Tensor Q-Rank: A New Data Dependent Tensor Rank

Hao Kong, Zhouchen Lin
Key Lab. of Machine Perception (MoE),
School of EECS, Peking University
{konghao, zlin}@pku.edu.cn

Abstract

Recently, the *Tensor Nuclear Norm (TNN)* regularization based on t-SVD has been widely used in various low tubal-rank tensor recovery tasks. However, these models usually require smooth change of data along the third dimension to ensure their low rank structures. In this paper, we propose a new definition of tensor rank named *tensor Q-rank* by a column orthonormal matrix $\mathbf{Q}$, and further make $\mathbf{Q}$ data-dependent. With $\mathbf{Q}$ satisfying our orthogonal proximal constraint, the data tensor may have a more significant low tensor Q-rank structure than that of low tubal-rank structure. We also provide a corresponding envelope of our rank function and apply it to the low rank tensor completion problem. Then we give an effective algorithm and briefly analyze why our method works better than TNN based methods in the case of complex data with low sampling rate. Finally, experimental results on real-world datasets demonstrate the superiority of our proposed model in the tensor completion problem.

Introduction

With the development of data science, the multi-dimensional data structures are becoming more and more complex. The low-rank tensor recovery problem, which aims to recover a low-rank tensor from observed tensor, has also been extensively studied and applied. The problem can be formulated as the following model:

$$\min_{\mathcal{X}} \ \operatorname{rank}(\mathcal{X}), \quad s.t. \ \Psi(\mathcal{X}) = \mathcal{Y}, \tag{1}$$

where $\mathcal{Y}$ is the observed measurement by a linear operator $\Psi(\cdot)$ and $\mathcal{X}$ is the clean data. Generally, it is difficult to solve Eq. (1) directly. Different rank definitions correspond to different models. Based on different definitions of tensor singular values, various tensor nuclear norms are proposed as the rank surrogates (Liu et al. 2013; Friedland and Lim 2018; Kilmer and Martin 2011). Liu et al. (Liu et al. 2013) define a kind of tensor nuclear norm named SNN (Sum of Nuclear Norm) based on the Tucker decomposition (Tucker 1966). However, Paredes et al. (Romera-Paredes and Pontil 2013) point out that SNN is not the tightest convex relaxation of the Tucker-n-rank (Kruskal 1989), and is actually an overlap regularization of it. (Tomioka, Hayashi, and Kashima 2010; Tomioka and Suzuki 2013; Wimalawarne, Sugiyama, and Tomioka 2014) also proposed a new regularizer named Latent Trace Norm to better approximate the tensor rank function.

Due to unfolding the tensor directly along each dimension, the information utilization of SNN based model is insufficient. Kilmer et al. (Kilmer and Martin 2011) propose a tensor decomposition named t-SVD, and Zhang et al. (Zhang et al. 2014) give a definition of the nuclear norm corresponding to t-SVD, i.e., Tensor Nuclear Norm (TNN). But using a fixed Fourier transform matrix $\mathbf{F}$ may bring some limitations.

Firstly, TNN needs to implement Singular Value Decomposition (SVD) in the complex field $\mathbb{C}$, which is slightly slower than that in the real field $\mathbb{R}$. Besides, the experiments in related papers (Zhang et al. 2014; Lu et al. 2018; Zhou et al. 2018; Kong, Xie, and Lin 2018) are usually based on some special dataset which have smooth change along the third dimension, such as RGB images and short videos. Those non-smooth data may increase the number of non-zero tensor singular values (Kilmer and Martin 2011; Zhang et al. 2014), weakening the significance of low rank structure. Since tensor multi-rank (Zhang et al. 2014) is actually the rank of each projection matrix on different Fourier basis, the non-smooth change along the third dimension may lead to large singular values appearing on high frequency projection matrix slices. Meanwhile, we also find that SNN based methods always perform better than TNN when dealing with non-smooth data. Therefore, we consider that a better set of data-dependent projection bases can effectively overcome these limitations.

For the cases mentioned above, we point out that: **(1)** each element of tensor multi-rank is actually the rank of each frontal slice, which is obtained by the Tucker Product (Tucker 1966) of the original tensor and the Fourier transform matrix $\mathbf{F}$ along the third-dimension; **(2)** each singular value of SNN is the Frobenius norm of each frontal slice, which is obtained by the Tucker Product (Tucker 1966) of the original tensor and the singular matrix $\mathbf{V}_{(dim)}$ along each dimension.

Considering the data dependence of $\mathbf{V}_{(dim)}$ in SNN, we hope to find a data-dependent $\mathbf{Q}$ instead of $\mathbf{F}$, which can reduce the number of non-zero singular values of each projected slices. The matrix inequality $\|\cdot\|_2 \leq \|\cdot\|_F$ (spectral

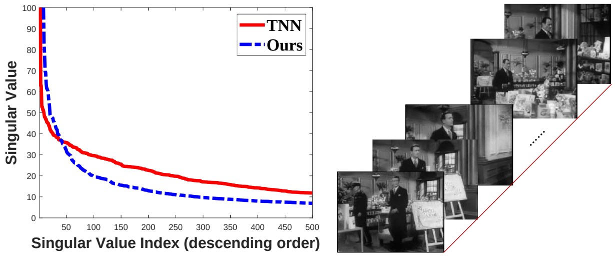

Figure 1: Compare the two different low rank structures between our proposed regularization and TNN regularization in non-smooth video data. **Left:** the first 500 sorted singular values by TNN regularization (divided by $\sqrt{n_3}$) and ours. **Right:** the short video with background changes.

norm and Frobenius norm, respectively) implies that, the closer $\|\cdot\|_F$ is to zero, the more singular values close to zero, which will lead to a more significant low rank structure. It should be pointed out that our proposed method seems to be similar to SNN, but our definition of singular value is quite different from it, which makes our method maintain the tensor internal structure better. In summary, our main contributions include:

- We generalize the Fourier transform matrix $\mathbf{F}$ to any fixed real orthonormal matrix $\mathbf{Q}$ and propose a new definition of tensor rank, named tensor Q-rank and denoted as rank_Q. The low rank tensor recovery problem can be rewritten as:
$$\min_{\mathcal{X}} \ \text{rank}_Q(\mathcal{X}), \quad s.t. \ \Psi(\mathcal{X}) = \mathcal{Y}. \qquad (2)$$
- We also provide an envelope of the tensor Q-rank within an appropriate region, named tensor Q-nuclear norm, as a regularizer. We further set $\mathbf{Q}$ to be a learnable variable w.r.t. the data to make the data have a more significant low rank structure under our tensor Q-rank. We study the Orthogonal Proximal Constraint as a heuristic relation between $\mathbf{Q}$ and the data tensor, and analyze the advantages of this choice. **Figure** 1 shows an example with background changing video data that, under this constraint, our low rank structure is more significant than that of TNN.
- Finally, we apply the proposed regularizer of adaptive $\mathbf{Q}$ to the tensor completion problem. As for the special case that $\mathbf{Q}$ is fixed, we give a complete proof of the convergence for corresponding completion algorithm and the performance guarantee for exact completion.

Notations and Preliminaries

We introduce some notations and necessary definitions which will be used later. Tensors are represented by uppercase curlycue letters, e.g., $\mathcal{T}$. Matrices are represented by boldface uppercase letters, e.g., $\mathbf{M}$. Vectors are represented by boldface lowercase letters, e.g., $\mathbf{v}$. Scalars are represented by lowercase letters, e.g., s. Given a 3-order tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we use $\mathbf{T}^{(k)}$ to represent its k-th frontal slice $\mathcal{T}(:,:,k)$. Its (i,j,k)-th entry is represented as $\mathcal{T}_{ijk}$. $\mathbf{T}_{(k)}$ denotes unfolding the tensor $\mathcal{T}$ along the k-th dimension (Kolda and Bader 2009). $\sigma_i(\mathbf{X})$ denotes the i-th singular value of matrix $\mathbf{X}$. $\mathbf{X}^+$ denotes the pseudo-inverse matrix of $\mathbf{X}$. Due to limited space, for the definitions of $\mathcal{P}_{\mathcal{T}}$ (Lu et al. 2016), Tucker product (Tucker 1966), t-product (Kilmer and Martin 2011), and so on, please refer to our Supplementary Materials.

Motivation

In the tensor completion task, we found that when dealing with some non-smooth data (which means that matrix data are arranged disorderly along the third dimension), Tensor Nuclear Norm (TNN) based methods usually perform worse than the cases with smooth data. At the same time, Sum of Nuclear Norm (SNN) based methods are almost unaffected by the smoothness of data. However, for most dataset, TNN based methods are much better than SNN based methods.

Our motivation is that we can combine the advantages of these two norm based methods, and make new methods more robust to data smoothness and maintain good performance similar to TNN. We consider that **(1):** the robustness along the third dimension of SNN based methods comes from its mode-3 regularizer, which assumes that the dimension of $X_{(3)}$'s column subspace (projected by its right singular matrix $\mathbf{V}_{(3)}$) is quite small. But the right singular matrix is instead by a fixed Fourier matrix in TNN. **(2):** the excellent performance of TNN based methods comes from its definition of tensor singular value. In the mode-3 regularizer of SNN, the singular value of mode-3 unfolding matrix is too simple to approximate the true subspace. In summary, we combine the data-dependent orthogonal matrix of SNN and the reasonable definition of tensor singular value of TNN and then propose our tensor Q-norm.

Main Result

Tensor Q-rank

For a given tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and a Fourier transform matrix $\mathbf{F} \in \mathbb{C}^{n_3 \times n_3}$, if we use $\mathbf{G}^{(i)}$ to represent the i-th frontal slice of tensor $\mathcal{G}$, then the tensor multi-rank and tensor nuclear norm (TNN) of $\mathcal{T}$ can be formulated by Tucker product as follows:

$$\text{rank}_\text{m} := \left\{(r_1,\ldots,r_{n_3}) \middle| r_i = \text{rank}\left(\mathbf{G}^{(i)}\right), \mathcal{G} = \mathcal{T} \times_3 \mathbf{F}\right\}, (3)$$
$$\|\mathcal{T}\|_* := \frac{1}{n_3}\sum_{i=1}^{n_3}\left\|\mathbf{G}^{(i)}\right\|_*, \quad \text{where } \mathcal{G} = \mathcal{T} \times_3 \mathbf{F}. \qquad (4)$$

We extend the multiplier of Fourier transform matrix $\mathbf{F} \in \mathbb{C}^{n_3 \times n_3}$ to a general real orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$ and then propose a new definition of tensor rank named Tensor Q-rank.

Definition 1. ***(Tensor Q-rank)*** *Given a tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and a fixed real orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$, the tensor Q-rank of $\mathcal{T}$ is defined as the following:*

$$\text{rank}_Q(\mathcal{T}) := \sum_{i=1}^{n_3} \text{rank}\left(\mathbf{G}^{(i)}\right), \textit{where } \mathcal{G} = \mathcal{T} \times_3 \mathbf{Q}. \quad (5)$$

Generally in the low-rank recovery models, it is quite difficult to minimize the rank function directly. Therefore, some auxiliary norms are needed to relax the rank function.

When an orthogonal matrix $\mathbf{Q}$ is fixed, the corresponding tensor spectral norm and tensor nuclear norm of $\mathcal{T}$ can also be given.

Definition 2. ***(Tensor Q-spectral norm)*** *Given a tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and a fixed real orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$, the tensor Q-spectral norm of $\mathcal{T}$ is defined as the following:*

$$\|\mathcal{T}\|_{Q,\sigma} := \max_{i} \left\{ \left\| \mathbf{G}^{(i)} \right\|_{\sigma} \middle| \mathcal{G} = \mathcal{T} \times_3 \mathbf{Q} \right\}. \tag{6}$$

Definition 3. ***(Tensor Q-nuclear norm)*** *Given a tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and a fixed real orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$, the tensor Q-nuclear norm of $\mathcal{T}$ is defined as the following:*

$$\|\mathcal{T}\|_{Q,*} := \sum_{i=1}^{n_3} \left\| \mathbf{G}^{(i)} \right\|_{*}, \quad \textit{where } \mathcal{G} = \mathcal{T} \times_3 \mathbf{Q}. \tag{7}$$

Moreover, with a fixed $\mathbf{Q}$, the convexity, duality, and envelope properties are all preserved.

Property 1. ***(Convexity)*** *Tensor Q-nuclear norm and Tensor Q-spectral norm are both convex.*

Property 2. ***(Duality)*** *Tensor Q-nuclear norm is the dual norm of Tensor Q-spectral norm, and vice versa.*

Property 3. ***(Convex Envelope)*** *Tensor Q-nuclear norm is the tightest convex envelope of the Tensor Q-rank within the unit ball of the Tensor Q-spectral norm.*

These three properties are very important in the low rank recovery theory. Property 3 implies that we can use the tensor Q-nuclear norm as a rank surrogate. That is to say, when the orthogonal matrix $\mathbf{Q}$ is given, we can replace the low tensor Q-rank model (2) with model (8) to recover the original tensor:

$$\min_{\mathcal{X}} \|\mathcal{X}\|_{Q,*}, \quad s.t.\ \Psi(\mathcal{X}) = \mathcal{Y}. \tag{8}$$

Orthogonal Proximal Constraint

In practical problems, the selection of $\mathbf{Q}$ often has a tremendous impact on the performance of the model (8). If $\mathbf{Q}$ is an identity matrix $\mathbf{I}$, it is equivalent to solving each frontal slice separately by the low rank matrix methods (Candès and Recht 2009). Or if $\mathbf{Q}$ is a Fourier transform matrix $\mathbf{F}$, it is equivalent to the TNN-based methods (Zhang et al. 2014; Lu et al. 2016).

For a given data $\mathcal{X}$, those $\mathbf{Q}$ that make $\operatorname{rank}_Q(\mathcal{X})$ lower usually make the recovery problem (8) easier (We will discuss this conclusion in Theorem 4). We can solve problem (8) by finding such a $\mathbf{Q}$ prophetically and then fixing it. However, it is usually difficult to estimate $\mathbf{Q}$ only by the observed data. Therefore, we consider making the selection of $\mathbf{Q}$ data-dependent. Fortunately, we can use a bilevel model to compute an adaptive $\mathbf{Q}$, and the following definition shows the details.

Definition 4. ***(Low Tensor Q-rank model with adaptive*** $\mathbf{Q}$***)*** *By setting the adaptive $\mathbf{Q}$ module as a low-level sub-problem, the low tensor Q-rank model (2) is transformed into the following:*

$$\begin{aligned} \min_{\mathcal{X},\mathbf{Q}} \quad & \operatorname{rank}_Q(\mathcal{X}), \\ s.t. \quad & \Psi(\mathcal{X}) = \mathcal{Y},\ \mathbf{Q} \in \operatorname*{argmin}_{\mathbf{Q}} f(\mathbf{Q},\mathcal{X}),\ \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_{n_3}. \end{aligned} \tag{9}$$

And the corresponding surrogate model (8) is also replaced by the following:

$$\begin{aligned} \min_{\mathcal{X},\mathbf{Q}} \quad & \|\mathcal{X}\|_{Q,*}, \\ s.t. \quad & \Psi(\mathcal{X}) = \mathcal{Y},\ \mathbf{Q} \in \operatorname*{argmin}_{\mathbf{Q}} f(\mathbf{Q},\mathcal{X}),\ \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_{n_3}. \end{aligned} \tag{10}$$

In Eqs. (9) and (10), the function $f(\cdot,\cdot)$ represents an arbitrary binary function and $\mathbf{Q}$ is an orthogonal matrix.

Within this framework, the orthogonal matrix $\mathbf{Q}$ is related to tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. And the constraint $\mathbf{Q} \in \operatorname{argmin}_{\mathbf{Q}} f(\mathbf{Q},\mathcal{X})$ should make $\operatorname{rank}_Q(\mathcal{X})$ as low as possible. As we analyzed in Introduction, there should be more "small" frontal slices of $\mathcal{X} \times_3 \mathbf{Q}$, whose Frobenius norms are close to 0.

Consider the case that $\mathbf{Q} \in \mathbb{R}^{n_3 \times r}$ is only a column orthonormal matrix: $\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_r$. Inspired by the greedy algorithm, with $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_r]$ and $\mathcal{G}_i = \mathcal{X} \times_3 \mathbf{q}_i \times_3 \mathbf{q}_i^\top$, we hope that $\mathcal{G}_1$ is the optimal "rank-one approximation" of $\mathcal{X}$, which means $\mathcal{G}_1$ has the largest Frobenius norm. Similarly, $\mathcal{G}_i$ has the largest Frobenius norm with $\mathcal{G}_i \perp \operatorname{span}\{\mathcal{G}_1, \ldots, \mathcal{G}_{i-1}\}$ in order to make the column orthonormal matrix $\mathbf{Q}$ extract more information from the data. After i goes from 1 to N, we obtain the expected column orthonormal matrix $\mathbf{Q}$, where $N = \min\{n_1 n_2, n_3\}$.

With the definition of mode-3 Tucker product, this procedure can be derived from the best rank-r approximation theory (De Lathauwer, De Moor, and Vandewalle 2000). Thus we set $f(\mathbf{Q},\mathcal{X}) = \|\mathcal{X} \times_3 \mathbf{Q} \times_3 \mathbf{Q}^\top - \mathcal{X}\|_F^2 = \|\mathcal{X} \times_3 (\mathbf{Q}\mathbf{Q}^\top) - \mathcal{X}\|_F^2$ in Definition 4. Together with $\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_r$, we named these constraints as **Orthogonal Proximal Constraint**.

Unfold the tensor $\mathcal{X}$ along the 3-rd dimension (Kolda and Bader 2009) to get a matrix $\mathbf{X}_{(3)} \in \mathbb{R}^{n_1 n_2 \times n_3}$, and then apply the Eckart-Young-Mirsky theorem (Eckart and Young 1936) to the problem $\mathbf{Q} = \operatorname{argmin}_{\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_r} \|\mathbf{X}_{(3)} \mathbf{Q}\mathbf{Q}^\top - \mathbf{X}_{(3)}\|_F^2$. As can be seen, $\mathbf{Q}$ is the first r right singular vectors of $\mathbf{X}_{(3)}$, and $\mathcal{X} \times_3 \mathbf{Q}$ is actually the first r principal component matrices of all frontal slices $\{\mathbf{X}^{(i)}\}_{i=1}^{n_3}$. For convenience, we use $\operatorname{PCA}(\mathcal{X}, 3, r)$ to represent this optimization problem: $\operatorname{PCA}(\mathcal{X}, 3, r) := \arg\min_{\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_r} \|\mathbf{X}_{(3)} \mathbf{Q}\mathbf{Q}^\top - \mathbf{X}_{(3)}\|_F^2$. There is a special case that if $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$ has full column rank, we let $\operatorname{PCA}(\mathcal{X}, 3, n_3)$ be the right singular matrix of $\mathbf{X}_{(3)}$. Then we have the following proposition:

Proposition 1. *Through the above deduction, if we choose $f(\mathbf{Q},\mathcal{X}) := \left\| \mathcal{X} \times_3 (\mathbf{Q}\mathbf{Q}^\top) - \mathcal{X} \right\|_F^2$ and let $\operatorname{PCA}(\mathcal{X}, 3, r) := \arg\min_{\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_r} \|\mathbf{X}_{(3)} \mathbf{Q}\mathbf{Q}^\top - \mathbf{X}_{(3)}\|_F^2$ be the operator to obtain the column orthonormal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times r}$, where $r = \min\{n_1 n_2, n_3\}$, then the models (9) and (10) can be abbreviated as follows:*

$$\min_{\mathcal{X}} \operatorname{rank}_Q(\mathcal{X}), \quad s.t.\ \Psi(\mathcal{X}) = \mathcal{Y},\ \mathbf{Q} = \operatorname{PCA}(\mathcal{X}, 3, r), \tag{11}$$

$$\min_{\mathcal{X}} \|\mathcal{X}\|_{Q,*}, \quad s.t.\ \Psi(\mathcal{X}) = \mathcal{Y},\ \mathbf{Q} = \operatorname{PCA}(\mathcal{X}, 3, r). \tag{12}$$

Remark 1. *$\mathbf{Q}$ can be selected in many ways, as long as $\mathbf{Q}$ is pseudo-invertible to make $\mathcal{X} = \mathcal{X} \times_3 \mathbf{Q} \times_3 \mathbf{Q}^{+}$ holds. For example, $f(\mathbf{Q},\mathcal{X}) = \operatorname{argmin}_{\mathbf{Q}} \|\mathcal{X}\|_{Q,*}$ is also acceptable* *but difficult to optimize. Our experiments shows the PSNR results of our models with a random* $\mathbf{Q}$ *and an Oracle* $\mathbf{Q}$.

In fact, this selection method of $\mathbf{Q}$ guarantees the low tensor Q-rank structure of data with high probability. Let $\mathcal{G} = \mathcal{X} \times_3 \mathbf{Q}$, and $\{\mathbf{G}^{(i)}\}_{i=1}^{r}$ denotes the frontal slices of $\mathcal{G}$, then the i-th singular value of $\mathbf{X}_{(3)}$ satisfies $\sigma_i(\mathbf{X}_{(3)}) = \|\mathbf{G}^{(i)}\|_F$. According to the distribution property of singular values (Sengupta and Mitra 1999), for general data matrix $\mathbf{X}_{(3)}$ (or random matrix), the first few $\sigma_i(\mathbf{X}_{(3)})$ are much larger than the others with high probability, and most of the rest singular values are close to 0. Therefore, the inequality $\frac{1}{n}\sum_{j=1}^{n}\sigma_j(\mathbf{G}^{(i)}) \leq \|\mathbf{G}^{(i)}\|_\sigma \leq \|\mathbf{G}^{(i)}\|_F = \sigma_i(\mathbf{X}_{(3)})$ can guarantee that most singular values $\sigma_j(\mathbf{G}^{(i)})$ are also close to 0 with high probability.

Now the question is whether the function $\|\mathcal{X}\|_{Q,*}$ in Eq. (12) is still an envelope of the rank function $\text{rank}_Q(\mathcal{X})$ in Eq. (11) within an appropriate region. The following theorem shows that even if $\|\mathcal{X}\|_{Q,*}$ is no longer a convex function in the bilevel framework (12) since $\mathbf{Q}$ is dependent on $\mathcal{X}$, we can still use it as a surrogate for a lower bound of $\text{rank}_Q(\mathcal{X})$ in Eq. (11).

Theorem 1. *Given a column orthonormal matrix* $\mathbf{Q} \in \mathbb{R}^{n_3\times r}$, *where* $r = \min\{n_1n_2, n_3\}$, *we use* $\text{rank}_{PCA}(\mathcal{X})$, $\|\mathcal{X}\|_{PCA,\sigma}$, *and* $\|\mathcal{X}\|_{PCA,*}$ *to abbreviate the corresponding concepts as follows:*

$$\text{rank}_{PCA}(\mathcal{X}) := \text{rank}_Q(\mathcal{X}), \textit{where } \mathbf{Q} = \text{PCA}(\mathcal{X},3,r), \quad (13)$$

$$\|\mathcal{X}\|_{PCA,\sigma} := \|\mathcal{X}\|_{Q,\sigma}, \quad \textit{where } \mathbf{Q} = \text{PCA}(\mathcal{X},3,r), \quad (14)$$

$$\|\mathcal{X}\|_{PCA,*} := \|\mathcal{X}\|_{Q,*}, \quad \textit{where } \mathbf{Q} = \text{PCA}(\mathcal{X},3,r). \quad (15)$$

Then within the region of $\mathcal{D} = \{\mathcal{X} \mid \|\mathcal{X}\|_{PCA,\sigma} \leq 1\}$, *the inequality* $\|\mathcal{X}\|_{PCA,*} \leq \text{rank}_{PCA}(\mathcal{X})$ *holds. Moreover, for every fixed* $\mathbf{Q}$, *let* $\mathcal{S}_Q$ *denote the space* $\{\mathcal{X} \mid \mathbf{Q} \in \text{PCA}(\mathcal{X},3,r)\}$. *Then Property 3 indicates that* $\|\mathcal{X}\|_{PCA,*}$ *is still the tightest convex envelope of* $\text{rank}_{PCA}(\mathcal{X})$ *in* $\mathcal{S}_Q \cap \mathcal{D}$.

Remark 2. *For any* $\mathbf{Q} \in \text{argmin}_{\mathbf{Q}^\top\mathbf{Q}=\mathbf{I}_r} f(\mathbf{Q},\mathcal{X})$ *in Definition 4, the corresponding conclusion also holds as long as* $\mathcal{X} \times_3 (\mathbf{Q}\mathbf{Q}^\top) = \mathcal{X}$. *That is to say,* $\|\mathcal{X}\|_{Q,*} \leq$ *rank*$_Q(\mathcal{X})$ *holds within the region* $\{\mathcal{X} \mid \|\mathcal{X}\|_{Q,\sigma} \leq 1\}$. *Therefore, Eq. (10) can be used as a relaxed model for Eq. (9).*

Theorem 1 shows that though $\|\mathcal{X}\|_{PCA,*}$ could be nonconvex, its function value is always below $\text{rank}_{PCA}(\mathcal{X})$. Therefore, model (12) can be regarded as a reasonable low rank tensor recovery model. Notice that it is actually a bilevel optimization problem.

Application to Tensor Completion

Model

In the 3-order tensor completion task, Ω is an index set consisting of the indices $\{(i,j,k)\}$ which can be observed, and the operator Ψ in Eqs. (11) and (12) is replaced by an orthogonal projection operator $\mathcal{P}_\Omega$, where $\mathcal{P}_\Omega(\mathcal{X}_{ijk}) = \mathcal{X}_{ijk}$ if$(i,j,k) \in \Omega$ and 0 otherwise. The observed tensor $\mathcal{Y}$ satisfies $\mathcal{Y} = \mathcal{P}_\Omega(\mathcal{Y})$. Then the tensor completion model based on our assumption is given by:

$$\min_{\mathcal{X}} \|\mathcal{X}\|_{Q,*}, \quad s.t.\ \mathcal{P}_\Omega(\mathcal{X}) = \mathcal{Y},\ \mathbf{Q} = \text{PCA}(\mathcal{X},3,r), \quad (16)$$

where $\mathcal{X}$ is the tensor that has low rank structure, and $\mathbf{Q} \in \mathbb{R}^{n_3\times r}$ is an column orthonormal matrix with $r = \min\{n_1n_2, n_3\}$. To solve the model by ADMM based method (Lu et al. 2017), we introduce an intermediate tensor $\mathcal{E}$ to separate $\mathcal{X}$ from $\mathcal{P}_\Omega(\cdot)$. Let $\mathcal{E} = \mathcal{P}_\Omega(\mathcal{X}) - \mathcal{X}$, then $\mathcal{P}_\Omega(\mathcal{X}) = \mathcal{Y}$ is translated to $\mathcal{X}+\mathcal{E} = \mathcal{Y}$, $\mathcal{P}_\Omega(\mathcal{E}) = \mathcal{O}$, where $\mathcal{O}$ is an all-zero tensor. Then we get the following model:

$$\begin{aligned} &\min_{\mathcal{X},\mathcal{E},\mathbf{Q}} \|\mathcal{X}\|_{Q,*}, \\ &s.t.\ \mathcal{X}+\mathcal{E} = \mathcal{Y},\ \mathcal{P}_\Omega(\mathcal{E}) = \mathcal{O},\ \mathbf{Q} = \text{PCA}(\mathcal{X},3,r). \end{aligned} \quad (17)$$

Optimization

Since $\mathbf{Q}$ is dependent on $\mathcal{X}$, it is difficult to solve the model (17) w.r.t. $\{\mathcal{X},\mathbf{Q}\}$ directly. Here we adopt the idea of alternating minimization to solve $\mathcal{X}$ and $\mathbf{Q}$ alternately. We separate the sub-problem of solving $\mathbf{Q}$ as a sub-step in every K-iteration, and then update $\mathcal{X}$ with a fixed $\mathbf{Q}$ by the ADMM method (Lu et al. 2017; Lu et al. 2018). The partial augmented Lagrangian function of Eq. (17) is

$$L(\mathcal{X},\mathcal{E},\mathcal{Z},\mu) = \|\mathcal{X}\|_{Q,*} + \langle\mathcal{Z}, \mathcal{Y}-\mathcal{X}-\mathcal{E}\rangle + \frac{\mu}{2}\|\mathcal{Y}-\mathcal{X}-\mathcal{E}\|_F^2, \quad (18)$$

where $\mathcal{Z}$ is the dual variable and $\mu > 0$ is the penalty parameter. Then we can update each component $\mathbf{Q}$, $\mathcal{X}$, $\mathcal{E}$, and $\mathcal{Z}$ alternately. Algorithms 1 and 2 show the details about the optimization methods to Eq. (17). Note that there is one operator $\mathbf{Prox}$ in the sub-step of updating $\mathcal{X}$ as follows:

$$\mathcal{X} = \mathbf{Prox}_{\lambda,\|\cdot\|_{Q,*}}(\mathcal{T}) := \operatorname*{argmin}_{\mathcal{X}} \lambda\|\mathcal{X}\|_{Q,*} + \frac{1}{2}\|\mathcal{X}-\mathcal{T}\|_F^2, \quad (19)$$

where $\mathbf{Q} \in \mathbb{R}^{n_3\times r}$ is a given column orthonormal matrix and $\|\mathcal{X}\|_{Q,*}$ is the tensor Q-nuclear norm of $\mathcal{X}$ which is defined in Eq. (7). Algorithm 2 shows the details of solving this operator.

In Eq. (20), with the convergence of iteration, there will be $\lim\limits_{k\to\infty}(\mathcal{Y} - \mathcal{E}_{k-1} + \frac{\mathcal{Z}_{k-1}}{\mu_{k-1}}) = \mathcal{X}_k$. Naturally, $\lim\limits_{k\to\infty} Q_k = \text{PCA}(\mathcal{X}_k,3,r)$ holds, which corresponds to the constraints of the original problem (16). As for the case that $\mathbf{Q}$ is a fixed orthogonal matrix, the corresponding optimization algorithm is provided in Supplementary Materials.

Convergence and Performance Analyses

Convergence: For the models (16) or (17), it is hard to analyze the convergence of the corresponding optimization method directly. The constraint on $\mathbf{Q}$ is non-linear and the objective function is essentially non-convex, which increase the difficulty of analysis. However, the conclusions of (Lu et al. 2017; Lin, Liu, and Li 2015; Xu and Yin 2015; Lin, Liu, and Su 2011; Absil, Mahony, and Sepulchre 2009) guarantee the convergence to some extent. In practical applications, we can fix $\mathbf{Q}_k = \mathbf{Q}$ in every K iterations and then use Theorem 3 to obtain a conditional optimal solution under $\mathbf{Q}$ with proper K and ρ.

Theorem 2. *With a proper* ρ *and* $r = \min\{n_1n_2, n_3\}$ *in Algorithm 1, the sequence* $\{\mathcal{X}_k,\mathcal{E}_k,\mathcal{Z}_k\}$ *generated by Algorithm 1 is convergent, and* $\|\mathcal{Y}-\mathcal{X}_k-\mathcal{E}_k\|_\infty \to 0$, $\mathbf{Q}_k \to$ *PCA*$(\mathcal{X}_k,3,r)$.

Algorithm 1 Solving the problem (17) by ADMM.

```
Input: Observation samples $\mathcal{Y}_{ijk}$, $(i,j,k) \in \Omega$, of tensor $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$.
Initialize: $\mathcal{X}_0,\ \mathcal{E}_0,\ \mathcal{Z}_0,\ \mathbf{Q}_0 \in \mathbb{R}^{n_3 \times r}$. Parameters $k = 1,\ \rho > 1,\ \mu_0,\ \mu_{max},\ \varepsilon,\ K$.
While not converge do
  1. Update $\mathbf{Q}_k$ by
       $\mathbf{Q}_k = \begin{cases} \mathbf{Q}_{k-1}, & k \bmod K \neq 1, \\ \text{PCA}\left(\mathcal{Y} - \mathcal{E}_{k-1} + \frac{\mathcal{Z}_{k-1}}{\mu_{k-1}}, 3, r\right), & k \bmod K = 1. \end{cases}$   (20)
  2. Update $\mathcal{X}_k$ by
       $\mathcal{X}_k = \mathbf{Prox}_{\mu_{k-1}^{-1}, \|\cdot\|_{Q_k,*}} \left(\mathcal{Y} - \mathcal{E}_{k-1} + \frac{\mathcal{Z}_{k-1}}{\mu_{k-1}}\right).$   (21)
  3. Update $\mathcal{E}_k$ by
       $\mathcal{E}_k = \mathcal{P}_{\Omega^{\complement}} \left(\mathcal{Y} - \mathcal{X}_k + \frac{\mathcal{Z}_{k-1}}{\mu_{k-1}}\right),$   (22)
     where $\Omega^{\complement}$ is the complement of $\Omega$.
  4. Update the dual variable $\mathcal{Z}_k$ by
       $\mathcal{Z}_k = \mathcal{Z}_{k-1} + \mu_{k-1}\left(\mathcal{Y} - \mathcal{X}_k - \mathcal{E}_k\right).$   (23)
  5. Update $\mu_k$ by
       $\mu_k = \min\{\rho\mu_{k-1}, \mu_{max}\}.$   (24)
  6. Check the convergence condition: $\|\mathcal{X}_k - \mathcal{X}_{k-1}\|_\infty \leq \varepsilon$, $\|\mathcal{E}_k - \mathcal{E}_{k-1}\|_\infty \leq \varepsilon$, and $\|\mathcal{Y} - \mathcal{X}_k - \mathcal{E}_k\|_\infty \leq \varepsilon$.
  7. $k \leftarrow k + 1$.
end While
Output: The target tensor $\mathcal{X}_k$.
```

Theorem 3. *Given a fixed $\mathbf{Q}$ in every K iterations, the tensor completion model (17) can be solved effectively by Algorithm 1 with $\mathbf{Q}_k = \mathbf{Q}$ in Eq. (20), where Ψ is replaced by $\mathcal{P}_\Omega$. The rigorous convergence guarantees can be obtained directly due to the convexity (See Supplementary Materials).*

Complexity: The per-iteration complexity in Algorithm 1 is $\mathcal{O}\left(n_1 n_2 r(\min\{n_1, n_2\} + \frac{n_3}{K})\right)$. One iteration means updating all variables once in order. As for the TNN based model in (Lu et al. 2016), the computational complexity at each iteration is $\mathcal{O}(n_1 n_2 n_3(\min\{n_1, n_2\} + \log n_3))$. The size of tensor data have a great influence on the computational complexity. While n_3 is much larger than $r = n_1 n_2$ or K is large enough, our method will be more efficient than TNN. In addition, thanks to $\mathbf{Q} \in \mathbb{R}^{n \times r}$, our method avoids the multiplication and SVD in $\mathbb{C}$, which slightly reduces the computational cost. Our running time experiment in **Figure** 3 also validate the efficiency of our Algorithm in some cases.

Performance: With a fixed $\mathbf{Q}$, the exact tensor completion guarantee for model (8) is shown in Theorem 4. Our synthetic experiments also verify the conclusion to some extent.

Algorithm 2 Solving the proximal operator $\mathbf{Prox}_{\lambda, \|\cdot\|_{Q,*}}(\mathcal{T})$ in Eq. (19) and (21).

```
Input: Tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, column orthonormal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times r}$.
  1. $\mathcal{G} = \mathcal{T} \times_3 \mathbf{Q}$.
  2. for $i = 1$ to $r$:
       $[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{SVD}(\mathbf{G}^{(i)})$.
       $\mathbf{G}^{(i)} = \mathbf{U}(\mathbf{S} - \lambda I)_+ \mathbf{V}^\top$,   where $(x)_+ = \max\{x, 0\}$.
  3. end for
  4. $\mathcal{X} = \mathcal{G} \times_3 \mathbf{Q}^\top + \mathcal{T} \times_3 (\mathbf{I} - \mathbf{Q}\mathbf{Q}^\top)$.
Output: Tensor $\mathcal{X}$.
```

Theorem 4. *Given a fixed orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{n_3 \times n_3}$ and $\Omega \sim Ber(p)$, assume that tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ $(n_1 \geq n_2)$ has a low tensor Q-rank structure and* $\text{rank}_Q(\mathcal{X}) = R$. *If $|\Omega| \geq \mathcal{O}(\mu R n_1 \log(n_1 n_3))$, then $\mathcal{X}$ is the unique solution to Eq.(8) with high probability, where Ψ is replaced by $\mathcal{P}_\Omega$, and μ is the corresponding incoherence parameter (See Supplementary Materials).*

Through the proof of (Lu et al. 2018), the sampling rate p should be proportional to $\max\{\|\mathcal{P}_{\mathcal{T}}(\mathfrak{e}_{ijk})\|_F^2\}$. (The definition of projection operators $\mathcal{P}_{\mathcal{T}}$ and $\mathfrak{e}_{ijk}$ can be found in (Lu et al. 2016; Lu et al. 2018) or in Supplementary materials, where $\mathcal{T}$ is the singular space of the ground-truth.) **Proposition 15** in (Lu et al. 2018) also implies that for any $\Delta \in \mathcal{T}$, we need to have $\mathcal{P}_\Omega(\Delta) = 0 \Leftrightarrow \Delta = 0$. These two conditions indicate that once the spatial dimension of $\mathcal{T}$ is large, a larger sampling rate p is needed. And **Figure 3** in (Lu et al. 2018) verifies the rationality of this deduction by experiment.

In fact, the smoothness of data along the third dimension has a great influence on the Degree of Freedom (DoF) of space $\mathcal{T}$. Non-smooth change along the third dimension is likely to increase the spatial dimension of $\mathcal{T}$ under the Fourier basis vectors, which makes the TNN based methods ineffective. Our experiments on CIFAR-10 (Table 1) confirm this conclusion.

As for the model (16) with adaptive $\mathbf{Q}$, extracting principal components along the third dimension makes the spatial dimension of corresponding $\mathcal{T}_Q$ as small as possible, where $\mathcal{T}_Q$ is the singular space of the ground-truth under $\mathbf{Q}$. In other words, for more complex data with non-smoothness along the third dimension, the adaptive $\mathbf{Q}$ may reduce the dimension of $\mathcal{T}_Q$ and make $\max\{\|\mathcal{P}_{\mathcal{T}_Q}(\mathfrak{e}_{ijk})\|_F^2\}$ smaller than $\max\{\|\mathcal{P}_{\mathcal{T}}(\mathfrak{e}_{ijk})\|_F^2\}$, leading to a lower bound for the sampling rate p. Our experimental results in **Figure** 2 and **Tables** 1 and 2 also illustrate that our proposed method performs better than TNN based method in the case of complex data with lower sampling rates.

Experiments

In this section, we conduct numerical experiments to evaluate our proposed model (16). Assume that the observed corrupted tensor is $\mathcal{Y}$, and the true tensor is $\mathcal{X}_0 \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. We represent the recovered tensor (output of the algorithms) as

$\mathcal{X}$, and use Peak Signal-to-Noise Ratio (PSNR) to measure the reconstruction error:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{n_1 n_2 n_3 \|\mathcal{X}_0\|_\infty^2}{\|\mathcal{X} - \mathcal{X}_0\|_F^2}\right). \qquad (25)$$

Synthetic Experiments

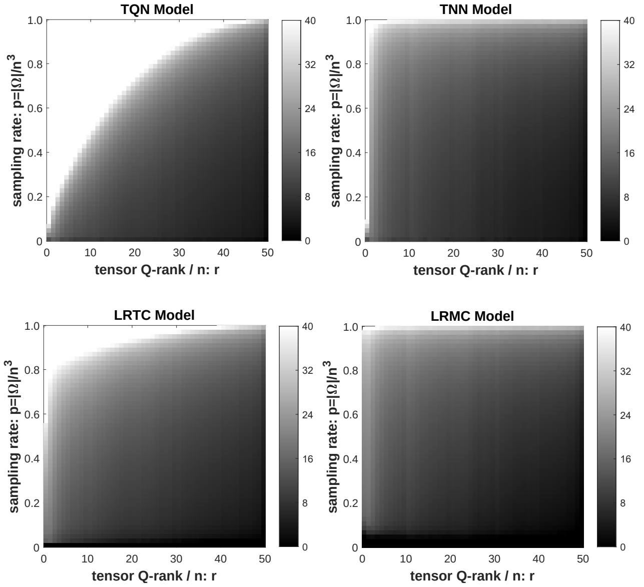

Figure 2: The numbers plotted on the above figure are the average PSNRs within 10 random trials. The gray scale reflects the quality of completion results of four different models (TQN, TNN, LRTC, LRMC). Here $n_1 = n_2 = n_3 = 50$ and the while area represents a maximum PSNR of 40.

In this part we compare our proposed method (named TQN model) with other mainstream algorithms, including TNN (Zhang et al. 2014; Lu et al. 2018), SiLRTC (Liu et al. 2013), and LRMC (Candès and Recht 2009).

We examine the completion task with varying tensor Q-rank of tensor $\mathcal{Y}$ and varying sampling rate p. Firstly, we generate a random tensor $\mathcal{M} \in \mathbb{R}^{50\times 50\times 50}$, whose entries are independently sampled from an $\mathcal{N}(0, 1/50)$ distribution. Then we choose p in $[0.01 : 0.02 : 0.99]$ and r in $[1 : 1 : 50]$, where the column orthonormal matrix $\mathbf{W} \in \mathbb{R}^{50\times r}$ satisfies $\mathbf{W} = \mathrm{PCA}(\mathcal{M}, 3, r)$. We let $\mathcal{Y} = \mathcal{M} \times_3 \mathbf{W} \times_3 \mathbf{W}^\top$ be the true tensor. After that, we create the index set Ω by using a Bernoulli model to randomly sample a subset from $\{1, \ldots, 50\} \times \{1, \ldots, 50\} \times \{1, \ldots, 50\}$. The sampling rate p is $|\Omega|/50^3$. For each pair of (p, r), we simulate 10 times with different random seeds and take the average as the final result. As for the parameters of TQN model in Algorithm 1, we set $\rho = 1.1$, $\mu_0 = 10^{-4}$, $\mu_{max} = 10^{10}$, and $\epsilon = 10^{-8}$. For the LRTC model, we set $\alpha = [1, 1, 1]$.

As shown in the upper left corner regions of TQN model in Figure 2, Algorithm 1 can effectively solve our proposed model (16). The larger tensor Q-rank it is, the larger the sampling rate p is needed, which is consistent with our Performance Analysis in Theorem 4.

Table 1: Comparisons of PSNR results on CIFAR images with different sampling rates. **Up:** experiments on the case $\mathcal{Y}_1 \in \mathbb{R}^{32\times 32\times 3000}$. **Down:** experiments on the case $\mathcal{Y}_2 \in \mathbb{R}^{32\times 32\times 10000}$.

Sampling Rate p	0.1	0.2	0.3	0.4	0.5	0.6
TQN with Random **Q** (Ours)	10.86	15.47	18.09	20.20	22.30	24.49
TQN with Oracle **Q** (Ours)	25.39	30.85	39.43	**109.52**	>**200**	>**200**
TQN with Adaptive **Q** (Ours)	**18.83**	**21.10**	**22.89**	**24.56**	**26.26**	**28.07**
TNN (Lu et al. 2018)	9.84	12.73	15.68	18.71	21.60	24.26
SiLRTC (Liu et al. 2013)	16.87	20.04	21.99	23.80	25.62	27.57
LRMC (Candès and Recht 2009)	11.20	15.81	18.26	20.41	22.51	24.72

Sampling Rate p	0.1	0.2	0.3	0.4	0.5	0.6
TQN with Random **Q** (Ours)	10.84	15.45	18.06	20.19	22.29	24.48
TQN with Oracle **Q** (Ours)	45.75	>**200**	>**200**	>**200**	>**200**	>**200**
TQN with Adaptive **Q** (Ours)	**19.06**	**21.43**	**23.27**	**24.97**	**26.65**	**28.42**
TNN (Lu et al. 2018)	8.18	10.10	12.19	14.63	17.59	21.20
SiLRTC (Liu et al. 2013)	14.02	19.65	22.44	24.38	26.21	28.12
LRMC (Candès and Recht 2009)	11.15	15.79	18.25	20.40	22.51	24.72

By comparing the results of four methods, we can find that TNN and LRMC have very poor robustness to the data with non-smooth change. And the results of TQN, LRTC, and TNN demonstrate our assumptions (Motivation), which may imply that TQN combines the advantages of TNN and SNN.

Real-World Datasets

In this part we compare our proposed method with TNN, SiLRTC, LRMC, Latent Trace Norm (Tomioka and Suzuki 2013), and t-Schatten-p norm (Kong, Xie, and Lin 2018) with $p = 2/3$. For other improved or matrix factorization-based algorithms, such as (Zhou et al. 2018; Xu et al. 2017), our model can also be extended in ways similar to theirs. **For the sake of fairness**, we only compare our method with the basic mainstream framework. We validate our algorithm on three datasets: (1) CIFAR-10[1]; (2) COIL-20[2]; (3) HMDB51[3]. We set $\rho = 1.1$, $\mu_0 = 10^{-4}$, $\mu_{max} = 10^{10}$, $\epsilon = 10^{-8}$, and $K = 1$ in our method TQN, and set $\mathbf{p} = [2, 1]$ in tSp. As for the others, we use the default settings of Lu et al.[4]

Influences of Q Corresponding to Remark 1, we use a Random orthogonal matrix and an Oracle matrix (the right singular matrix of the ground-truth unfolding matrix) to test the influence of **Q**. The results of three TQN based models in Tables 1 and 2 show that **Q** play an important role in tensor recovery. Comparing with Random **Q** case, our Algorithm 1 is effective for searching a better **Q**. Table 1 also shows that a proper **Q** may make recover the ground-truth more easily. For example, with sampling rate $p \geq 0.2$ on 10000 images, an Oracle matrix **Q** can lead to an "exact" recovery.

CIFAR-10 We consider the worst case for TNN based methods that there is almost no smoothness along the third dimension of the data. We randomly selected 3000 and 10000 images from one batch of CIFAR-10 (Krizhevsky and Hinton 2009) as our true tensors $\mathcal{Y}_1 \in \mathbb{R}^{32\times 32\times 3000}$

[1] http://www.cs.toronto.edu/~kriz/cifar.html.

[2] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php.

[3] http://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/.

[4] https://github.com/canyilu/LibADMM.

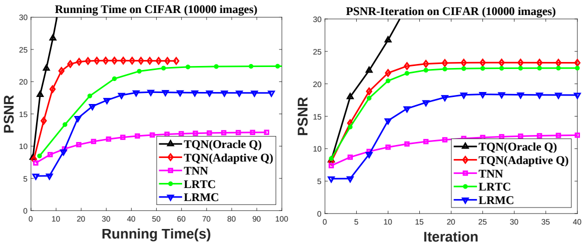

Figure 3: Running time comparisons of different methods, where $\mathcal{Y} \in \mathbb{R}^{32\times32\times10000}$ and sampling rate $p = 0.3$.

and $\mathcal{Y}_2 \in \mathbb{R}^{32\times32\times10000}$, respectively. Then we solve the model (17) with our proposed Algorithm 1. The results are shown in Table 1. In the latter case, $n_3 \gg n_1 n_2 = r$ holds.

Table 1 verifies our hypothesis that TNN regularization performs badly on data with non-smooth change along the third dimension. Our method and SiLRTC are obviously better than the other two methods in the case of low sampling rate. Moreover, by comparing the two groups of experiments, we can see that TQN and SiLRTC perform better in $\mathcal{Y}_2$. This may be due to that increasing the data volume may make the principal components more significant. The first few principal components of $\mathcal{Y}_1$ and $\mathcal{Y}_2$ are shown in Supplementary Materials for further explanation.

The above analyses confirm that our proposed regularization are data-dependent just like SNN regularization. And based on better definition of tensor singular value, we can make better use of internal structure than SNN regularization.

Running time on CIFAR As shown in Figure 3, we test the running times of different models. The two figures indicate that, when $n_3 \gg n_1 n_2$, our TQN models has the highest computational efficiency in each iteration. For the case $n_3 < n_1 n_2$, Figure 4 implies that setting $r < n_1 n_2$ can balance computational efficiency and recovery accuracy.

COIL-20 and Short Video from HMDB51 COIL-20 (Nene et al. 1996) contains 1440 images of 20 objects which are taken from different angles. The size of each image is processed as 128×128, which means $\mathcal{Y} \in \mathbb{R}^{128\times128\times1440}$. The upper part of Table 2 shows the results of the numerical experiments. We select a background-changing video from HMDB51 (Kuehne et al. 2011) for the video inpainting task, where $\mathcal{Y} \in \mathbb{R}^{240\times320\times146}$. Figure 1 shows some frames of this video. The lower part of Table 2 shows the results.

We can see that "Latent trace norm" is much better than TNN in COIL, which validates our assumption that tensor trace norm is much more robust than TNN in processing non-smooth data.

Overall, both TQN and TNN perform better than the other methods on these two datasets due to the higher smoothness along the third dimension. This is mainly because the definitions of tensor singular value in TQN and TNN make better use of the tensor internal structure, and this is also the main difference between TQN and SNN. Meanwhile, our method is obviously better than the others at all sampling rates, which reflects the superiority of our data dependent $\mathbf{Q}$. All visual comparisons are provided in Supplementary Materials.

Table 2: Comparisons of PSNR results on COIL images and video inpainting with different sampling rates. **Up:** the COIL dataset with $\mathcal{Y} \in \mathbb{R}^{128\times128\times1440}$. **Down:** a short video from HMDB51 with $\mathcal{Y} \in \mathbb{R}^{240\times320\times126}$.

Sampling Rate p	0.1	0.2	0.3	0.4	0.5	0.6
TQN with Random $\mathbf{Q}$ (Ours)	16.05	20.07	23.02	25.57	27.95	30.34
TQN with Oracle $\mathbf{Q}$ (Ours)	22.97	25.32	27.18	28.90	30.68	32.51
TQN with Adaptive $\mathbf{Q}$ (Ours)	**22.79**	**25.34**	**27.29**	**29.08**	**30.86**	**32.74**
TNN (Lu et al. 2018)	19.20	22.08	24.45	26.61	28.72	30.91
SiLRTC (Liu et al. 2013)	18.87	21.80	23.89	25.67	27.37	29.14
Latent Trace Norm (Tomioka and Suzuki 2013)	19.09	22.98	25.75	28.11	30.40	32.42
LRMC (Candès and Recht 2009)	16.32	20.11	22.91	25.34	27.65	29.98

Sampling Rate p	0.1	0.2	0.3	0.4	0.5	0.6
TQN with Random $\mathbf{Q}$ (Ours)	18.85	22.76	25.87	28.73	31.55	34.48
TQN with Oracle $\mathbf{Q}$ (Ours)	23.44	27.61	31.37	35.11	38.92	42.74
TQN with Adaptive $\mathbf{Q}$ (Ours)	**23.97**	**28.09**	**31.76**	**35.33**	**39.06**	**42.87**
TNN (Lu et al. 2018)	22.40	25.58	28.28	30.88	33.55	36.41
tSp (p=2/3) (Kong, Xie, and Lin 2018)	22.41	25.32	27.67	31.26	34.23	36.98
SiLRTC (Liu et al. 2013)	18.42	22.33	25.76	29.15	32.59	36.15
Latent Trace Norm (Tomioka and Suzuki 2013)	18.94	22.72	25.65	28.26	30.79	33.48
LRMC (Candès and Recht 2009)	18.87	22.79	25.94	28.82	31.65	34.61

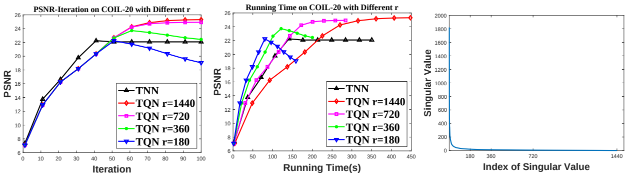

Figure 4: The relations among running times, different r, and the singular values of $\mathbf{T}_{(3)}$ on COIL, where $p = 0.2$.

Influence of r in $\mathbf{Q} \in \mathbb{R}^{n_3 \times r}$ Remarks 1 and 2 imply that r denotes the apriori assumption of the subspace dimension of the ground-truth. It means that the dimension of the frontal slice subspace of the true tensor $\mathcal{T}$ (also as the column subspace of mode-3 unfolding matrix $\mathbf{T}_{(3)}$) is no more than r. Figure 4 illustrates the relations among running times, different r, and the singular values of $\mathbf{T}_{(3)}$. We project the solution $\mathcal{X}_k$ (in Eq. (21)) onto the subspace of $\mathbf{Q}_k$, which means $\hat{\mathcal{X}}_k := \mathcal{X}_k \times_3 (\mathbf{Q}_k \mathbf{Q}_k^\top)$.

As shown in the conduct of Figure 4, the column subspace of $\mathbf{T}_{(3)}$ is more than 360. If $r \leq 360$, the algorithm will converge to a bad point which only has an r-dimensional subspace. Therefore, in our experiments, we usually set $r = \min\{n_1 n_2, n_3\}$ to make sure that r is greater than the true tensor's subspace dimension. This apriori assumption is commonly used in factorization-based algorithms.

The running time decreases with the decrease of r. Although $r = 1440$ needs more time to converge than TNN, it obtains a better recovery. And a smaller r does speed up the calculation but harms the accuracy.

Conclusions

We propose a new definition of tensor rank named tensor Q-rank, and further define an Orthogonal Proximal Constraint on $\mathbf{Q}$ to make it become a learnable variable w.r.t. the data. We also provide an envelope of our rank function and apply it to the tensor completion problem. We analyze why our method may perform better than TNN based methods in non-smooth data (along the third dimension) with low sampling rates, and conduct experiments to verify our conclusions.

References

[Absil, Mahony, and Sepulchre 2009] Absil, P.-A.; Mahony, R.; and Sepulchre, R. 2009. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.

[Candès and Recht 2009] Candès, E. J., and Recht, B. 2009. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics* 9(6):717.

[De Lathauwer, De Moor, and Vandewalle 2000] De Lathauwer, L.; De Moor, B.; and Vandewalle, J. 2000. On the best rank-1 and rank-(R_1, R_2,...,R_N) approximation of higher-order tensors. *SIAM journal on Matrix Analysis and Applications* 21(4):1324–1342.

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

[Friedland and Lim 2018] Friedland, S., and Lim, L.-H. 2018. Nuclear norm of higher-order tensors. *Mathematics of Computation* 87(311):1255–1281.

[Kilmer and Martin 2011] Kilmer, M. E., and Martin, C. D. 2011. Factorization strategies for third-order tensors. *Linear Algebra and its Applications* 435(3):641–658.

[Kolda and Bader 2009] Kolda, T. G., and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM review* 51(3):455–500.

[Kong, Xie, and Lin 2018] Kong, H.; Xie, X.; and Lin, Z. 2018. t-schatten-p norm for low-rank tensor recovery. *IEEE Journal of Selected Topics in Signal Processing* 12(6):1405–1419.

[Krizhevsky and Hinton 2009] Krizhevsky, A., and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, Citeseer.

[Kruskal 1989] Kruskal, J. B. 1989. *Rank, decomposition, and uniqueness for 3-way and n-way arrays*. North-Holland Publishing Co.

[Kuehne et al. 2011] Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: a large video database for human motion recognition. In *IEEE International Conference on Computer Vision*, 2556–2563.

[Lin, Liu, and Li 2015] Lin, Z.; Liu, R.; and Li, H. 2015. Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. *Machine Learning* 99(2):287.

[Lin, Liu, and Su 2011] Lin, Z.; Liu, R.; and Su, Z. 2011. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Advances in neural information processing systems*, 612–620.

[Liu et al. 2013] Liu, J.; Musialski, P.; Wonka, P.; and Ye, J. 2013. Tensor completion for estimating missing values in visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(1):208–220.

[Lu et al. 2016] Lu, C.; Feng, J.; Chen, Y.; Liu, W.; Lin, Z.; and Yan, S. 2016. Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5249–5257.

[Lu et al. 2017] Lu, C.; Feng, J.; Yan, S.; and Lin, Z. 2017. A unified alternating direction method of multipliers by majorization minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(3):527–541.

[Lu et al. 2018] Lu, C.; Feng, J.; Lin, Z.; and Yan, S. 2018. Exact low tubal rank tensor recovery from gaussian measurements. In *International Conference on Artificial Intelligence*.

[Nene et al. 1996] Nene, S. A.; Nayar, S. K.; Murase, H.; et al. 1996. Columbia object image library (coil-20).

[Romera-Paredes and Pontil 2013] Romera-Paredes, B., and Pontil, M. 2013. A new convex relaxation for tensor completion. In *Advances in Neural Information Processing Systems*, 2967–2975.

[Sengupta and Mitra 1999] Sengupta, A. M., and Mitra, P. P. 1999. Distributions of singular values for some random matrices. *Physical Review E* 60(3):3389.

[Tomioka and Suzuki 2013] Tomioka, R., and Suzuki, T. 2013. Convex tensor decomposition via structured schatten norm regularization. In *Advances in neural information processing systems*, 1331–1339.

[Tomioka, Hayashi, and Kashima 2010] Tomioka, R.; Hayashi, K.; and Kashima, H. 2010. On the extension of trace norm to tensors. In *NIPS Workshop on Tensors, Kernels, and Machine Learning*, 7.

[Tucker 1966] Tucker, L. R. 1966. Some mathematical notes on three-mode factor analysis. *Psychometrika* 31(3):279–311.

[Wimalawarne, Sugiyama, and Tomioka 2014] Wimalawarne, K.; Sugiyama, M.; and Tomioka, R. 2014. Multitask learning meets tensor factorization: task imputation via convex optimization. In *Advances in neural information processing systems*, 2825–2833.

[Xu and Yin 2015] Xu, Y., and Yin, W. 2015. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences* 6(3):1758–1789.

[Xu et al. 2017] Xu, Y.; Hao, R.; Yin, W.; and Su, Z. 2017. Parallel matrix factorization for low-rank tensor completion. *Inverse Problems & Imaging* 9(2):601–624.

[Zhang et al. 2014] Zhang, Z.; Ely, G.; Aeron, S.; Hao, N.; and Kilmer, M. 2014. Novel methods for multilinear data completion and de-noising based on tensor-SVD. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3842–3849.

[Zhou et al. 2018] Zhou, P.; Lu, C.; Lin, Z.; and Zhang, C. 2018. Tensor factorization for low-rank tensor completion. *IEEE Transactions on Image Processing* 27(3):1152–1163.