compute the $b_{ij}^{\downarrow}$ and $b_{ij}^{\uparrow}$ as the lower and upper bounds for X_{ij} with respect to the GRF. Given these, we use the following loss, $L_{\text{Round}} = \sum (b_{ij}^{\downarrow} - U_i V_j)_+ + (U_i V_j - b_{ij}^{\uparrow})_+$, where $(.)_+ = \max(., 0)$. Considering the regularization term as well, we apply stochastic gradient descent as before, computing gradients using a differentiable form of max with respect to $\mathbf{U}$, $\mathbf{V}$, and $\boldsymbol{\tau}$.

Multi-Sigmoid: Although the above loss captures the goal of the GRR-based factorization accurately, it contains both discontinuities and flat regions, and thus is difficult to optimize. Instead, we also propose to use a smoother and noise tolerant approximation of the GRF function. The sigmoid function, $\sigma(x) = \frac{1}{1+e^{-x}}$, for example, is often used to approximate the sign function. When used as a link function in factorization, we can further show that it approximates the sign-rank well.

Theorem 3.2.1. *For any $\epsilon > 0$ and matrix $\mathbf{Y}$, sign-rank$(\mathbf{Y}) = \epsilon$-rank$_\sigma(\mathbf{Y})$.*

Proof. Let $\mathcal{B}_\sigma^\epsilon(k) = \{\mathbf{B} \in \{0,1\}^{n \times m}; \epsilon\text{-rank}_\sigma(\mathbf{B}) = k\}$, i.e. the set of binary matrices whose ϵ-rank$_\sigma$ is equal to k, and $\mathcal{B}_+(k) = \{\mathbf{B} \in \{0,1\}^{n \times m}; \text{sign-rank}(\mathbf{B}) = k\}$. We prove the theorem by showing both directions. $\underline{\mathcal{B}_+ \subseteq \mathcal{B}_\sigma}$: Any $\mathbf{U}, \mathbf{V}$ that works for $+$ should work with σ if multiplied by a very large number, i.e. take a sufficiently large η, and $\mathbf{U}_\sigma = \eta \mathbf{U}_+, \mathbf{V}_\sigma = \eta \mathbf{V}_+$. Then, $\mathbf{X}_\sigma = \eta^2 \mathbf{X}_+$ and if we set $\theta_\sigma = \eta^2 \theta_+$, then

27

$(\mathbf{X}_\sigma - \theta_\sigma) = \eta^2(\mathbf{X}_+ - \theta_+)$, therefore will have the same sign, and $\mathbf{Y}_\sigma = \sigma(\mathbf{X}_\sigma)$ will be arbitrarily close to 0 and 1 in $\mathbf{Y}_+$. $\underline{\mathcal{B}_\sigma \subseteq \mathcal{B}_+}$: Any $\mathbf{U}, \mathbf{V}$ that works for σ will directly work with $+$. $\qquad\square$

Remark 3.2.1. *To extend Theorem 4.3 to multi-ordinal cases, we need to show that for any arbitrary set of thresholds in GRR, there exists another set of thresholds for multi-sigmoid function which will satisfy the condition in theorem 4.3 for multi-ordinal matrices. The procedure of proof is similar to binary cases. The only difference is the fact that after multiplying our matrices into a big enough constant, we need to choose multi-sigmoids thresholds in a way that will guarantee the multi-sigmoid(X) is close enough of to GRF(X)(which is equal to Y).*

We can similarly approximate GRF using a sum of sigmoid functions that we call Multi-sigmoid defined as $\psi_{\boldsymbol{\tau}}^{m\sigma}(x) = \sum_{d=1}^{N} \sigma(x - \tau_d)$, for which the above properties also hold. The resulting loss function that minimizes the squared error is $L_{\text{multi-sigmoid}} = \sum (Y_{ij} - \psi_{\boldsymbol{\tau}}^{m\sigma}(U_i V_j))^2$.

In our experiments, we evaluate both of our proposed loss functions, and compare their relative performance. We study variations in which the thresholds $\boldsymbol{\tau}$ are either pre-fixed or updated (using $\frac{\partial}{\partial \boldsymbol{\tau}} L$) during training. All the parameters of the optimization, such as learning rate and early stopping, and the hyper-parameters of our approaches, such as regularization, are tuned on validation data.

Chapter 4

Experiments

In this chapter we evaluate the capabilities of our proposed GRR factorization relative to linear factorization first through variety of simulations, followed by considering *smallnetflix* and *MovieLens 100K*[1] datasets. Unless otherwise noted, all of evaluations are based on Root Mean Square Error (RMSE).

4.1   Matrix Recovery

We first consider the problem of recovering a fully known matrix $\mathbf{Y}$ from its factorization, thus all entries are considered observed. We create three matrices in order to evaluate our approaches for recovery: (a) Random 10×10 matrix with $N = 5$ that has GRR ≤ 2 (create by randomly generating $\boldsymbol{\tau}$, $\mathbf{U}$, and $\mathbf{V}$), (b) Binary upper triangle matrix with size 10 (GRR of 1), and (c) Band-diagonal matrix of size 10 and bandwidth 3, which has the linear rank of 8 and GRR of 2. Figures 4.1, 4.2, and 4.3 present

---

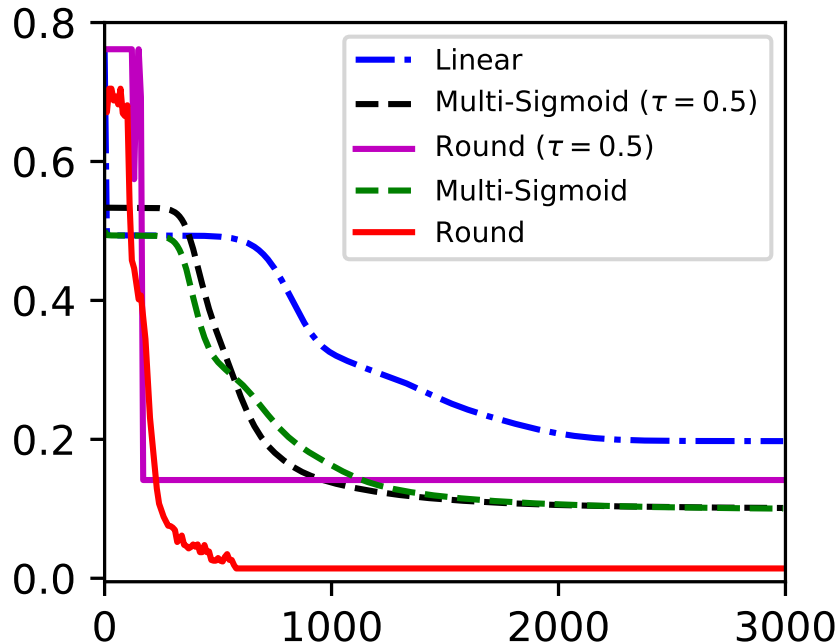[1]The codes available at: `https://github.com/pouyapez/GRR-Matrix-Factorization`

Figure 4.1: Band Diagonal matrices that are reconstructed using their $k = 2$-dimensional factorization with different representations. We plot RMSE of the reconstruction vs the number of training iterations, demonstrating the efficiency of GRR-based methods, especially without fixed thresholds.

the RMSE comparison of these three matrices as training progresses. For the upper triangle and the band diagonal, we fix threshold to $\tau = 0.5$. The results show that Round works far better than others by converging to zero. Moreover, linear approach is outperformed by the Multi-sigmoid without fixed thresholds in all, demonstrating it cannot recover even simple matrices.

4.2 Matrix Completion

Instead of fully-observed matrices, we now evaluate completion of the matrix when only a few of the entries are observed. We consider 50×50 upper-triangle and band-
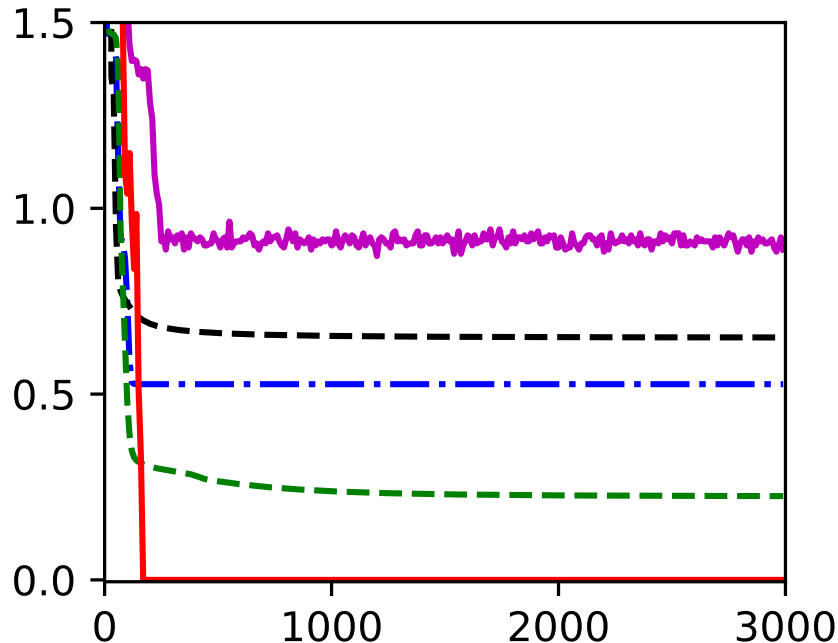
Figure 4.2: Random matrices that are reconstructed using their $k = 2$-dimensional factorization with different representations. We plot RMSE of the reconstruction vs the number of training iterations, demonstrating the efficiency of GRR-based methods, especially without fixed thresholds.

diagonal (bandwidth 10) matrices, and sample entries from them, to illustrate how well our approaches can complete them. Results on held-out 20% entries are given in Tables 4.1 and 4.2. In addition, we build a random matrix with size 50 and GRR 2, and present the results for this matrix in Table 4.3. As we can see, linear factorization in all three cases is outperformed by our proposed approaches. In band-diagonal, because of over-fitting of the Round approach, Multi-sigmoid performs a little better, and for upper-triangle, we achieve the best result for Round method by fixing $\tau = 0.5$.
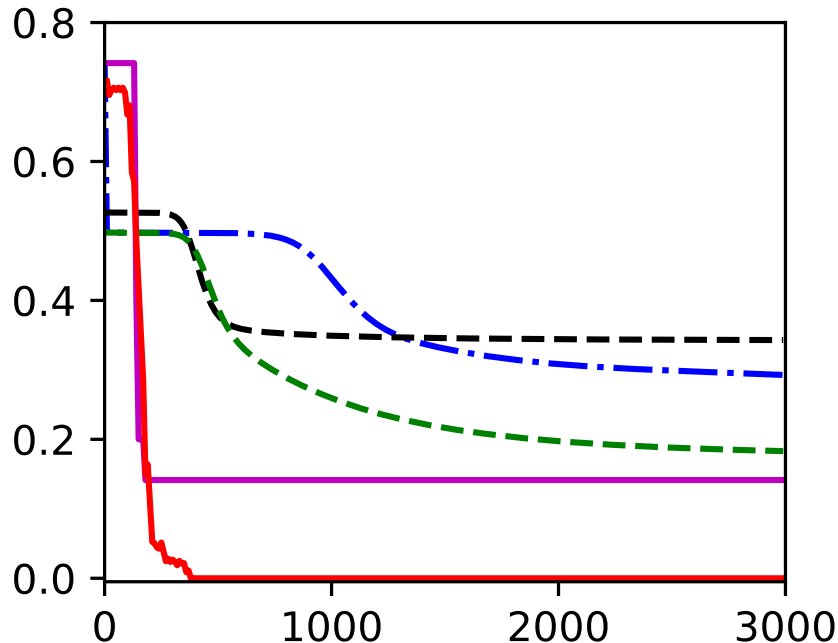
Figure 4.3: Upper Triangle matrices that are reconstructed using their $k = 1$-dimensional factorization with different representations. We plot RMSE of the reconstruction vs the number of training iterations, demonstrating the efficiency of GRR-based methods, especially without fixed thresholds.

4.3 Matrix Completion on Real Data

In this section we use the *smallnetflix* movie ratings data for 95526 users and 3561 movies, where the training dataset contains $3,298,163$ ratings and validation contains $545,177$ ratings, while each one of ratings is an integer in $\{1,2,3,4,5\}$. We also evaluate on a second movie recommendation dataset, *Movielens 100k*, with $100,000$ ratings from 1000 users on 1700 movies, with the same range as *smallnetflix*. For this recommendation systems, in addition to RMSE, we also consider the notion of accuracy that is more appropriate for the task, calculated as the fraction of predicted ratings that are within ± 0.5 of the real ratings. As shown in Figure 4.6, for *smallnetflix*, linear factorization is better than Round approach from RMSE perspective, probably because

Table 4.1: Matrix completion for **Upper Triangular Matrices** ($k = 1$)

Proportion of Observations	10%	20%	30%	40%	50%	60%	70%	80%
Linear	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
Multi-Sigmoid	0.51	0.30	0.25	0.25	0.26	0.25	0.23	0.23
Multi-Sigmoid, $\tau = 0.5$	0.58	0.37	0.36	0.36	0.35	0.35	0.34	0.34
Round	0.46	0.34	0.27	0.25	0.26	0.21	0.20	0.16
Round, $\tau = 0.5$	**0.38**	**0.26**	**0.23**	**0.19**	**0.15**	**0.13**	**0.15**	**0.13**

Table 4.2: Matrix completion for **Band Diagonal Matrices** ($k = 2$)

Proportion of Observations	10%	20%	30%	40%	50%	60%	70%	80%
Linear	0.49	0.46	0.46	0.46	0.46	0.46	0.46	0.46
Multi-Sigmoid	**0.39**	**0.26**	**0.23**	**0.23**	**0.22**	**0.21**	**0.20**	**0.20**
Multi-Sigmoid, $\tau = 0.5$	0.48	0.49	0.33	0.31	0.30	0.29	0.29	0.29
Round	0.71	0.41	0.35	0.29	0.29	0.27	0.23	0.22
Round, $\tau = 0.5$	0.61	0.57	0.39	0.52	0.58	0.30	0.29	0.34

linear is more robust to noise. On the other hand, Multi-sigmoid achieves better RMSE than linear method. Furthermore, both Round and Multi-sigmoid outperform the linear factorization in accuracy. *Movielens* results for the percentage metric shows similar behavior as *smallnetflix*, demonstrating that GRR-based factorization can provide benefits to real-world applications. Furthermore, a comparison of our models with existing approaches on *Movielens* dataset is provided in Table 4.4. We choose the RMSE result for smallest k presented in those works. As we can see, our Multi-sigmoid method

Table 4.3: Matrix completion with different number of samples for **Random low-GRR Matrices**

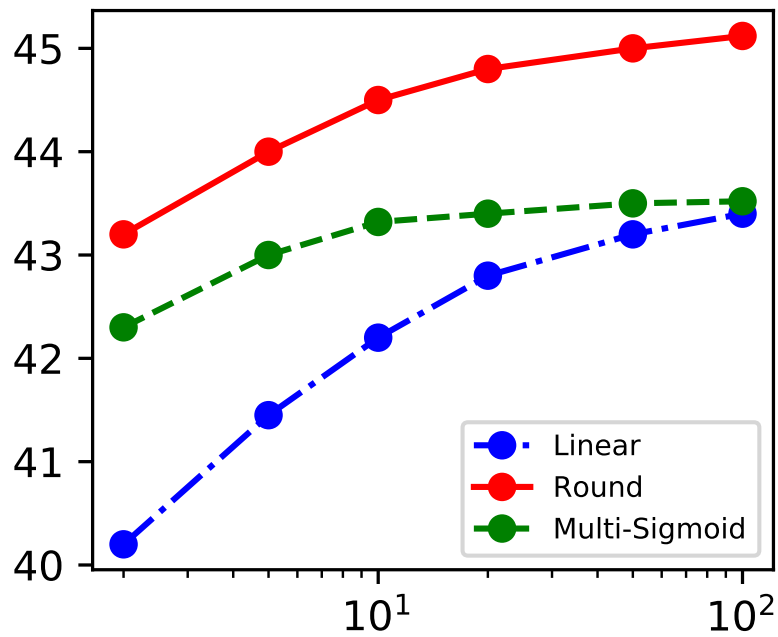| Proportion of Observations	10%	20%	30%	40%	50%	60%	70%	80%
Linear	1.73	1.06	0.97	0.90	0.85	0.85	0.87	0.83
Multi-Sigmoid	1.92	**0.53**	**0.48**	**0.42**	**0.39**	**0.38**	0.36	0.35
Multi-Sigmoid (Fixed τ)	1.96	1.54	1.37	1.32	1.29	1.28	1.25	1.23
Round	**1.49**	0.92	0.60	0.48	0.48	0.39	**0.30**	**0.28**
Round (Fixed τ)	2.44	1.50	1.50	1.43	1.36	1.39	1.44	1.34

Figure 4.4: Performance of percentage accuracy on smallnetflix datasets, as k in increase.

appear very good in comparison with other methods, while our Round approach result suffer from existence of noise in the dataset as before.
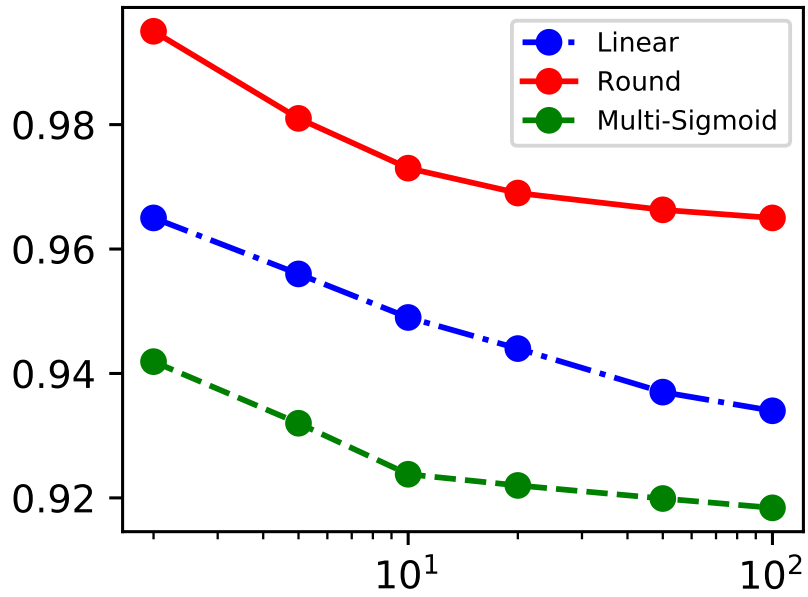
Figure 4.5: Performance of RMSE on smallnetflix datasets, as k in increase.
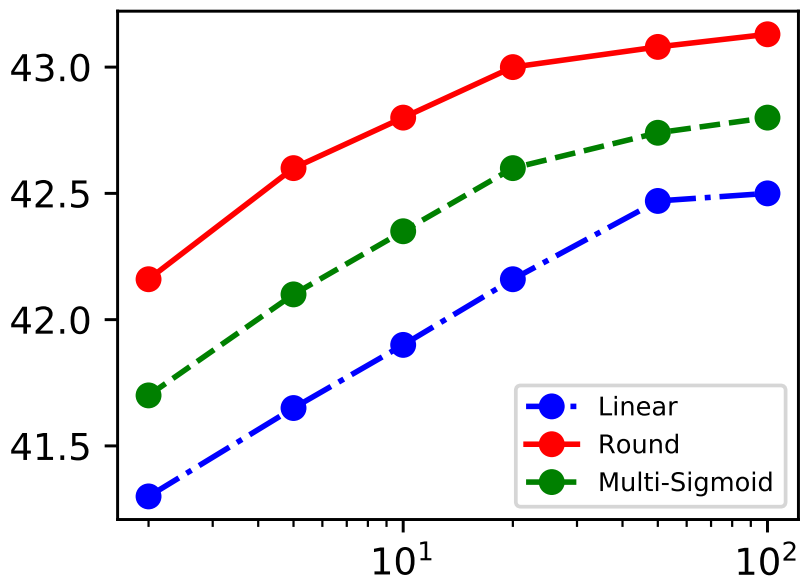

Figure 4.6: Performance of percentage accuracy on movielens datasets, as k in increased

Table 4.4: RMSE on Movielens-100k for a variety of models with different low-rank approximation (k).

Models	Low-rank approximation	RMSE
APG [14]	k=70	1.037
AIS-Impute [14]	k=70	1.037
CWOCFI [18]	k=10	1.01
our Round	k=10	1.007
our Linear	k=10	0.995
UCMF [31]	-	0.948
our Multi-sigmoid	k=10	0.928
SVDPlusPlus [8]	k=10	0.911
SIAFactorModel [8]	k=10	0.908
GG [15]	k=30	0.907

Chapter 5

Related Work

There is a rich literature on matrix factorization and its applications. To date, a number of link functions have been used, along with different losses for each, however here we are first to focus on expressive capabilities of these link functions, in particular of the ordinal-valued matrices [29, 12, 25, 30]. [23] addressed tensor factorization problem and showed improved performance when using a sigmoid link function. [19] introduced the concept of matrix factorization based on interval uncertainty, which results in a similar objective as our algorithm. However, not only is our proposed algorithm going beyond by updating the thresholds and supporting sigmoid-based smoothing, but we present results on the representation capabilities of the round-link function.

A number of methods have approached matrix factorization from a probabilistic view, primarily describing solutions when faced with different forms of noise, resulting, interestingly, in link functions as well. [4] introduced a generalization of PCA method to loss function for non real-valued data, such as binary-valued. [27] focused on Bayesian treatment of probabilistic matrix factorization, identifying the appropriate priors to

encode various *link* functions. On the other hand, [16] have analyzed non-linear matrix factorization based on Gaussian process and used SGD to optimize their model. However, these approaches do not explicitly investigate the representation capabilities, in particular, the significant difference in *rank* when link functions are taken into account.

Sign-rank and its properties have been studied by [22, 3, 5], and more recently, [20] provides in-depth analysis of round-rank. Although these have some similarity to GRR, sign-rank and round-rank are limited to binary matrices, while GRR is more suitable for most practical applications, and further, we present extension of their results in this work that apply to round-rank as well. Since we can view matrix factorization as a simple neural-network, research in understanding the complexity of neural networks [10], in particular with rectifier units [24], is relevant, however the results differ significantly in the aspects of representation we focus on.

Chapter 6

Conclusions and Future Work

In this work, we demonstrated the expressive power of using link functions for matrix factorization, specifically the generalized round-rank (GRR) for ordinal-value matrices. We show that not only are there full-rank matrices that are low GRR, but further, that these matrices cannot even be approximated by low linear factorization. Furthermore, we provide uniqueness conditions of this formulation, and provide gradient descent-based algorithms to perform such a factorization. We present evaluation on synthetic and real-world datasets that demonstrate that GRR-based factorization works significantly better than linear factorization: converging faster while requiring fewer observations. In future work, we will investigate theoretical properties of our optimization algorithm, in particular explore convex relaxations to obtain convergence and analyze sample complexity. We are interested in the connection of link-rank with different probabilistic interpretations, in particular, robustness to noise. Finally, we are also interested in practical applications of these ideas to different link functions and domains.

Bibliography

[1] N. Alon, T. Lee, A. Shraibman, and S. Vempala. The approximate rank of a matrix and its algorithmic applications: Approximate rank. In *Symposium on Theory of Computing (STOC)*, pages 675–684, 2013.

[2] M.-F. Balcan, Y. Liang, D. P. Woodruff, and H. Zhang. Optimal sample complexity for matrix completion and related problems via ℓ_2-regularization. *arXiv preprint arXiv:1704.08683*, 2017.

[3] G. Bouchard, S. Singh, and T. Trouillon. On approximate reasoning capabilities of low-rank vector spaces. In *AAAI Spring Syposium on Knowledge Representation and Reasoning (KRR): Integrating Symbolic and Neural Approaches*, 2015.

[4] M. Collins, S. Dasgupta, and R. E. Schapire. A generalization of principal components analysis to the exponential family. In *Neural Information Processing Systems (NIPS)*, pages 617–624, 2001.

[5] M. A. Davenport, Y. Plan, E. van den Berg, and M. Wootters. 1-bit matrix completion. *Information and Inference*, 3(3):189–223, 2014.

[6] C. De Sa, K. Olukotun, and C. Ré. Global convergence of stochastic gradient descent for some non-convex matrix problems. *arXiv preprint arXiv:1411.1134*, 2014.

[7] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[8] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme. Mymedialite: a free recommender system library. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 305–308. ACM, 2011.

[9] M. Hardt. Understanding alternating minimization for matrix completion. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 651–660. IEEE, 2014.

[10] G.-B. Huang. Learning capability and storage capacity of two-hidden-layer feedforward networks. *IEEE Transactions on Neural Networks*, 14(2):274–281, 2003.

[11] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674. ACM, 2013.

[12] Y. Koren and J. Sill. Ordrec: an ordinal model for predicting personalized item rating distributions. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 117–124. ACM, 2011.

[13] A. Krishnamurthy and A. Singh. Low-rank matrix and tensor completion via adaptive sampling. In *Advances in Neural Information Processing Systems*, pages 836–844, 2013.

[14] Q. Y. J. T. Kwok. Accelerated inexact soft-impute for fast large-scale matrix completion. 2015.

[15] B. Lakshminarayanan, G. Bouchard, and C. Archambeau. Robust bayesian matrix factorisation. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 425–433, 2011.

[16] N. D. Lawrence and R. Urtasun. Non-linear matrix factorization with gaussian processes. In *International Conference on Machine Learning (ICML)*, pages 601–608. ACM, 2009.

[17] M. M. Lin, B. Dong, and M. T. Chu. Integer matrix factorization and its application. *preprint*, 2009.

[18] J. Lu, S. Hoi, and J. Wang. Second order online collaborative filtering. In *Asian Conference on Machine Learning*, pages 325–340, 2013.

[19] J. Mareček, P. Richtárik, and M. Takáč. Matrix completion under interval uncertainty. *European Journal of Operational Research*, 256(1):35–43, 2017.

[20] S. Neumann. *On Some Problems of Rounding Rank*. PhD thesis, Universität des Saarlandes Saarbrücken, 2015.

[21] S. Neumann, R. Gemulla, and P. Miettinen. What you will gain by rounding: Theory and algorithms for rounding rank. In *International Conference on Data Mining (ICDM)*, 2016.

[22] M. Nickel, X. Jiang, and V. Tresp. Reducing the rank in relational factorization models by including observable patterns. In *Neural Information Processing Systems (NIPS)*, pages 1179–1187, 2014.

[23] M. Nickel and V. Tresp. Logistic tensor factorization for multi-relational data. In *ICML Workshop - Structured Learning: Inferring Graphs from Structured and Unstructured Inputs (SLG)*, 2013.

[24] X. Pan and V. Srikumar. Expressiveness of rectifier networks. In *International Conference on Machine Learning*, pages 2427–2435, 2016.

[25] U. Paquet, B. Thomson, and O. Winther. A hierarchical model for ordinal matrix factorization. *Statistics and Computing*, 22(4):945–957, 2012.

[26] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin. Relation extraction with matrix factorization and universal schemas. In *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT)*, pages 74–84, 2013.

[27] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *International Conference on Machine Learning (ICML)*, volume 25, 2008.

[28] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, volume 20, 2008.

[29] A. P. Singh and G. J. Gordon. A unified view of matrix factorization models. In *Machine Learning and Knowledge Discovery in Databases*, pages 358–373. Springer, 2008.

[30] M. Udell, C. Horn, R. Zadeh, and S. Boyd. Generalized low rank models. *arXiv preprint arXiv:1410.0342 (Working manuscript)*, 2014.

[31] C.-X. Zhang, Z.-K. Zhang, L. Yu, C. Liu, H. Liu, and X.-Y. Yan. Information filtering via collaborative user clustering modeling. *Physica A: Statistical Mechanics and its Applications*, 396:195–203, 2014.

Appendices

A   Complementary Lemmas on GRR

Lemma A.1. *For matrices $A, B \in \{0, ..., N\}^{n \times m}$:*

$$GRR_{\tau_1,...\tau_N}(A) \leq min(n, m) \tag{A.1}$$

$$GRR_{\tau_1,...\tau_N}(A) = GRR_{\tau_1,...\tau_N}(A^T) \tag{A.2}$$

$$GRR_{\tau_1,...\tau_N}(A + B) \leq GRR_{\tau_1,...\tau_N}(A) + GRR_{\tau_1,...\tau_N}(B) \tag{A.3}$$

Where $+$ is in the real numbers and $A + B \in \{0, ..., N\}^{n \times m}$.

Proof. According to definition of GRR and the fact that if $A = \mathrm{GRF}(C)$ then $r(C) \leq min(n, m)$ we can conclude the first inequality. Furthermore, Since we know for any matrix C, $r(C) = r(C^T)$ and use the fact that if $A = \mathrm{GRF}(C)$ then $A^T = \mathrm{GRF}(C^t)$ we can show the second inequality as well. And the third inequality is the direct result of following famous inequality:

$$r(A + b) \leq r(A) + r(B) \tag{A.4}$$

43

□

Lemma A.2. *the following decomposition holds for Generalized Round function:*

$$GRF_{\tau_1,...,\tau_N}(A) = \sum_{i=1}^{N} Round_{\tau_i}(A) \tag{A.5}$$

Proof. Base on definition of Round Function $\sum_{i=1}^{N} Round_{\tau_i}(A)$, counts the number of thresholds which are smaller than A, and this number is clearly equal to $GRF_{\tau_1,...,\tau_N}(A)$.

□

Lemma A.3. *For any arbitrary subset of thresholds $T = \{\tau_{i_1}, ..., \tau_{i_r}\}$:*

$$GRR_{\tau_1,...\tau_N}(A) \geq GRR_T(\bar{A}) \tag{A.6}$$

Where $\bar{A}$ attained by the following transformation in matrix A:

$$\bar{A} = [b_{ij}]_{n \times m} \tag{A.7}$$

$$b_{ij} = \begin{cases} 0, & \text{if } a_{ij} \in \{0,..,i_1-1\} \\ 1, & \text{if } a_{ij} \in \{i_1,..,i_2-1\} \\ \vdots \\ r-1, & \text{if } a_{ij} \in \{i_r,..,N-1\} \end{cases} \tag{A.8}$$

Proof. We define $\mathcal{B}$ and $\bar{\mathcal{B}}$ as follows:

$$\mathcal{B} = \{B | \text{GRF}_{\tau_1,...,\tau_N}(B) = A\} \tag{A.9}$$

$$\bar{\mathcal{B}} = \{\bar{B} | \text{GRF}_T(\bar{B}) = A\} \tag{A.10}$$

In result for any $B \in \mathcal{B}$, it is clear that $B \in \bar{\mathcal{B}}$ □

Lemma A.4. *Following inequality holds for GRR:*

$$GRR_{\tau_1,...,\tau_N}(A) \leq GRR_{\tau_1,...,\tau_N,\tau_{N+1}}(A) \tag{A.11}$$

Proof. Similar to previous Lemma, if we define $\mathcal{B}$ and $\bar{\mathcal{B}}$ as follows:

$$\mathcal{B} = \{B | \mathrm{GRF}_{\tau_1,...,\tau_N}(B) = A\} \tag{A.12}$$

$$\bar{\mathcal{B}} = \{\bar{B} | \mathrm{GRF}_{\tau_1,...,\tau_N,\tau_{N+1}}(\bar{B}) = A\} \tag{A.13}$$

Then it is clear that for any $\bar{B} \in \bar{\mathcal{B}}$, we have $\bar{B} \in \mathcal{B}$ $\qquad \square$

Lemma A.5. *Lets define the function $F : R^N \to N$ as follows:*

$$F(\tau_1, ..., \tau_N) = GRR_{\tau_1,...,\tau_N}(A) \tag{A.14}$$

Where A is a fix matrix in $\{0, ..., N\}^{n \times m}$. Then we have the following inequality:

$$F((\tau_1 + \tau_1')/2, ..., \tau_N) \leq F(\tau_1, ..., \tau_N) + F(\tau_1', ..., \tau_N) \tag{A.15}$$

Proof. We define $\mathcal{B}$, $\mathcal{B}'$ and $\bar{\mathcal{B}}$ as follows:

$$\mathcal{B} = \{B | \mathrm{GRF}_{\tau_1,...,\tau_N}(B) = A\} \tag{A.16}$$

$$\mathcal{B}' = \{B' | \mathrm{GRF}_{\tau_1',...,\tau_N}(B') = A\} \tag{A.17}$$

$$\bar{\mathcal{B}} = \{\bar{B} | \mathrm{GRF}_{(\tau_1+\tau_1')/2,...,\tau_N}(\bar{B}) = A\} \tag{A.18}$$

Accordingly, for any $B \in \mathcal{B}$ and $B' \in \mathcal{B}'$ we know $\frac{B+B'}{2} \in \bar{\mathcal{B}}$. Furthermore, since $\mathrm{r}(\frac{B+B'}{2}) = \mathrm{r}(B + B')$ and $\mathrm{r}(B + B') \leq \mathrm{r}(B) + \mathrm{r}(B')$ we can clearly prove the inequality.

$\qquad \square$

Lemma A.6. *We have the following inequality:*

$$F(\tau_1 + \tau_1', ..., \tau_N + \tau_N') \leq F(\tau_1, ..., \tau_N) + F(\tau_1', ..., \tau_N') \tag{A.19}$$

Proof. Similar to previous Lemma, if we define $\mathcal{B}$, $\mathcal{B}'$ and $\bar{\mathcal{B}}$ as follows:

$$\mathcal{B} = \{B | \mathrm{GRF}_{\tau_1, ..., \tau_N}(B) = A\} \tag{A.20}$$

$$\mathcal{B}' = \{B' | \mathrm{GRF}_{\tau_1', ..., \tau_N'}(B') = A\} \tag{A.21}$$

$$\bar{\mathcal{B}} = \{\bar{B} | \mathrm{GRF}_{\tau_1 + \tau_1', ..., \tau_N + \tau_N'}(\bar{B}) = A\} \tag{A.22}$$

For any $B \in \mathcal{B}$ and $B' \in \mathcal{B}'$ we know $B + B' \in \bar{\mathcal{B}}$. And since $\mathrm{r}(B + B') \leq \mathrm{r}(B) + \mathrm{r}(B')$ we can clearly prove the inequality. $\square$