Stochastic Gradient Descent on Tensors with Missing Data

Anna Ma[1], **Deanna Needell**[2], **Alexander Xue**[2]

[1]University of California, Irvine, Department of Mathematics
[2]University of California, Los Angeles, Department of Mathematics

Abstract

Large tensor linear systems of equations pose a challenge due to the sheer amount of data stored. They quickly become difficult to solve, due to either time constraints or memory constraints, and they become even more challenging when some of the data is missing. We present a stochastic iterative method to address both of these issues by adapting Stochastic Gradient Descent to the tensor case and adding a correction term to debias the gradient for missing data. We prove convergence results for our method and experimentally verify these results on synthetic data.

Introduction

Due to the ever-increasing need to store and process large datasets in fields like computer vision and machine learning, the need to store data in multidimensional arrays, called tensors, has become essential. Tensors allow for efficient representation and manipulation of complex, high-dimensional data, enabling algorithms that can take advantage of the multidimensional structure.

In (Kilmer and Martin 2011), the authors define a *t-product* between tensors. This t-product allows one to adapt concepts from linear algebra in the matrix case to tensors. As a linear operator, the t-product preserves the multidimensional structure of the tensors and thus avoids any structural loss that would occur by naively flattening tensors into a matrix and performing matrix multiplication. The t-product has seen applications in computer vision (Yin et al. 2019), in neural networks (Newman et al. 2018), and in data completion (Hu et al. 2017), among many others.

In this paper, we consider the problem of solving linear systems for third-order tensors assuming that some of the data in a tensor is missing. This is a common problem in applications where e.g. the data stems from inactive sensors, incomplete survey data, collaborative filtering, or memory needs. This extends (Ma and Needell 2018), where the authors address a similar problem but in the matrix setting.

Background on Tensors

We focus on third-order tensors throughout. Let $\mathcal{A} \in \mathbb{R}^{m\times\ell\times n}$ be a third-order tensor. We use the notations $\mathcal{A}_{i::} \in \mathbb{R}^{1\times\ell\times n}$ and $\mathcal{A}_{::k} \in \mathbb{R}^{m\times\ell\times 1}$ to denote the ith row slice and kth frontal slice of $\mathcal{A}$, respectively. We use the bold capital letter notation $\mathbf{A}_k \in \mathbb{R}^{m\times\ell}$ to mean the frontal slice $\mathcal{A}_{::k}$ viewed as a matrix.

Define $\text{bcirc}(\mathcal{A})$ to be the block-circulant matrix created from the frontal slices of $\mathcal{A}$. That is,

$$\text{bcirc}(\mathcal{A}) := \begin{pmatrix} \mathbf{A}_0 & \mathbf{A}_{n-1} & \mathbf{A}_{n-2} & \dots & \mathbf{A}_1 \\ \mathbf{A}_1 & \mathbf{A}_0 & \mathbf{A}_{n-1} & \dots & \mathbf{A}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{n-1} & \mathbf{A}_{n-2} & \mathbf{A}_{n-3} & \dots & \mathbf{A}_0 \end{pmatrix}.$$

Next, define $\text{unfold}(\mathcal{A})$ to be the flattened matrix version of $\mathcal{A}$, i.e. it is the $mn \times \ell$ matrix

$$\text{unfold}(\mathcal{A}) := \begin{pmatrix} \mathbf{A}_0 \\ \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{n-1} \end{pmatrix}.$$

Also let $\text{fold}(\cdot)$ be the function that reverses $\text{unfold}(\cdot)$, so that $\text{fold}(\text{unfold}(\mathcal{A})) = \mathcal{A}$. Note that fold requires the number of frontal slices n in order to do the unfolding. All of our tensors will have the same number n of frontal slices, so there is no ambiguity in how fold works.

We can now define the t-product. Let $\mathcal{X}$ be a $\mathbb{R}^{\ell\times q\times n}$ tensor. The t-product, as defined in (Kilmer and Martin 2011), between $\mathcal{A}$ and $\mathcal{X}$ is

$$\mathcal{A} * \mathcal{X} := \text{fold}\left(\text{bcirc}(\mathcal{A})\,\text{unfold}(\mathcal{X})\right),$$

where the product $\text{bcirc}(\mathcal{A})\,\text{unfold}(\mathcal{X})$ is the usual matrix-matrix product.

Next, the conjugate transpose of a tensor is denoted by $\mathcal{A}^*$, and it is the tensor obtained by taking the conjugate transpose of every frontal slice and also reversing the order of the frontal slices $1, 2, \dots, n-1$. In other words,

$$\text{unfold}(\mathcal{A}^*) = \begin{pmatrix} \mathbf{A}_0^* \\ \mathbf{A}_{n-1}^* \\ \vdots \\ \mathbf{A}_1^* \end{pmatrix}.$$

When using both slice and transpose notation, we apply the slice first then apply the transpose, e.g. $\mathcal{A}_{i::}^* := (\mathcal{A}_{i::})^*$.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Lastly, we define the inner product of two tensors of the same dimension as the sum of the elementwise products of the tensors,

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i,j,k} \mathcal{A}_{ijk} \mathcal{B}_{ijk},$$

and we define the norm $\|\mathcal{A}\|$ to be the Frobenius norm,

$$\|\mathcal{A}\| = \sum_{i,j,k} |\mathcal{A}_{ijk}|^2.$$

Tensor Linear Systems with Missing Data

Let $\mathcal{A} \in \mathbb{R}^{m \times \ell \times n}, \mathcal{X} \in \mathbb{R}^{\ell \times q \times n}$, and $\mathcal{B} \in \mathbb{R}^{m \times q \times n}$ be third-order tensors. $\mathcal{A}$ represents the data, $\mathcal{X}$ represents the unknown variables, and $\mathcal{B}$ represents the measurements. We are interested in solving the tensor linear system $\mathcal{A} * \mathcal{X} = \mathcal{B}$ given $\mathcal{B}$ and only partial knowledge of $\mathcal{A}$. In particular, we assume that every entry of $\mathcal{A}$ is missing independently with a probability $1 - p$, a reasonable assumption to make in the scenarios mentioned above.

Formally, let $\mathcal{D} \in \{0, 1\}^{m \times \ell \times n}$ be a *binary mask* indicating which elements of $\mathcal{A}$ we observe, where the entries of $\mathcal{D}$ are i.i.d. Bernoulli random variables with parameter p. The tensor $\tilde{\mathcal{A}}$ of observed elements satisfies $\tilde{\mathcal{A}} = \mathcal{D} \circ \mathcal{A}$, where $\circ$ indicates the element-wise product. Then, we aim to solve the optimization program

$$\text{Given } \tilde{\mathcal{A}} = \mathcal{D} \circ \mathcal{A}, \mathcal{B}, \text{ and } p$$
$$\text{Find } \mathcal{X}_\star = \operatorname{argmin}_{\mathcal{X} \in \mathcal{W}} \frac{1}{2m} \|\mathcal{A} * \mathcal{X} - \mathcal{B}\|^2,$$

where $\mathcal{W}$ is a convex domain containing the solution $\mathcal{X}_\star$ of $\mathcal{A} * \mathcal{X} = \mathcal{B}$.

In (Ma and Needell 2018), the authors considered a similar problem but in the matrix linear case $Ax = b$, where A is a matrix and x and b are vectors. Like in our tensor case, the entries of A are missing independently with probability $1 - p$. The tensor problem is a generalization of this matrix problem. Indeed, if we set $\mathcal{A} \in \mathbb{R}^{m \times \ell \times 1}$, $\mathcal{X} \in \mathbb{R}^{\ell \times 1 \times 1}$, and $\mathcal{B} \in \mathbb{R}^{m \times 1 \times 1}$ to be the tensors satisfying $\mathbf{A}_0 = A, \mathbf{X}_0 = x$, and $\mathbf{B}_0 = b$, then the tensor problem is exactly the same as the matrix problem.

Furthermore, the tensor problem cannot be reduced to the specific matrix problem solved by Ma and Needell. Although the tensor equation $\mathcal{A} * \mathcal{X} = \mathcal{B}$ can be reduced to a matrix equation $AX = B$ for some appropriate matrices A, X, and B, the missing data model is different – entries in $\mathcal{A}$ are missing independently, while entries in the corresponding matrix A are no longer missing independently, since one entry missing in $\mathcal{A}$ corresponds to many missing entries in A.

Similar to what was done in the matrix case, we will solve the tensor case with Stochastic Gradient Descent.

Stochastic Gradient Descent for Tensors with Missing Data

Stochastic Gradient Descent (SGD) is a popular iterative method used to minimize a convex objective function $F(\mathcal{X})$ over a convex domain $\mathcal{W}$. It works by using an unbiased estimate of the gradient to determine what direction to step in at each iteration. If g is a random function satisfying $\mathbb{E}[g(\mathcal{X})] = \nabla F(\mathcal{X})$, then SGD proposes to update iterates via

$$\mathcal{X}^{t+1} = \mathcal{P}_{\mathcal{W}}(\mathcal{X}^t - \alpha_t g(\mathcal{X}^t)), \tag{1}$$

where α_t is an appropriately chosen learning rate or step size, and $\mathcal{P}_{\mathcal{W}}$ is the projection onto the convex set $\mathcal{W}$.

Previous works have used variants of SGD for tensor decomposition (Maehara, Hayashi, and Kawarabayashi 2016; Kolda and Hong 2020), completion (Papastergiou and Megalooikonomou 2017), and recovery (Chen and Qin 2021; Ma and Molitor 2022). For example, (Grotheer et al. 2024) combines SGD with thresholding for sparse tensor recovery and (Papastergiou and Megalooikonomou 2017) introduces proximal SGD-based algorithms for tensor completion. Randomized Kaczmarz, a special case of SGD, has also been studied on tensors. (Chen and Qin 2021) applies a tensor version of Kaczmarz using a learning rate proportional to $1/\|\mathcal{A}\|^2$ to solve tensor recovery problems, while (Ma and Molitor 2022) proposes a more complex update step by incorporating a tensor inverse. It should be noted that these works either solve tensor recovery with all data or deal with missing data (e.g., via tensor completion) independently. In this work, we tackle the recovery and missing data problems simultaneously.

With SGD, we can obtain various convergence results depending on additional properties on F or g. The following previously proven theorem is what we use for one of our convergence results. It is a powerful result that does not require many conditions on F or g.

Lemma 1. *((Shamir and Zhang 2013) Theorem 2) Suppose that F is convex and $\mathcal{W}$ is a closed convex domain containing the solution $\mathcal{X}_\star$. Furthermore, suppose that for some constants G, K, it holds that $\mathbb{E}[g(\mathcal{X})] = \nabla F(\mathcal{X})$ and $\mathbb{E}[\|g(\mathcal{X})\|^2] \leq G$ for all $\mathcal{X} \in \mathcal{W}$, and $\sup_{\mathcal{W}_1, \mathcal{W}_2 \in \mathcal{W}} \|\mathcal{W}_1 - \mathcal{W}_2\| \leq K$. Using step size $\alpha_t = C/\sqrt{t}$ where $C > 0$ is a constant, and using the SGD update in equation 1, the resulting iterates satisfy*

$$\mathbb{E}[F(\mathcal{X}^T) - F(\mathcal{X}_\star)] \leq \left(\frac{K^2}{C} + CG \right) \frac{2 + \log(T)}{\sqrt{T}}.$$

Here, C represents the step size constant factor, G gives a bound on the expected squared norm of g, and K gives a bound on the distance between any two elements $\mathcal{W}_1$ and $\mathcal{W}_2$ of $\mathcal{W}$, which can be interpreted as a bound on the size of $\mathcal{W}$.

The main challenge in applying this lemma to our missing data setting is in choosing a suitable g that provides an unbiased estimate for ∇F, as we cannot directly compute ∇F without knowing $\mathcal{A}$. Let

$$F(\mathcal{X}) := \frac{1}{2m} \|\mathcal{A} * \mathcal{X} - \mathcal{B}\|^2$$

be our objective function and let fdiag $(\cdot)$ be the function that zeroes out all the entries of a tensor except those on the main

diagonal of the zeroth frontal slice. Then, the choice

$$g(\mathcal{X}) := \frac{1}{p^2}\left(\tilde{\mathcal{A}}^*_{i::} * (\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::})\right) - \frac{1-p}{p^2}\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}$$

suffices.

Algorithm 1 straightforwardly uses this g for SGD for a budget of T iterations. During each iteration, a row index $i \in [m] := \{0, 1, \ldots, m-1\}$ is chosen uniformly at random, $g(\mathcal{X}^t)$ is computed based on the choice, and then $\mathcal{X}^t$ is updated via equation (1). We call this algorithm missing data SGD for tensors (mSGDT).

Algorithm 1 mSGDT

Input: $\mathcal{X}^0 \in \mathbb{R}^{\ell\times q\times n}$, $\tilde{\mathcal{A}} \in \mathbb{R}^{m\times\ell\times n}$, $\mathcal{B} \in \mathbb{R}^{m\times q\times n}$, $p \in (0,1)$.
procedure ($\tilde{\mathcal{A}}, \mathcal{B}, T, p, \{\alpha_t\}$)
  for $t = 0, 1, \ldots, T-1$ **do**
    Choose row index $i \in [m]$ uniformly at random
    $g(\mathcal{X}^t) = \frac{1}{p^2}\left(\tilde{\mathcal{A}}^*_{i::} * (\tilde{\mathcal{A}}_{i::} * \mathcal{X}^t - p\mathcal{B}_{i::})\right) - \frac{1-p}{p^2}\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}^t$
    $\mathcal{X}^{t+1} = \mathcal{P}_{\mathcal{W}}\left(\mathcal{X}^t - \alpha_t g(\mathcal{X}^t)\right)$
  end for
  Output $\mathcal{X}^T$
end procedure

There are two sources of randomness for g: the choice of the row index i and the binary mask $\mathcal{D}$. Let $\mathbb{E}_i[\cdot]$ denote the expectation with respect to the choice i, and let $\mathbb{E}_\delta[\cdot]$ denote the expectation with respect to the mask $\mathcal{D}$. Thus the expectation of $g(\mathcal{X})$ is the expectation over i and δ. That is,

$$\mathbb{E}[g(\mathcal{X})] = \mathbb{E}_i[\mathbb{E}_\delta[g(\mathcal{X})]].$$

We will the need the following assumption for when we calculate $\mathbb{E}_\delta[g(\mathcal{X})]$ later.

Assumption. Either every row of $\mathcal{A}_{i::}$ is chosen at most once during Algorithm 1, or the mask is redrawn at each iteration, meaning the data that is available from each row may change with time. If $\mathcal{A}$ has a large number of rows, it is reasonable to assume that the first stipulation is satisfied.

Convergence Results

We present two convergence results, one result for a learning rate proportional to $1/\sqrt{t}$ as Lemma 1 suggests and the other result for a constant step size.

Theorem 1 handles the $\alpha_t \propto 1/\sqrt{t}$ case. It says that the objective function will converge to 0 as the number of iterations goes to infinity. The main tool we will use is Lemma 1. As a remark, (Shamir and Zhang 2013) also proved a theoretically stronger result for a step size proportional to $1/t$, and the only extra ingredient we would need is to show that F is μ-strongly convex for some $\mu > 0$. In the supplemental material, we outline when such a case would occur. However, we experimentally observed that the $\alpha_t \propto 1/\sqrt{t}$ step size still performed better, despite the theoretical results.

Theorem 1. *Let $\mathcal{X}_\star \in \mathbb{R}^{\ell\times q\times n}$ be such that $\mathcal{A} * \mathcal{X}_\star = \mathcal{B}$, let $\mathcal{D}$ be a binary mask with i.i.d. Bernoulli random variables with parameter p, and let $\tilde{\mathcal{A}} = \mathcal{D}\circ\mathcal{A}$. Choosing $\alpha_t = C/\sqrt{t}$, Algorithm 1 converges in expectation such that the error in objective function $F(\mathcal{X}) = \frac{1}{2m}\|\mathcal{A} * \mathcal{X} - \mathcal{B}\|^2$ satisfies*

$$\mathbb{E}[F(\mathcal{X}^{t+1}) - F(\mathcal{X}_\star)] \le \left(\frac{K^2}{C} + CG\right)\frac{2+\log(T)}{\sqrt{T}},$$

where $G = \frac{2R^2(n^2+(1-p)^2 n)}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4 + \frac{4n^{3/2}R}{mp}\sum_i \|\mathcal{A}_{i::}\|^3\|\mathcal{B}_{i::}\| + \frac{2n}{m}\sum_i \|\mathcal{A}_{i::}\|^2\|\mathcal{B}_{i::}\|^2$, $R = \max_{\mathcal{X}\in\mathcal{W}}\|\mathcal{X}\|$, and K is a constant satisfying $\sup_{\mathcal{W}_1,\mathcal{W}_2\in\mathcal{W}}\|\mathcal{W}_1 - \mathcal{W}_2\| \le K$.

Proof. To use Lemma 1, we need to prove three facts about $F(\mathcal{X}) = \frac{1}{2m}\|\mathcal{A} * \mathcal{X} - \mathcal{B}\|^2$ and $g(\mathcal{X})$.

1. $F(\mathcal{X})$ is convex. This is straightforward.
2. $\mathbb{E}[g(\mathcal{X})] = \nabla F(\mathcal{X})$. This is computed in the supplemental material in Lemma 3.
3. $\mathbb{E}[\|g(\mathcal{X})\|^2] \le G$. Note that since $\mathcal{W}$ is bounded, $\mathbb{E}[\|g(\mathcal{X})\|^2]$ is necessarily bounded. But we explicitly calculate G to make the discovered bound more clear. This is done in the supplemental material in Lemma 4.

□

We now prove a theorem that explores the behavior of mSGDT in the case of a fixed step size. Theorem 2 shows that the error in the algorithm decreases quickly until it reaches a convergence horizon. Thus, the optimal way to apply the algorithm may be to first set a fixed step size and change it to a decreasing step size proportional to $1/\sqrt{t}$ after the convergence horizon is reached.

Theorem 2. *Let $\mathcal{X}_\star \in \mathbb{R}^{\ell\times q\times n}$ be such that $\mathcal{A} * \mathcal{X}_\star = \mathcal{B}$, let $\mathcal{D}$ be a binary mask with i.i.d. Bernoulli random variables with parameter p, and let $\tilde{\mathcal{A}} = \mathcal{D}\circ\mathcal{A}$. Suppose F is μ-strongly convex for some $\mu > 0$. Also let L_g be a Lipschitz constant for the update function g in Algorithm 1 over all possible binary masks $\mathcal{D}$. Lastly, let $G_\star$ be such that $\mathbb{E}[\|g(\mathcal{X}_\star)\|^2] \le G_\star$.*

Then, with fixed step size $\alpha < 1/L_g$, Algorithm 1 converges with expected error, for all $t > 0$,

$$\mathbb{E}[\|\mathcal{X}^t - \mathcal{X}_\star\|^2] \le r^t\|\mathcal{X}^0 - \mathcal{X}_\star\|^2 + \frac{\alpha G_\star}{\mu(1-\alpha L_g)},$$

where $r := (1 - 2\alpha\mu(1-\alpha L_g))$.

The three components μ, L_g, and $G_\star$ are investigated in the supplemental material. For $G_\star$ in particular, we again remark that such a $G_\star$ must exist because $\mathcal{X}_\star$ is contained within the bounded $\mathcal{W}$.

We need the following elementary lemma to rewrite $\|g(\mathcal{X}^t) - g(\mathcal{X}_\star)\|^2$ later.

Lemma 2 ((Needell, Ward, and Srebro 2014) Lemma A.1)**.** *If g is a function such that there exists a smooth f with $\nabla f = g$, and g has Lipschitz constant L, then*

$$\|g(\mathcal{X}) - g(\mathcal{Y})\|^2 \le L\langle \mathcal{X} - \mathcal{Y}, g(\mathcal{X}) - g(\mathcal{Y})\rangle.$$

For our function g, choosing $f(\mathcal{X}) := \frac{1}{2p^2}\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2 - \frac{1-p}{2p^2}\|\sqrt{\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right)} * \mathcal{X}\|^2$ satisfies the requirement $\nabla f(\mathcal{X}) = g(\mathcal{X})$ (see the supplemental material), where $\sqrt{\cdot}$ refers to the elementwise square root.

Proof of Theorem 2. Let $\mathcal{E}^t = \mathcal{X}^t - \mathcal{X}_\star$ denote the error at iteration t. Let $\mathbb{E}_{t-1}[\cdot]$ denote the expected value conditioned on the previous $t-1$ iterations. We have

$$
\begin{aligned}
&\mathbb{E}_{t-1}\|\mathcal{E}^{t+1}\|^2 \\
&\leq \mathbb{E}_{t-1}\left[\|\mathcal{X}^t - \alpha g(\mathcal{X}^t) - \mathcal{X}_\star\|^2\right] \\
&= \|\mathcal{E}^t\|^2 - 2\alpha\langle \mathcal{E}^t, \mathbb{E}_{t-1}\left[g(\mathcal{X}^t)\right]\rangle \\
&\quad + \alpha^2\mathbb{E}_{t-1}\left[\|g(\mathcal{X}^t)\|^2\right] \\
&= \|\mathcal{E}^t\|^2 - 2\alpha\langle \mathcal{E}^t, \nabla F(\mathcal{X}^t) - \nabla F(\mathcal{X}_\star)\rangle \\
&\quad + \alpha^2\mathbb{E}_{t-1}\left[\|g(\mathcal{X}^t)\|^2\right] \\
&\leq \|\mathcal{E}^t\|^2 - 2\alpha\langle \mathcal{E}^t, \nabla F(\mathcal{X}^t) - \nabla F(\mathcal{X}_\star)\rangle \\
&\quad + 2\alpha^2\mathbb{E}_{t-1}\left[\|g(\mathcal{X}^t) - g(\mathcal{X}_\star)\|^2\right] + 2\alpha^2\mathbb{E}_{t-1}\left[\|g(\mathcal{X}_\star)\|^2\right] \\
&\leq \|\mathcal{E}^t\|^2 - 2\alpha\langle \mathcal{E}^t, \nabla F(\mathcal{X}^t) - \nabla F(\mathcal{X}_\star)\rangle \\
&\quad + 2\alpha^2 L_g\langle \mathcal{E}^t, \mathbb{E}_{t-1}[g(\mathcal{X}^t)] - \mathbb{E}_{t-1}[g(\mathcal{X}_\star)]\rangle + 2\alpha^2 G_\star \\
&\leq \|\mathcal{E}^t\|^2 - 2\alpha(1-\alpha L_g)\langle \mathcal{E}^t, \nabla F(\mathcal{X}^t) - \nabla F(\mathcal{X}_\star)\rangle \\
&\quad + 2\alpha^2 G_\star \\
&\leq \|\mathcal{E}^t\|^2 - 2\alpha(1-\alpha L_g)\mu\|\mathcal{E}^t\|^2 + 2\alpha^2 G_\star \\
&= (1 - 2\alpha\mu(1-\alpha L_g))\,\|\mathcal{E}^t\|^2 + 2\alpha^2 G_\star \\
&= r\|\mathcal{E}^t\|^2 + 2\alpha^2 G_\star.
\end{aligned}
$$

The desired bound follows by iteratively applying this result. □

Experiments

In this section, we demonstrate the performance of mSGDT on synthetic data. We generate elements of $\mathcal{A} \in \mathbb{R}^{500\times 20\times 10}$ and $\mathcal{X} \in \mathbb{R}^{20\times 10\times 10}$ by drawing i.i.d. from a standard Gaussian distribution. Then, to simulate the missing data, whenever we draw a row $\mathcal{A}_{i::}$ from $\mathcal{A}$, we sample a new binary mask $\mathcal{D}_{i::} \in \mathbb{R}^{1\times 20\times 10}$ with entries drawn i.i.d. from a Bernoulli distribution with parameter p and compute $\tilde{\mathcal{A}}_{i::} = \mathcal{D}_{i::} \circ \mathcal{A}_{i::}$. Note that when we sample the same row repeatedly, a new mask is generated each time. This ensures that our assumption used in calculating $\mathbb{E}_\delta[g(\mathcal{X})]$ is satisfied.

Figures 1 and 2 both have the x-axis as the iteration and the y-axis as the approximation error, $\|\mathcal{X}^t - \mathcal{X}_\star\|$. We also vary the parameter p in the two figures, taking on values in $\{0.3, 0.5, 0.7, 0.99\}$.

Figure 1 presents the results for using a step size proportional to $1/\sqrt{t}$ as suggested by Theorem 1. We specifically used a step size of $\alpha_t = p^2/(10\sqrt{t})$. The size of the factor here depends on p, as with a larger p value, a more aggressive step size can be chosen, while with a lower p value, a less aggressive step size should be chosen.

Figure 2 shows the results for using a constant step size as suggested by Theorem 2. We chose a step size of $\alpha = p^2/3000$, again including a p^2 factor so that we use more aggressive step sizes for larger values of p. The Lipschitz constant L_g was computed to be approximately $2600p^2$, hence the choice of $\alpha = p^2/3000 < 1/L_g$. Experimentally, we found that larger step sizes as large as $\alpha = p^2/100$ also achieved a convergence horizon. Significantly larger step sizes, such as $\alpha = p^2/10$, led to divergence.

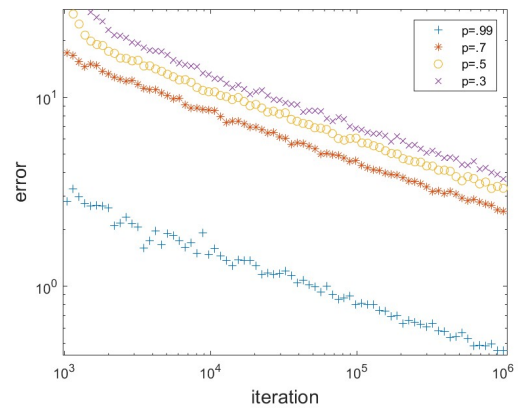

Figure 1: Step size is $p^2/(10\sqrt{t})$. $N = 10^6$ iterations are used. For all values of p, we see convergence, with larger values of p corresponding to faster convergence.

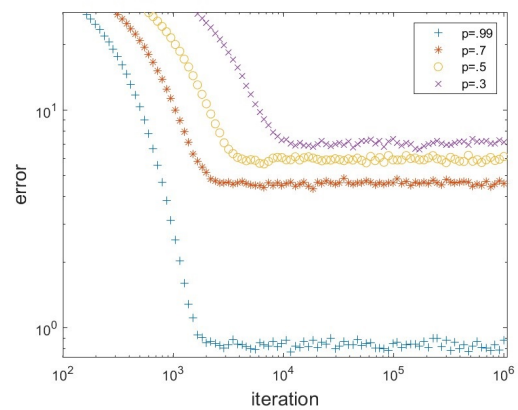

Figure 2: Step size is $p^2/3000$. $N = 10^6$ iterations are used. For all values of p, we see that a convergence horizon is quickly reached, with larger values of p corresponding to a smaller convergence horizon.

Conclusion

In this paper, we presented mSGDT, a stochastic iterative method to solve tensor linear systems with missing data. We showed two theoretical convergence results. With a step size proportional to $1/\sqrt{t}$, mSGDT converges to the correct solution, with faster progress for large values of p. With a constant step size, mSGDT quickly approaches a convergence horizon but does not converge to the desired solution. Our experiments on synthetic data support these two results.

References

Chen, X.; and Qin, J. 2021. Regularized Kaczmarz Algorithms for Tensor Recovery. *SIAM Journal on Imaging Sciences*, 14(4): 1439–1471.

Grotheer, R.; Li, S.; Ma, A.; Needell, D.; and Qin, J. 2024. Iterative singular tube hard thresholding algorithms for tensor recovery. *Inverse Problems and Imaging*, 18(4): 889–907.

Hu, W.; Tao, D.; Zhang, W.; Xie, Y.; and Yang, Y. 2017. The Twist Tensor Nuclear Norm for Video Completion. *IEEE Transactions on Neural Networks and Learning Systems*, 28(12): 2961–2973.

Kilmer, M. E.; and Martin, C. D. 2011. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3): 641–658. Special Issue: Dedication to Pete Stewart on the occasion of his 70th birthday.

Kolda, T. G.; and Hong, D. 2020. Stochastic gradients for large-scale tensor decomposition. *SIAM Journal on Mathematics of Data Science*, 2(4): 1066–1095.

Ma, A.; and Molitor, D. 2022. Randomized Kaczmarz for tensor linear systems. *BIT Numerical Mathematics*, 62(1): 171–194.

Ma, A.; and Needell, D. 2018. Stochastic Gradient Descent for Linear Systems with Missing Data. *Numerical Mathematics: Theory, Methods and Applications*, 12(1): 1–20.

Maehara, T.; Hayashi, K.; and Kawarabayashi, K.-i. 2016. Expected tensor decomposition with stochastic gradient descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30(1).

Needell, D.; Ward, R.; and Srebro, N. 2014. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Advances in neural information processing systems*, 27.

Newman, E.; Horesh, L.; Avron, H.; and Kilmer, M. 2018. Stable Tensor Neural Networks for Rapid Deep Learning. arXiv:1811.06569.

Papastergiou, T.; and Megalooikonomou, V. 2017. A distributed proximal gradient descent method for tensor completion. In *2017 IEEE International Conference on Big Data (Big Data)*, 2056–2065.

Shamir, O.; and Zhang, T. 2013. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. Proc. Int. Conf. Machine Learning, 71–79.

Yin, M.; Gao, J.; Xie, S.; and Guo, Y. 2019. Multiview Subspace Clustering via Tensorial t-Product Representation. *IEEE Transactions on Neural Networks and Learning Systems*, 30(3): 851–864.

Supplement

We collect a few lemmas that describe the constants or terms we use in our theorems, or are otherwise heavy in computation. Lemma 3 checks that $\mathbb{E}[g(\mathcal{X})] = \nabla F(\mathcal{X})$. Lemma 4 gives a bound on $\mathbb{E}[\|g(\mathcal{X})\|^2]$, Lemma 5 gives a bound on $\mathbb{E}[\|g(\mathcal{X}_\star)\|^2]$, Lemma 6 gives a Lipschitz constant for g, and Lemma 7 gives a description for the strongly convex parameter of the objective function F. Lemma 8 verifies $\nabla f(\mathcal{X}) = g(\mathcal{X})$.

Lemma 3. *We have that* $\mathbb{E}[g(\mathcal{X})] = \nabla F(\mathcal{X})$.

Proof. One can verify that $\nabla F(\mathcal{X}) = \frac{1}{m}\mathcal{A}^* * (\mathcal{A} * \mathcal{X} - \mathcal{B})$. Hence, we want to show that

$$\mathbb{E}[g(\mathcal{X})] = \mathbb{E}_i[\mathbb{E}_\delta[g(\mathcal{X})]] = \frac{1}{m}\mathcal{A}^* * (\mathcal{A} * \mathcal{X} - \mathcal{B}). \quad (2)$$

We first focus on the expectation with respect to the binary mask. Given a row index i, denote the frontal slices of the ith row slice of $\mathcal{A}$ (resp. $\tilde{\mathcal{A}}$) by $a_0, a_1, \ldots, a_{n-1}$ (resp. $\tilde{a}_0, \tilde{a}_1, \ldots, \tilde{a}_{n-1}$). Note that these are vectors, hence the lowercase notation. Observe that

$$\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::} = \text{fold}\begin{pmatrix} \tilde{a}_0^*\tilde{a}_0 + \tilde{a}_1^*\tilde{a}_1 + \cdots + \tilde{a}_{n-1}^*\tilde{a}_{n-1} \\ \tilde{a}_{n-1}^*\tilde{a}_0 + \tilde{a}_0^*\tilde{a}_1 + \cdots + \tilde{a}_{n-2}^*\tilde{a}_{n-1} \\ \vdots \\ \tilde{a}_1^*\tilde{a}_0 + \tilde{a}_2^*\tilde{a}_1 + \cdots + \tilde{a}_0^*\tilde{a}_{n-1} \end{pmatrix}.$$

Inside the fold $(\cdot)$, we have many outer products of vectors, which are easy to analyze. In general,

$$\mathbb{E}_\delta[(\tilde{a}_j^*\tilde{a}_k)_{xy}] = \begin{cases} p^2(a_j^*a_k)_{xy}, & j \neq k \text{ or } x \neq y \\ p(a_j^*a_k)_{xy}, & j = k \text{ and } x = y. \end{cases}$$

Therefore,

$$\mathbb{E}_\delta[\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}] = p^2\mathcal{A}^*_{i::} * \mathcal{A}_{i::} + (p - p^2)\text{fdiag}\left(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}\right),$$

where one should think of $\text{fdiag}\left(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}\right)$ as a correction term for those cases of x, y where $\mathbb{E}_\delta[(\tilde{a}_j^*\tilde{a}_k)_{xy}]$ equals $p(a_j^*a_k)_{xy}$ instead of $p^2(a_j^*a_k)_{xy}$. Then,

$$\begin{aligned} &\mathbb{E}_\delta[\tilde{\mathcal{A}}^*_{i::} * (\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::})] \\ =&(p^2\mathcal{A}^*_{i::} * \mathcal{A}_{i::} + (p - p^2)\text{fdiag}\left(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}\right)) * \mathcal{X} \\ &- p\mathbb{E}_\delta[\tilde{\mathcal{A}}^*_{i::} * \mathcal{B}_{i::}] \\ =&(p^2\mathcal{A}^*_{i::} * \mathcal{A}_{i::} + (p - p^2)\text{fdiag}\left(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}\right)) * \mathcal{X} \\ &- p^2\mathcal{A}_{i::} * \mathcal{B}_{i::}. \end{aligned}$$

After noticing that

$$\mathbb{E}_\delta\left[\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right)\right] = p\,\text{fdiag}\left(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}\right),$$

it readily follows that

$$\mathbb{E}_\delta[g(\mathcal{X})] = \mathcal{A}^*_{i::} * (\mathcal{A}_{i::} * \mathcal{X} - \mathcal{B}_{i::}).$$

Finally, one can check that equation (2) follows. □

Lemma 4. $\mathbb{E}[\|g(\mathcal{X})\|^2] \leq G$, *where* $G = \frac{2R^2(n^2+(1-p)^2n)}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4 + \frac{4n^{3/2}R}{mp}\sum_i \|\mathcal{A}_{i::}\|^3\|\mathcal{B}_{i::}\| + \frac{2n}{m}\sum_i \|\mathcal{A}_{i::}\|^2\|\mathcal{B}_{i::}\|^2$.

Proof. We have

$$
\begin{aligned}
\mathbb{E}[\|g(\mathcal{X})\|^2] &= \mathbb{E}\Bigg[\Bigg\|\frac{1}{p^2}\left(\tilde{\mathcal{A}}^*_{i::} * (\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::})\right) \\
&\quad - \frac{(1-p)}{p^2}\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\Bigg\|^2\Bigg] \\
&\le \frac{2}{p^4}\mathbb{E}\left[\left\|\tilde{\mathcal{A}}^*_{i::} * (\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::})\right\|^2\right] \\
&\quad + \frac{2(1-p)^2}{p^4}\mathbb{E}\left[\|\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\|^2\right] \\
&\le \frac{2n}{p^4}\mathbb{E}\left[\|\tilde{\mathcal{A}}_{i::}\|^2\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2\right] \\
&\quad + \frac{2(1-p)^2}{p^4}\mathbb{E}\left[\|\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\|^2\right] \\
&\le \frac{2n}{p^4}\mathbb{E}\left[\|\mathcal{A}_{i::}\|^2\,\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2\right] \\
&\quad + \frac{2(1-p)^2}{p^4}\mathbb{E}\left[\|\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\|^2\right] \\
&= \frac{2n}{p^4}\mathbb{E}_i\big[\,\|\mathcal{A}_{i::}\|^2\,\underbrace{\mathbb{E}_\delta[\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2]}_{\text{(A)}}\big] \\
&\quad + \frac{2(1-p)^2}{p^4}\mathbb{E}_i\Bigg[\underbrace{\mathbb{E}_\delta\left[\|\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\|^2\right]}_{\text{(B)}}\Bigg].
\end{aligned} \tag{3}
$$

Let's compute (A) first. We first rewrite (A) to have $\|\mathcal{A}_{i::}\mathcal{X} - p\mathcal{B}_{i::}\|^2$ as a term. Let $\mathcal{U} \in \mathbb{R}^{\ell\times 1\times n}, \mathcal{V} \in \mathbb{R}^{1\times 1\times n}$ be tensors. Since the frontal slices of $\mathcal{U}$ (resp. $\mathcal{V}$) are vectors (resp. scalars), we denote them by lowercase notation $u_0, \ldots, u_{n-1}$ (resp. $v_0, \ldots, v_{n-1}$). Observe

$$
\begin{aligned}
&\|\tilde{\mathcal{A}}_{i::} * \mathcal{U} - p\mathcal{V}\|^2 \\
&= \left\|\begin{pmatrix} \tilde{a}_0 & \ldots & \tilde{a}_1 \\ \vdots & \ddots & \vdots \\ \tilde{a}_{n-1} & \ldots & \tilde{a}_0 \end{pmatrix}\begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1}\end{pmatrix} - p\begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1}\end{pmatrix}\right\|^2 \\
&= \sum_{k=0}^{n-1}(\tilde{a}_k u_0 + \cdots + \tilde{a}_{k+1}u_{n-1} - pv_k)^2.
\end{aligned}
$$

When we take the expectation of the expansion of this sum over δ, we see that every term has a coefficient of p^2, except for the terms $(\tilde{a}_k u_0)^2, \ldots, (\tilde{a}_{k+1}u_{n-1})^2$, which have a coefficient of p. Therefore,

$$
\begin{aligned}
&\mathbb{E}_\delta[\|\tilde{\mathcal{A}}_{i::} * \mathcal{U} - p\mathcal{V}\|^2] \\
&\quad = p^2\|\mathcal{A}_{i::} * \mathcal{U} - p\mathcal{V}\|^2 \\
&\qquad + (p-p^2)\sum_{k=0}^{n-1}\left[(a_k u_0)^2 + \cdots + (a_{k+1}u_{n-1})^2\right] \\
&\quad \le p^2\|\mathcal{A}_{i::} * \mathcal{U} - p\mathcal{V}\|^2 + (p-p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{U}\|^2.
\end{aligned}
$$

Hence, setting $\mathcal{U}$ to be a jth column slice of $\mathcal{X}$ and ranging over j,

$$
\begin{aligned}
&\mathbb{E}_\delta[\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2] \\
&= \sum_{j=0}^{\ell-1}\mathbb{E}_\delta[\|\tilde{\mathcal{A}}_{i::} * \mathcal{X}_{:j:} - p\mathcal{B}_{ij:}\|^2] \\
&\le \sum_{j=0}^{\ell-1} p^2\|\mathcal{A}_{i::} * \mathcal{X}_{:j:} - p\mathcal{B}_{ij:}\|^2 + (p-p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{X}_{:j:}\|^2 \\
&= p^2\|\mathcal{A}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2 + (p-p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{X}\|^2.
\end{aligned}
$$

Then, by expanding some terms,

$$
\begin{aligned}
&\mathbb{E}_\delta[\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2] \\
&\quad \le p^2\|\mathcal{A}_{i::} * \mathcal{X}\|^2 - 2p^3\langle \mathcal{A}_{i::} * \mathcal{X}, \mathcal{B}_{i::}\rangle \\
&\qquad + p^4\|\mathcal{B}_{i::}\|^2 + (p-p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{X}\|^2 \\
&\quad \le (p^2 n + p - p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{X}\|^2 \\
&\qquad + 2p^3\sqrt{n}\|\mathcal{A}_{i::}\|\|\mathcal{X}\|\|\mathcal{B}_{i::}\| + p^4\|\mathcal{B}_{i::}\|^2 \\
&\quad \le pnR^2\|\mathcal{A}_{i::}\|^2 + 2p^3\sqrt{n}R\|\mathcal{A}_{i::}\|\|\mathcal{B}_{i::}\| + p^4\|\mathcal{B}_{i::}\|^2.
\end{aligned}
$$

Denote diag $(\cdot)$ to be the diagonal matrix that has the same diagonal elements as the input matrix. For (B), we note

$$
\begin{aligned}
&\mathbb{E}_\delta\left[\|\text{fdiag}\left(\tilde{\mathcal{A}}^*_{i::} * \tilde{\mathcal{A}}_{i::}\right) * \mathcal{X}\|^2\right] \\
&= p\,\|\text{fdiag}\,(\mathcal{A}^*_{i::} * \mathcal{A}_{i::}) * \mathcal{X}\|^2 \\
&\le pn\|\text{fdiag}\,(\mathcal{A}^*_{i::} * \mathcal{A}_{i::})\,\|^2\|\mathcal{X}\|^2 \\
&\le pn\|\text{diag}\left(a_0^*a_0 + a_1^*a_1 + \cdots + a_{n-1}^*a_{n-1}\right)\|^2 R^2 \\
&\le pn\left(\sum_k a_k^2\right)^2 R^2 \\
&= pn(\|\mathcal{A}_{i::}\|^2)^2R^2 = pn\|\mathcal{A}_{i::}\|^4R^2.
\end{aligned} \tag{4}
$$

Thus,

$$
\begin{aligned}
&\mathbb{E}\left[\|g(\mathcal{X})\|^2\right] \\
&\le \frac{2n}{p^4}\mathbb{E}_i[\|\mathcal{A}_{i::}\|^2(pnR^2\|\mathcal{A}_{i::}\|^2 \\
&\quad + 2p^3\sqrt{n}R\|\mathcal{A}_{i::}\|\|\mathcal{B}_{i::}\| + p^4\|\mathcal{B}_{i::}\|^2)] \\
&\quad + \frac{2(1-p)^2}{p^4}\mathbb{E}_i\left[pn\|\mathcal{A}_{i::}\|^4R^2\right] \\
&= \frac{2n^2R^2}{mp^3}\sum_i\|\mathcal{A}_{i::}\|^4 + \frac{4n^{3/2}R}{mp}\sum_i\|\mathcal{A}_{i::}\|^3\|\mathcal{B}_{i::}\| \\
&\quad + \frac{2n}{m}\sum_i\|\mathcal{A}_{i::}\|^2\|\mathcal{B}_{i::}\|^2 + \frac{2(1-p)^2nR^2}{mp^3}\sum_i\|\mathcal{A}_{i::}\|^4 \\
&= \frac{2R^2(n^2+(1-p)^2n)}{mp^3}\sum_i\|\mathcal{A}_{i::}\|^4 \\
&\quad + \frac{4n^{3/2}R}{mp}\sum_i\|\mathcal{A}_{i::}\|^3\|\mathcal{B}_{i::}\| + \frac{2n}{m}\sum_i\|\mathcal{A}_{i::}\|^2\|\mathcal{B}_{i::}\|^2.
\end{aligned}
$$

□

Lemma 5. $\mathbb{E}[\|g(\mathcal{X}_\star)\|^2] \leq G_\star$, *where* $G_\star = \frac{2(1-p)(2-p)nR^2}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4$.

Proof. In Lemma 4 we obtained (3), and we later found that (A) could be rewritten as $p^2\|\mathcal{A}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2 + (p - p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{X}\|^2$. For (B), we again use (4), which bounds (B) by $pn\|\mathcal{A}_{i::}\|^4 R^2$.

Using (3) by substituting in $\mathcal{X} = \mathcal{X}_\star$ so that $\|\mathcal{A}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2$ becomes 0, we deduce

$$\begin{aligned}
&\mathbb{E}[\|g(\mathcal{X}_\star)\|^2] \\
&\quad\leq \frac{2n}{p^4}\mathbb{E}_i\left[(p-p^2)\|\mathcal{A}_{i::}\|^2\|\mathcal{A}_{i::}\|^2\|\mathcal{X}_\star\|^2\right] \\
&\qquad + \frac{2(1-p)^2nR^2}{p^3}\mathbb{E}_i[\|\mathcal{A}_{i::}\|^4] \\
&\quad\leq \frac{2(1-p)n}{p^3}\mathbb{E}_i\left[\|\mathcal{A}_{i::}\|^4\|\mathcal{X}_\star\|^2\right] \\
&\qquad + \frac{2(1-p)^2nR^2}{p^3}\mathbb{E}_i[\|\mathcal{A}_{i::}\|^4] \\
&\quad\leq \frac{2(1-p)nR^2}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4 \\
&\qquad + \frac{2(1-p)^2nR^2}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4 \\
&\quad= \frac{2(1-p)(2-p)nR^2}{mp^3}\sum_i \|\mathcal{A}_{i::}\|^4.
\end{aligned}$$

□

Lemma 6. *The update function $g(\mathcal{X})$ of Algorithm 1 is Lipschitz continuous, with the supremum of the Lipschitz constant over all possible binary masks bounded above by $L_g := na_{max}^2/p^2$, where a_{max} is the maximum Frobenius norm of a row slice of $\mathcal{A}$.*

Proof. We have

$$\begin{aligned}
&\|g(\mathcal{X}) - g(\mathcal{Y})\| \\
&= \left\| \left(\frac{1}{p^2}\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::} - \frac{1-p}{p^2}\text{fdiag}\left(\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::}\right)\right) \right. \\
&\qquad \left. * (\mathcal{X} - \mathcal{Y})\right\| \\
&\leq \frac{\sqrt{n}}{p^2}\left\|\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::} - (1-p)\text{fdiag}\left(\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::}\right)\right\| \\
&\qquad \|\mathcal{X} - \mathcal{Y}\| \\
&\leq \frac{\sqrt{n}}{p^2}\left\|\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::}\right\| \|\mathcal{X} - \mathcal{Y}\| \\
&\leq \frac{n}{p^2}\|\tilde{\mathcal{A}}_{i::}\|^2\|\mathcal{X} - \mathcal{Y}\| \\
&\leq \frac{n}{p^2}\|\mathcal{A}_{i::}\|^2\|\mathcal{X} - \mathcal{Y}\| \\
&= \frac{na_{max}^2}{p^2}\|\mathcal{X} - \mathcal{Y}\|.
\end{aligned}$$

□

The next lemma describes the strongly convex parameter of the objective function in terms of bdiag $\left(\widehat{\mathcal{A}}\right)$, where $\widehat{\mathcal{A}}$ is the tensor obtained by applying the discrete Fourier transform to the tubes of $\mathcal{A}$, and bdiag $(\cdot)$ returns the block diagonal matrix formed by the frontal slices of the input tensor. A *tube* of a tensor is obtained by fixing the first two indices and varying the third index. We view tubes as (column) vectors in $\mathbb{R}^n$. We denote the (i,j)th tube of $\mathcal{A}$ as $\mathcal{A}_{ij:}$.

Lemma 7. *Recall our objective function $F(\mathcal{X}) = \frac{1}{2m}\|\mathcal{A} * \mathcal{X} - \mathcal{B}\|^2$. Let $\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*$ be the T-SVD of $\mathcal{A}$ (Kilmer and Martin 2011). Assume that $\mathcal{A}$ is tall, i.e. $m > \ell$. Let σ_{min} be the smallest singular value of bdiag $\left(\widehat{\mathcal{A}}\right)$. Then F is σ_{min}^2/m-strongly convex.*

Proof. We have

$$\begin{aligned}
&\langle \nabla F(\mathcal{X}) - \nabla F(\mathcal{Y}), \mathcal{X} - \mathcal{Y}\rangle \\
&= \langle \frac{1}{m}\mathcal{A}^* * \mathcal{A} * (\mathcal{X} - \mathcal{Y}), \mathcal{X} - \mathcal{Y}) \\
&= \frac{1}{m}\langle \mathcal{A} * (\mathcal{X} - \mathcal{Y}), \mathcal{A} * (\mathcal{X} - \mathcal{Y})\rangle \\
&= \frac{1}{m}\|\mathcal{A} * (\mathcal{X} - \mathcal{Y})\|^2 \\
&= \frac{1}{m}\|\mathcal{U} * \mathcal{S} * \mathcal{V}^* * (\mathcal{X} - \mathcal{Y})\|^2 \\
&= \frac{1}{m}\|\mathcal{S} * \mathcal{V}^* * (\mathcal{X} - \mathcal{Y})\|^2.
\end{aligned}$$

Since $\mathcal{V}$ is orthogonal, $\|\mathcal{V}^* * (\mathcal{X} - \mathcal{Y})\| = \|\mathcal{X} - \mathcal{Y}\|$. So we will derive a bound on the norm of $\mathcal{S} * \mathcal{Z}$, where $\mathcal{Z} \in \mathbb{R}^{\ell\times q\times n}$ is such that $\|\mathcal{Z}\| = \|\mathcal{X} - \mathcal{Y}\|$. We will do so by examining each tube of $\mathcal{S} * \mathcal{Z}$.

The kth frontal slice of $\mathcal{S} * \mathcal{Z}$ is $\mathbf{S}_k\mathbf{Z}_0 + \mathbf{S}_{k-1}\mathbf{Z}_1 + \cdots + \mathbf{S}_{k+1}\mathbf{Z}_{n-1}$. For all $0 \leq i < \ell, 0 \leq j < q$, we then have

$$\begin{aligned}
&(\mathcal{S}*\mathcal{Z})_{ijk} \\
&\quad= (\mathbf{S}_k\mathbf{Z}_0)_{ij} + (\mathbf{S}_{k-1}\mathbf{Z}_1)_{ij} + \cdots + (\mathbf{S}_{k+1}\mathbf{Z}_{n-1})_{ij} \\
&\quad= \mathcal{S}_{iik}\mathcal{Z}_{ij0} + \mathcal{S}_{ii(k-1)}\mathcal{Z}_{ij1} + \cdots + \mathcal{S}_{ii(k+1)}\mathcal{Z}_{ij(n-1)} \\
&\quad= \begin{pmatrix}\mathcal{S}_{iik} & \mathcal{S}_{ii(k-1)} & \dots & \mathcal{S}_{ii(k+1)}\end{pmatrix}\mathcal{Z}_{ij:}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
(\mathcal{S} * \mathcal{Z})_{ij:} &= \begin{pmatrix} \mathcal{S}_{ii0} & \mathcal{S}_{ii(n-1)} & \dots & \mathcal{S}_{ii1} \\ \mathcal{S}_{ii1} & \mathcal{S}_{ii0} & \dots & \mathcal{S}_{ii2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{S}_{ii(n-1)} & \mathcal{S}_{ii(n-2)} & \dots & \mathcal{S}_{ii0} \end{pmatrix}\mathcal{Z}_{ij:} \\
&= \text{circ}\left(\mathcal{S}_{ii:}\right)\mathcal{Z}_{ij:}.
\end{aligned}$$

Let $\sigma(\cdot)$ denote the singular values of the input matrix, and let $\sigma_{min} = \min_{0\leq i<\ell}\sigma(\mathcal{S}_{ii:})$. By the above calculation, we deduce

$$\|(\mathcal{S} * \mathcal{Z})_{ij:}\|_F^2 \geq \sigma_{min}^2\|\mathcal{Z}_{ij:}\|_F^2.$$

Summing over $0 \leq i < \ell$ and $0 \leq j < q$ yields

$$\|\mathcal{S} * \mathcal{Z}\|^2 \geq \sigma_{min}^2\|\mathcal{Z}\|^2,$$

which means that F is σ_{min}^2/m-strongly convex.

It suffices to show that the singular values of the circ $(\mathcal{S}_{ii:})$ are the singular values of bdiag $\left(\widehat{\mathcal{A}}\right)$. Let

$$\Sigma = \bigcup_{0 \le i < \ell} \sigma(\text{circ}\,(\mathcal{S}_{ii:})).$$

It is well-known that circulant matrices can be diagonalized by the DFT. In particular, we get that circ $(\mathcal{S}_{ii:})$ is unitarily similar to diag $(\text{dft}\,(\mathcal{S}_{ii:}))$. Thus the singular values of circ $(\mathcal{S}_{ii:})$ are equal to the norms of the entries in dft $(\mathcal{S}_{ii:})$.

In other words, the singular values of all the circ $(\mathcal{S}_{ii:})$ are all the norms of the diagonal entries of

$$\text{bdiag}\left(\widehat{\mathcal{S}}\right) = \begin{pmatrix} \widehat{\mathbf{S}}_0 & & & \\ & \widehat{\mathbf{S}}_1 & & \\ & & \ddots & \\ & & & \widehat{\mathbf{S}}_{n-1} \end{pmatrix},$$

Since bdiag $\left(\widehat{\mathcal{S}}\right)$ is itself a diagonal matrix, we deduce that

$$\Sigma = \sigma(\text{bdiag}\left(\widehat{\mathcal{S}}\right)).$$

From (Kilmer and Martin 2011, Fact 2), with $\otimes$ being the Kronecker product,

$$\text{bdiag}\left(\widehat{\mathcal{S}}\right) = (\mathbf{F}_n \otimes \mathbf{I}_m)\text{bcirc}\,(\mathcal{S})\,(\mathbf{F}_n^* \otimes \mathbf{I}_l),$$

so

$$\sigma(\text{bdiag}\left(\widehat{\mathcal{S}}\right)) = \sigma(\text{bcirc}\,(\mathcal{S})).$$

Moreover,

$$\begin{aligned} &\text{bdiag}\left(\widehat{\mathcal{A}}\right) \\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\text{bcirc}\,(\mathcal{A})\,(\mathbf{F}_n^* \otimes \mathbf{I}_l) \\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\text{bcirc}\,(\mathcal{U} * \mathcal{S} * \mathcal{V}^*)\,(\mathbf{F}_n^* \otimes \mathbf{I}_l) \\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\text{bcirc}\,(\mathcal{U})\,\text{bcirc}\,(\mathcal{S})\,\text{bcirc}\,(\mathcal{V}^*)\,(\mathbf{F}_n^* \otimes \mathbf{I}_l) \end{aligned}$$

since bcirc $(\cdot)$ is multiplicative (Ma and Molitor 2022, Fact 1). Finally, because bcirc $(\mathcal{U})$ and bcirc $(\mathcal{V}^*)$ are orthogonal (Ma and Molitor 2022, Lemma 4), we deduce that, as desired:

$$\Sigma = \sigma\left(\text{bdiag}\left(\widehat{\mathcal{S}}\right)\right) = \sigma(\text{bcirc}\,(\mathcal{S})) = \sigma\left(\text{bdiag}\left(\widehat{\mathcal{A}}\right)\right).$$

□

Lemma 8. *With* $f(\mathcal{X}) = \frac{1}{2p^2}\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2 - \frac{1-p}{2p^2}\|\sqrt{\textit{fdiag}\left(\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::}\right)} * \mathcal{X}\|^2$, *we have* $\nabla f(\mathcal{X}) = g(\mathcal{X})$.

Proof. Recall that $\nabla(\frac{1}{2}\|\mathcal{U} * \mathcal{V} - \mathcal{Z}\|^2) = \mathcal{U}^* * (\mathcal{U} * \mathcal{V} - \mathcal{Z})$ for any tensors $\mathcal{U}, \mathcal{V}, \mathcal{Z}$. Hence

$$\nabla\left(\frac{1}{2p^2}\|\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}\|^2\right) = \frac{1}{p^2}\tilde{\mathcal{A}}_{i::}^* * (\tilde{\mathcal{A}}_{i::} * \mathcal{X} - p\mathcal{B}_{i::}). \tag{5}$$

For the second term in $f(\mathcal{X})$, let $\mathcal{U} = \text{fdiag}\left(\tilde{\mathcal{A}}_{i::}^* * \tilde{\mathcal{A}}_{i::}\right)$. Since the only nonzero frontal slice of $\mathcal{U}$ is potentially its zeroth one, we deduce

$$\sqrt{\mathcal{U}}^* * \sqrt{\mathcal{U}} = \mathcal{U}.$$

Thus,

$$\begin{aligned} \nabla\left(\frac{1-p}{2p^2}\|\sqrt{\mathcal{U}} * \mathcal{X}\|^2\right) &= \frac{1-p}{p^2}\sqrt{\mathcal{U}}^* * \sqrt{\mathcal{U}} * \mathcal{X} \\ &= \frac{1-p}{p^2}\mathcal{U} * \mathcal{X}. \end{aligned} \tag{6}$$

Combining (5) and (6) yields $\nabla f(\mathcal{X}) = g(\mathcal{X})$.

□