A Tensor SVD-based Classification Algorithm Applied to fMRI Data

Katherine Keegan[†], Tanvi Vishwanath[‡], and Yihua Xu[§]

Project advisors: Dr. Elizabeth Newman[¶], Vida John[‖]

Abstract. To analyze the abundance of multidimensional data, tensor-based frameworks have been developed. Traditionally, the matrix singular value decomposition (SVD) is used to extract the most dominant features from a matrix containing the vectorized data. While the SVD is highly useful for data that can be appropriately represented as a matrix, this step of vectorization causes us to lose the high-dimensional relationships intrinsic to the data. To facilitate efficient multidimensional feature extraction, we utilize a projection-based classification algorithm using the t-SVDM, a tensor analog of the matrix SVD. Our work extends the t-SVDM framework and the classification algorithm, both initially proposed for tensors of order 3, to any number of dimensions. We then apply this algorithm to a classification task using the StarPlus fMRI dataset. Our numerical experiments demonstrate that there exists a superior tensor-based approach to fMRI classification than the best possible equivalent matrix-based approach. Our results illustrate the advantages of our chosen tensor framework, provide insight into beneficial choices of parameters, and could be further developed for classification of more complex imaging data. We provide our Python implementation at https://github.com/elizabethnewman/tensor-fmri.

1. Introduction.

High-dimensional data has become increasingly prevalent in various fields such as video, semantic indexing, and chemistry [12]. One such area is medical imaging, in which the high-dimensional data of interest are medical images obtained using a modality such as functional Magnetic Resonance Imaging (fMRI). A single patient's fMRI scan can be thought of as a series of three-dimensional scans over time [1, 2].

fMRI data can serve an important role in disease detection, with applications ranging from Alzheimer's to depression [5, 15]. Human medical professionals can be trained to interpret fMRI scans and make diagnostic decisions based on visual inspection. Various researchers have investigated the viability of utilizing mathematical methods to automate this classification process. The key to classification algorithms is to extract meaningful features from training data belonging to a certain class. The Singular Value Decomposition (SVD) is one popular method that accomplishes this task successfully in various data analysis contexts. However, it requires that all of the input data is reshaped into a two-dimensional form.

A tensor is a higher-dimensional analog to a matrix. Unlike a matrix in traditional linear algebra, where all information is indexed according to two possible axes (rows and columns), a tensor can have arbitrarily many dimensions. The mathematical foundations of algebra using tensors have been well-developed; [12] provides some useful background on this subject. In order to identify dominant features from inherently multidimensional data such as fMRI scans, tensor-based SVD analogs have been developed that allow the data to retain its multi-dimensional form throughout the classification process, avoiding the pitfalls that occur with

[†]Department of Mathematics, Mary Baldwin University, Staunton, VA (keegank1624@marybaldwin.edu).
[‡]Department of Mathematics, Texas A & M University, College Station, TX (tanvivishwanath@tamu.edu).
[§]Department of Mathematics, Georgia Institute of Technology, Atlanta, GA (yxu604@gatech.edu).
[¶]Department of Mathematics, Emory University, Atlanta, GA (elizabeth.newman@emory.edu).
[‖]Tesselations School, Cupertino, CA

vectorizing such data [8].

We focus on the t-SVDM framework, a tensor SVD approach proposed in [10]. This particular framework not only allows for a kind of multiplication between tensors that appears similar to traditional matrix multiplication, but also can provide representations of multidimensional data that are provably better than representing the data in vectorized form as a matrix. In order to determine its viability in more complicated medical diagnostic classification tasks, we apply this framework to a classification task of determining whether or not a human subject is reading a sentence or viewing a picture. We illustrate that this t-SVDM classification algorithm successfully beats its matrix counterpart. Our results also shed some light on the potential limitations of this method.

In our paper, we illustrate an extension of the t-SVDM framework, describe how it can be used for a classification algorithm based on [18], and we apply this algorithm to the aforementioned fMRI classification task. The main extensions and contributions of our work are the following:

- **Dimension Extension:** The original t-SVDM framework is proposed for only three-dimensions. Our paper provides definitions for a p-dimensional tensor and illustrates the usability of this framework for a five-dimensional fMRI dataset.
- **Transformation Choices:** We select the t-SVDM not only for its ability to process high-dimensional data, but also for the flexibility that the framework introduces via the $\star_{\mathrm{M}}$-product, which enables one to strategically choose a mathematical transformation based on the nature of the data being analyzed.
- **Algorithm Flexibility:** The t-SVDM and our proposed classification procedure is a mathematically justified framework that can be applied to any labeled high-dimensional data. Thus, all of the methods described in our paper can easily be extended to other similar classification tasks with labeled data.
- **Region of Interest Implications:** When incorporating knowledge about specific regions of the brain into our classification procedure, we discovered that the most impactful regions vary between human subjects. This could illustrate the anatomical differences between humans when completing cognitive tasks and highlights the challenge of constructing the difficulty of creating a universal basis that represents all subjects with consistent accuracy.

The paper is organized as follows: Section 2 describes necessary background notations, the $\star_{\mathrm{M}}$-product and the algebraic framework it gives rise to, and the t-SVDM. Section 3 describes the local tensor SVD approaches specifically for classification and provides an intuition example to provide more explanation. Section 4 first describes specific choices of transformation matrices, and then examines numerical results from applying the classification algorithm to the StarPlus fMRI dataset. Finally, Section 5 concludes the paper and proposes some potential future directions.

2. Background and Preliminaries.

In this paper, a *tensor*, denoted with a capital calligraphic letter $\boldsymbol{\mathcal{A}}$, is a multidimensional array. The *order* of a tensor is the number of dimensions it has; if $\boldsymbol{\mathcal{A}}$ is a tensor of order-p, then the size of $\boldsymbol{\mathcal{A}}$ is $n_1 \times n_2 \times \cdots \times n_p$. We assume tensors are real-valued; that is, $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$. A *matrix* (order-2 tensor) is denoted with a bold capital letter, $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$, a *vector* (order-1 tensor) is denoted with a bold lowercase

letter, $\mathbf{a} \in \mathbb{R}^{n_1}$, and a *scalar* (order-0 tensor) is denoted by a lowercase letter, $a \in \mathbb{R}$. To provide relevant information needed to understand the t-SVDM framework, we will first denote the basic notation and definitions used within this paper. Next, we will introduce tensor products including $\star_{\mathrm{M}}$-product. This notation and terminology will then equip us with the tools needed to explain the t-SVDM and the important properties it offers for classification. For simplicity, we consider real-valued tensors, but the definitions and theory presented can be extended to complex-valued tensors as well.

2.1. Notation. Consider that an $m \times n$ matrix is structured with m rows and n columns. Using MATLAB notation, $\mathbf{A}(i,:)$ denotes the i-th row and $\mathbf{A}(:,j)$ denotes the j-th column. We can similarly describe the structure of a tensor. These analogs of matrix concepts to tensors are introduced and developed in [11] and [8]. We extend these definitions such that they describe any finite-dimensional tensor of order p.

Definition 2.1 (mode-k fibers). *Fibers are sections of a tensor $\boldsymbol{\mathcal{A}}$ such that all but the k-th dimension are fixed.*

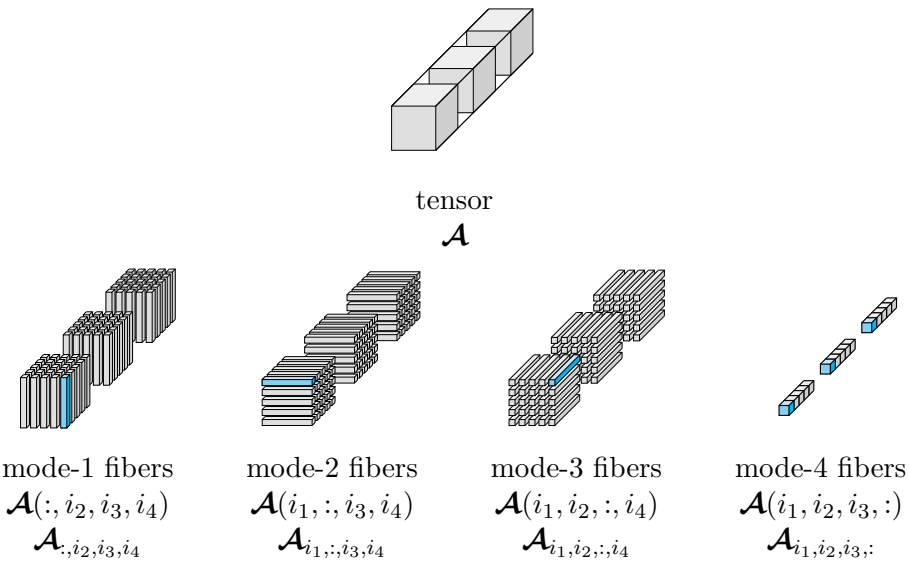
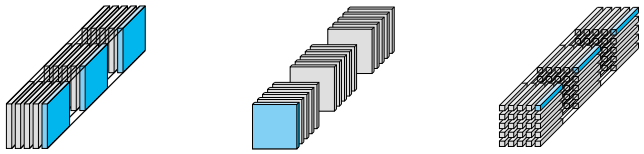


Figure 1: Visualization of a 4D tensor, its mode-1, 2, 3, and 4 fibers. For illustrative purposes, we only show a mode-4 fiber for $1 \times 1 \times n_3 \times n_4$ tensor.

Definition 2.2 (lateral slices, frontal slices, and tubes). *For a p-dimensional tensor $\boldsymbol{\mathcal{A}}$, we denote lateral slices as $\vec{\boldsymbol{\mathcal{A}}}_{i_2} = \boldsymbol{\mathcal{A}}(:, i_2, :, \ldots, :)$, and frontal slices as $\boldsymbol{\mathcal{A}}(:, :, i_3, i_4, \ldots, i_p)$. We denote tubes, or $1 \times 1 \times n_3 \times \cdots \times n_p$ tensors, as $\mathbf{a}_{i_1,i_2} = \boldsymbol{\mathcal{A}}(i_1, i_2, :, \ldots, :)$.*

Figure 2 visualizes a tensor of order 4, its lateral and frontal slices, and tubes.

Definition 2.3 (vectorize). *Matrix vectorization is the process of converting a matrix of*



lateral slices	frontal slices	tubes
$\vec{\boldsymbol{\mathcal{A}}}_{i_2} = \boldsymbol{\mathcal{A}}(:, i_2, :, :)$	$\boldsymbol{\mathcal{A}}(:, :, i_3, i_4)$	$\boldsymbol{\mathcal{A}}(i_1, i_2, :, :)$
$\boldsymbol{\mathcal{A}}_{:,i_2,\ldots}$	$\boldsymbol{\mathcal{A}}_{:,:,i_3,i_4}$	$\mathbf{a}_{i_1,i_2}$

Figure 2: Visualization of a 4D tensor, its lateral slices, frontal slices, and tubes. Notice that lateral slices fix the second dimension, frontal slices fix all dimensions except the first two, and tubes fix the first two dimensions.

values into a column vector. For example, for a matrix $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$*, we have*

$$\operatorname{vec}(\mathbf{A}) = \operatorname{vec}\left(\begin{bmatrix} \mathbf{A}_{:,1} & \cdots & \mathbf{A}_{:,n_2} \end{bmatrix}\right) = \begin{bmatrix} \mathbf{A}_{:,1} \\ \vdots \\ \mathbf{A}_{:,n_2} \end{bmatrix}$$

Tensor vectorization follows a similar recursive pattern.

$$\operatorname{vec}(\boldsymbol{\mathcal{A}}) = \begin{bmatrix} \operatorname{vec}(\boldsymbol{\mathcal{A}}_{:,:,1}) \\ \vdots \\ \operatorname{vec}(\boldsymbol{\mathcal{A}}_{:,:,n_3}) \end{bmatrix}, \text{ where } \boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$$

$$\operatorname{vec}(\boldsymbol{\mathcal{A}}) = \begin{bmatrix} \operatorname{vec}(\boldsymbol{\mathcal{A}}_{:,\ldots,:,1}) \\ \vdots \\ \operatorname{vec}(\boldsymbol{\mathcal{A}}_{:,\ldots,:,n_p}) \end{bmatrix}, \text{ where } \boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$$

Just as multiplication can be accomplished between matrices, one can also define multiplication between a tensor with a matrix. Definition 2.4 and Definition 2.5 describe this process.

Definition 2.4 (mode-k unfolding/folding). *A mode-k unfolding of a tensor* $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_p}$ *results in a matrix denoted by* $\boldsymbol{\mathcal{A}}_{(k)} \in \mathbb{R}^{n_k \times (n_1 \ldots n_{k-1} n_{k+1} \ldots n_p)}$ *such that the mode-k fibers are the columns of the resultant matrix. A mode-k folding is the reverse of this process.*

Definition 2.5 (mode-k product). *The mode-k product of a tensor* $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_p}$ *with a matrix* $\mathbf{M} \in \mathbb{R}^{d \times n_k}$ *results in a tensor whose mode-k unfolding is* $\mathbf{M}$ *multiplied with the mode-k unfolding of* $\boldsymbol{\mathcal{A}}$*, such that:*

$$\boldsymbol{\mathcal{A}} \times_k \mathbf{M} = \operatorname{fold}(\mathbf{M}\boldsymbol{\mathcal{A}}_{(k)}),$$

with $\operatorname{fold}(\mathbf{M}\boldsymbol{\mathcal{A}}_{(k)}) \in \mathbb{R}^{n_1 \times \cdots \times n_{k-1} \times d \times n_{k+1} \times n_p}$.

Definition 2.6 (Frobenius norm). *The Frobenius norm of an order-p tensor* $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_p}$ *is given by* $\|\boldsymbol{\mathcal{A}}\|_F = \sqrt{\sum_{i=1_1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_p=1}^{n_p} |\boldsymbol{\mathcal{A}}_{i_1,i_2,\ldots,i_p}|^2}$.

Definition 2.7 (f-diagonal). *A tensor is said to be facewise-diagonal, or f-diagonal, if its entries only lie along the diagonal of its frontal slices.*

2.2. Tensor-Tensor Products. In this section, we introduce the $\star_{\mathrm{M}}$-product and describe its flexibility, which gives rise to tensor analogies to various familiar matrix concepts such as the transpose, the identity, and orthogonality under $\star_{\mathrm{M}}$-product. Kilmer and Martin originated the method of multiplying matrices for the t-product in [11], and Kernfeld et al. extended this to the $\star_{\mathrm{M}}$-product in [8]. We describe the key definitions from these works for order-p tensors.

Definition 2.8 (Facewise Product). *The facewise product multiplies each of the frontal slices of two tensors in the transform domain in parallel. Given $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times m \times n_3 \times \cdots \times n_p}$, and $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{m \times n_2 \times n_3 \times \cdots \times n_p}$, the facewise product of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$, denoted using "$\triangle$" can be written as follows:*

$$\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{A}} \triangle \boldsymbol{\mathcal{B}}$$

where for each frontal slice of $\boldsymbol{\mathcal{C}}$

$$\boldsymbol{\mathcal{C}}(:,:,i_3,i_4,\ldots,i_p) = \boldsymbol{\mathcal{A}}(:,:,i_3,i_4,\ldots,i_p) \cdot \boldsymbol{\mathcal{B}}(:,:,i_3,i_4,\ldots,i_p),$$

for $i_k = 1,\ldots,n_k$, where $k = 3,\ldots,p$.

Now that we have defined both the mode-k product and the facewise product, we can introduce the $\star_{\mathrm{M}}$-product. Definition 2.9 defines the $\star_{\mathrm{M}}$-product, Algorithm 2.1 demonstrates its computation, Figure 3 demonstrates the product's computation for fourth-order tensors, and Example 2.10 illustrates a simple third-order example of computing the $\star_{\mathrm{M}}$-product.

Definition 2.9 ($\star_{\mathrm{M}}$-product). *Given $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times m \times n_3 \times \cdots \times n_p}$, and $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{m \times n_2 \times n_3 \times \cdots \times n_p}$, with invertible matrices $\mathbf{M}_3 \in \mathbb{R}^{n_3 \times n_3}, \ldots, \mathbf{M}_p \in \mathbb{R}^{n_p \times n_p}$, define $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ to be the $\star_{\mathrm{M}}$-product of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ such that*

$$\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{A}} \star_{\mathrm{M}} \boldsymbol{\mathcal{B}} = (\hat{\boldsymbol{\mathcal{A}}} \triangle \hat{\boldsymbol{\mathcal{B}}}) \times_3 \mathbf{M}_3^{-1} \times \cdots \times_p \mathbf{M}_p^{-1}$$

where

$$\hat{\boldsymbol{\mathcal{A}}} = \boldsymbol{\mathcal{A}} \times_3 \mathbf{M}_3 \times_4 \mathbf{M}_4 \times \cdots \times_p \mathbf{M}_p$$

.

Similar to how applying the Fourier Transform to a matrix of data will decouple relationships into frequencies, the purpose of the matrices $\mathbf{M}_3, \ldots, \mathbf{M}_p$ is to place a tensor in a transform domain where the higher-order relationships after the second dimension have been decoupled. The choice of $\mathbf{M}$ is left as a selectable parameter in the $\star_{\mathrm{M}}$-product. This introduces a means of flexibility into the framework in which one can strategically choose a transformation based on the nature of the data (e.g. the Discrete Fourier Transform for time series data). An additional important detail is that under the $\star_{\mathrm{M}}$-product, the lateral slices of a tensor are analogous to the columns of a matrix, and tubes are analogous to scalars.

Algorithm 2.1 $\star_{\mathrm{M}}$-product

Inputs: $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times m \times n_3 \times \cdots \times n_p}, \boldsymbol{\mathcal{B}} \in \mathbb{R}^{m \times n_2 \times n_3 \times \cdots \times n_p}$, invertible $\mathbf{M}_3 \in \mathbb{R}^{n_3 \times n_3}, \ldots, \mathbf{M}_p \in \mathbb{R}^{n_p \times n_p}$
$\hat{\boldsymbol{\mathcal{A}}} = \boldsymbol{\mathcal{A}}, \hat{\boldsymbol{\mathcal{B}}} = \boldsymbol{\mathcal{B}}$
for $k = 3, \ldots, p$ **do**
  $\hat{\boldsymbol{\mathcal{A}}} = \hat{\boldsymbol{\mathcal{A}}} \times_k \mathbf{M}_k$, $\hat{\boldsymbol{\mathcal{B}}} = \hat{\boldsymbol{\mathcal{B}}} \times_k \mathbf{M}_k$
end for
$\boldsymbol{\mathcal{C}} = (\hat{\boldsymbol{\mathcal{A}}} \triangle \hat{\boldsymbol{\mathcal{B}}})$
for $k = 3, \ldots, p$ **do**
  $\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{C}} \times_k \mathbf{M}_k^{-1}$
end for
Outputs: $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$

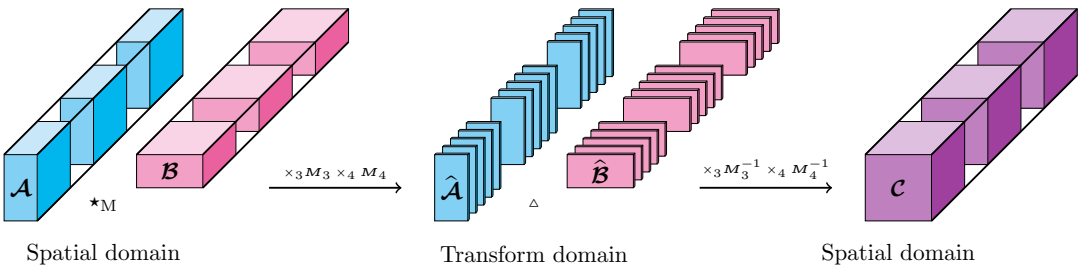


Figure 3: Visualization of $\star_{\mathrm{M}}$-product for 4D tensors

Example 2.10. *Given* $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{2\times 2\times 2}$ *and* $\vec{\boldsymbol{\mathcal{B}}} \in \mathbb{R}^{2\times 1\times 2}$ *where*

$$\boldsymbol{\mathcal{A}}(:,:,1) = \begin{bmatrix} 1 & 2 \\ 0 & -1 \end{bmatrix} \qquad \vec{\boldsymbol{\mathcal{B}}}(:,:,1) = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$
$$\boldsymbol{\mathcal{A}}(:,:,2) = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \qquad \vec{\boldsymbol{\mathcal{B}}}(:,:,2) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

We choose the following $\mathbf{M}$*:*

$$\mathbf{M} = \begin{bmatrix} 3 & 2 \\ 1 & 1 \end{bmatrix}, \qquad \mathbf{M}^{-1} = \begin{bmatrix} 1 & -2 \\ -1 & 3 \end{bmatrix}$$

We then move each tensor into the transform domain by taking the the mode-3 product of $\boldsymbol{\mathcal{A}}$ *and* $\vec{\boldsymbol{\mathcal{B}}}$ *with* $\mathbf{M}$*; that is,* $\hat{\boldsymbol{\mathcal{A}}} = \boldsymbol{\mathcal{A}} \times_3 \mathbf{M}$ *and* $\hat{\vec{\boldsymbol{\mathcal{B}}}} = \vec{\boldsymbol{\mathcal{B}}} \times_3 \mathbf{M}$ *which are*

$$\hat{\boldsymbol{\mathcal{A}}}(:,:,1) = \begin{bmatrix} 1 & 8 \\ 2 & -1 \end{bmatrix} \qquad \hat{\vec{\boldsymbol{\mathcal{B}}}}(:,:,1) = \begin{bmatrix} -3 \\ 5 \end{bmatrix}$$
$$\hat{\boldsymbol{\mathcal{A}}}(:,:,2) = \begin{bmatrix} 0 & 3 \\ 1 & 0 \end{bmatrix} \qquad \hat{\vec{\boldsymbol{\mathcal{B}}}}(:,:,2) = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$$

Now, we can take the facewise product of the two tensors to produce $\hat{\vec{\mathcal{C}}} = \hat{\mathcal{A}} \triangle \hat{\vec{\mathcal{B}}}$. *The frontal slices obtained by computing this facewise product are described below:*

$$\hat{\vec{\mathcal{C}}}(:,:,1) = \begin{bmatrix} 37 \\ -11 \end{bmatrix} \qquad \hat{\vec{\mathcal{C}}}(:,:,2) = \begin{bmatrix} 6 \\ -1 \end{bmatrix}$$

Finally, we take the mode-3 product of $\mathbf{M}^{-1}$ *with the tensor we computed above in order to obtain our final tensor* $\mathcal{C} = \hat{\mathcal{C}} \times_3 \mathbf{M}^{-1}$, *which is described below:*

$$\mathcal{C}(:,:,1) = \begin{bmatrix} 25 \\ -9 \end{bmatrix} \qquad \mathcal{C}(:,:,2) = \begin{bmatrix} -19 \\ 8 \end{bmatrix}$$

We also provide definitions of the $\star_{\mathrm{M}}$-transpose, the $\star_{\mathrm{M}}$-identity, and the notion of $\star_{\mathrm{M}}$-orthogonality. Note that these concepts arise directly from the $\star_{\mathrm{M}}$-product and are thus unique to this tensor framework.

Definition 2.11 ($\star_{\mathrm{M}}$-transpose). $\mathcal{A}^\top \in \mathbb{R}^{n_2 \times n_1 \times \cdots \times n_p}$ *of* $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ *is formed by transposing the frontal slices of* $\mathcal{A}$ *in the transform domain, or*

$$(\hat{\mathcal{A}}^\top)(:,:,i_3,...,i_p) = \left(\hat{\mathcal{A}}(:,:,i_3,...,i_p)\right)^\top \text{ for } i_k = 1,\dots,n_k, \text{ where } k = 3,\dots,p.$$

Definition 2.12 ($\star_{\mathrm{M}}$-identity). *The identity tensor* $\mathcal{I} \in \mathbb{R}^{n \times n \times \cdots \times n_p}$ *is the tensor such that for any tensor* $\mathcal{A}$,

$$\mathcal{A} \star_{\mathrm{M}} \mathcal{I} = \mathcal{I} \star_{\mathrm{M}} \mathcal{A} = \mathcal{A}.$$

Note that in the transform domain, we have

$$\widehat{\mathcal{I}}_{(:,:,i_3,\dots,i_p)} = \mathbf{I} \quad \text{for } i_k = 1,\dots,n_k \ , \ \text{where } k = 3,\dots,p.$$

Definition 2.13 ($\star_{\mathrm{M}}$-orthogonality). *A tensor* $\mathcal{Q} \in \mathbb{R}^{n \times n \times \cdots \times n_p}$ *is orthogonal if*

$$\mathcal{Q}^\top \star_{\mathrm{M}} \mathcal{Q} = \mathcal{Q} \star_{\mathrm{M}} \mathcal{Q}^\top = \mathcal{I}.$$

2.3. t-SVDM. From the concepts defined in Subsection 2.1 and Subsection 2.2, we can introduce the t-SVDM, a higher-dimensional analog of the SVD based upon the $\star_{\mathrm{M}}$-product. The t-SVDM and its properties are described in [10] for third-order tensors. In this section, we first provide the mathematical definition of the t-SVDM for tensors of order p and describe its computation. We then highlight the important properties offered by the t-SVDM that make it ideal for the analysis of multidimensional data, similar to the benefits offered by the traditional SVD for matrix data.

First, we define the matrix SVD, which is described in further detail in [21].

Definition 2.14 (SVD). *For a matrix* $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$, *the SVD is given by the decomposition*

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top,$$

where $\mathbf{U} \in \mathbb{R}^{n_1 \times n_1}$ *and* $\mathbf{V} \in \mathbb{R}^{n_2 \times n_2}$ *are orthogonal and* $\mathbf{\Sigma} \in \mathbb{R}^{n_1 \times n_2}$ *is a diagonal matrix with all positive entries* σ_i *for* $i = 1,...,r$. *These entries lie along the diagonal in descending order such that* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, *where* r *is the number of nonzero singular values.*

If we know that the matrix $\mathbf{A}$ is of rank r, then we can also express this decomposition as the sum of r rank-1 matrices formed from the SVD matrices. Here, $\mathbf{u}_i$ and $\mathbf{v}_i$ represent the i-th column of $\mathbf{U}$ and i-th column of $\mathbf{V}$, respectively.

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top$$

Now that we have reviewed the SVD for matrices, we can introduce the t-SVDM.

Definition 2.15 (t-SVDM). *For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$, the t-SVDM is given by the decomposition*

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} \star_{\mathrm{M}} \boldsymbol{\mathcal{S}} \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}^\top,$$

where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times n_1 \times n_3 \times \cdots \times n_p}$ and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times n_2 \times n_3 \times \cdots \times n_p}$ are $\star_{\mathrm{M}}$-orthogonal, and $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{n_1 \times n_2 \times n_3 \times \cdots \times n_p}$ is f-diagonal. We denote $\mathbf{s}_i$ as the (i,i)-tube of $\boldsymbol{\mathcal{S}}$. The tubes lie in descending magnitude such that $\|\mathbf{s}_1\|_F \geq \|\mathbf{s}_2\|_F \geq \cdots \geq \|\mathbf{s}_r\|_F > 0$ where r is the number of nonzero singular tubes.

A visualization of the t-SVDM for a third-order tensor is provided in Figure 4.

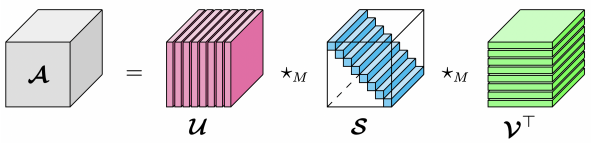


Figure 4: t-SVDM for third-order tensors

The t-SVDM also gives us a notion of rank for tensors, which we define in Definition 2.16.

Definition 2.16 (t-rank). *Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$, with its t-SVDM given by $\boldsymbol{\mathcal{U}} \star_{\mathrm{M}} \boldsymbol{\mathcal{S}} \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}^\top$. We say that the t-rank of $\boldsymbol{\mathcal{A}}$ is equal to the number of nonzero tubes in $\boldsymbol{\mathcal{S}}$, r.*

Suppose that $\boldsymbol{\mathcal{A}}$ has t-rank-r. Then, similar to the matrix SVD case, we can rewrite $\boldsymbol{\mathcal{A}}$ using the t-SVDM tensors as the sum of r t-rank-1 tensors formed from the lateral slices of $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$ and the tubes of $\boldsymbol{\mathcal{S}}$:

$$\boldsymbol{\mathcal{A}} = \sum_{i=1}^{r} \vec{\boldsymbol{\mathcal{U}}}_i \star_{\mathrm{M}} \mathbf{s}_{ii} \star_{\mathrm{M}} \vec{\boldsymbol{\mathcal{V}}}_i^\top.$$

We visualize this expansion and its relation to the t-SVDM tensors in Figure 5.

We then introduce the implementation of t-SVDM in Algorithm 2.2. Apart from the flexible choices of transformation matrices $\mathbf{M}$ as mentioned when discussing the $\star_{\mathrm{M}}$-product, the algorithm also has the advantage of allowing parallel computation as matrix SVD computations are not reliant on each other. This allows us to break the t-SVDM computation into smaller pieces.

The many useful properties of the matrix SVD are well-understood and documented [21]. We now highlight some similar beneficial properties offered by the t-SVDM.

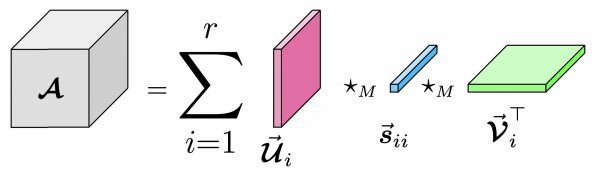


Figure 5: Expansion of t-rank-1 tensors derived from slices of $\mathcal{U}$ and $\mathcal{V}$ with diagonal tubes of $\mathcal{S}$

Algorithm 2.2 t-SVDM

Input: $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$, invertible $\mathbf{M}_3 \in \mathbb{R}^{n_3 \times n_3}, ..., \mathbf{M}_p \in \mathbb{R}^{n_p \times n_p}$
Move into transform domain: $\hat{\mathcal{A}} \leftarrow \mathcal{A}$
Concatenate frontal slices along third dimension: $\hat{\mathcal{A}} = \text{reshape}(\hat{\mathcal{A}}, [n_1, n_2, n_3 n_4 \cdots n_p])$
for $i = 1, \ldots, (n_3 n_4 \cdots n_p)$ **do**
    Compute matrix SVD: $\hat{\mathcal{A}}(:,:,i) = \hat{\mathcal{U}}(:,:,i) \cdot \hat{\mathcal{S}}(:,:,i) \cdot \hat{\mathcal{V}}(:,:,i)^\top$
end for
Reshape into p-dimensional tensors:
$\hat{\mathcal{U}} = \text{reshape}(\hat{\mathcal{U}}, [n_1, n_1, n_3..., n_p])$
$\hat{\mathcal{S}} = \text{reshape}(\hat{\mathcal{S}}, [n_1, n_2, n_3..., n_p])$
$\hat{\mathcal{V}} = \text{reshape}(\hat{\mathcal{V}}, [n_2, n_2, n_3..., n_p])$
Move back to original domain: $\mathcal{U} \leftarrow \hat{\mathcal{U}}, \mathcal{V} \leftarrow \hat{\mathcal{V}}, \mathcal{S} \leftarrow \hat{\mathcal{S}}$
Output: $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_p}$, $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times \cdots \times n_p}$, $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$

Basis: For matrices, one can take a linear combination of basis vectors to produce a new vector. In the matrix SVD, the columns of $\mathbf{U}$ form a basis for the range or column space of $\mathbf{A}$. This means that for each column j, we can find scalars $c_1, ..., c_r \in \mathbb{R}$ such that

$$\mathbf{A}(:, j) = c_1 \mathbf{U}(:, 1) + \cdots + c_r \mathbf{U}(:, r).$$

As mentioned earlier, the tensor analogs for columns and vectors are the lateral slices. The lateral slices of $\mathcal{U}$ (the $\vec{\mathcal{U}}_i$ slices) form a basis for approximating the original tensor $\mathcal{A}$. The lateral slices of $\mathcal{A}$ can be obtained by taking a tensor linear combination (or t-linear combination), in which one computes the $\star_{\text{M}}$-product of the lateral slices with tubes (the analog to scalars in the $\star_{\text{M}}$-framework) [9]. For each lateral slice j, we can find tubes $\mathbf{c}_1, ..., \mathbf{c}_r \in \mathbb{R}^{1 \times 1 \times n_3 \times \cdots \times n_p}$ such that

$$\vec{\mathcal{A}}_j = \vec{\mathcal{U}}_1 \star_{\text{M}} \mathbf{c}_1 + \cdots + \vec{\mathcal{U}}_r \star_{\text{M}} \mathbf{c}_r.$$

Eckart-Young Theorem in $\star_{\text{M}}$-framework: The Eckart-Young theorem [21] states that for any matrix $\mathbf{B}$ with rank k, we have $\|\mathbf{A} - \mathbf{A}_k\|_F \leq \|\mathbf{A} - \mathbf{B}\|_F$. An extension of the Eckart-Young theorem also exists for t-rank-k approximations obtained using the t-SVDM. Note that the following theorem simply extends work that have been provided in [10] to tensors of order-p rather than third order.

Theorem 2.17 (Eckart-Young for higher-order $\star_{\mathrm{M}}$-product). *Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ be a order-p tensor with the full t-SVDM $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} \star_{\mathrm{M}} \boldsymbol{\mathcal{S}} \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}^\top$ where the $\star_{\mathrm{M}}$-product consists of only multiples of orthogonal transformations. Define $\boldsymbol{\mathcal{A}}_k = \boldsymbol{\mathcal{U}}_k \star_{\mathrm{M}} \boldsymbol{\mathcal{S}}_k \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}_k^\top$ as the t-rank-k approximation of $\boldsymbol{\mathcal{A}}$ obtained through truncation. Then,*

$$\boldsymbol{\mathcal{A}}_k = \underset{\boldsymbol{\mathcal{B}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}}{\arg\min} \|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{B}}\|_F \quad \text{s.t.} \quad \text{t-rank}(\boldsymbol{\mathcal{B}}) = k.$$

Furthermore, the squared error is given by

$$\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_k\|_F^2 = \sum_{i=k+1}^{r} \|\mathbf{s}_i\|_F^2$$

where $\mathbf{s}_i$ is the (i,i)-tube of $\boldsymbol{\mathcal{S}}$ and r is the t-rank of $\boldsymbol{\mathcal{A}}$.

Proof. See Appendix A.1. ∎

Optimal t-rank-k Tensor Representation over rank-k Matrix Representation: Another important property is the provable optimality of a t-rank-k approximation of a tensor $\boldsymbol{\mathcal{A}}$ obtained using the t-SVDM compared to a rank-k approximation obtained from a matrix $\mathbf{A}$ containing the vectorized information of $\boldsymbol{\mathcal{A}}$, or

$$\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_k\|_F \leq \|\mathbf{A} - \mathbf{A}_k\|_F.$$

Here, $\boldsymbol{\mathcal{A}}_k$ and $\mathbf{A}_k$ represent a t-rank-k tensor and a rank-k matrix, respectively. The proof for this is provided in [10] and relies upon the Eckart-Young theorem for tensors as defined earlier. For our work, this fact illustrates that representing inherently high-dimensional data as a tensor is provably optimal to the corresponding matrix representation. In the following section, we introduce a classification approach based upon the $\boldsymbol{\mathcal{U}}$ obtained using the t-SVDM.

3. Local Tensor SVD Approaches for Classification.

Classification tasks rely on two basic assumptions: data from the same class share common features and data from distinct classes have different fundamental features. The crux of classification algorithms is the method by which one extracts meaningful features from the data. In this work, we consider supervised classification tasks, where the class labels are known a priori. In our case, the class labels are whether the subject is viewing a picture or reading a sentence. We assume that fMRI trials with either one of these class labels share some fundamental commonalities, i.e. that brains viewing a picture are responding in a distinct way from which they would respond while reading a sentence. We also assume that such differences are detectable using fMRI.

3.1. Algorithm Overview.

Our work uses a projection-based classification approach as presented in [19]. We generalize this approach to higher-order tensors and a family of tensor-tensor products. This approach begins by extracting features from our training data through building a local basis for each class. Then, we orthogonally project a test image onto the spaces spanned by each local basis, as illustrated in Figure 6 in $\mathbb{R}^3$. We make a classification decision based on the class of the local basis for which the projection produces the smallest norm difference with the original test image. The key to successful projection-based classification

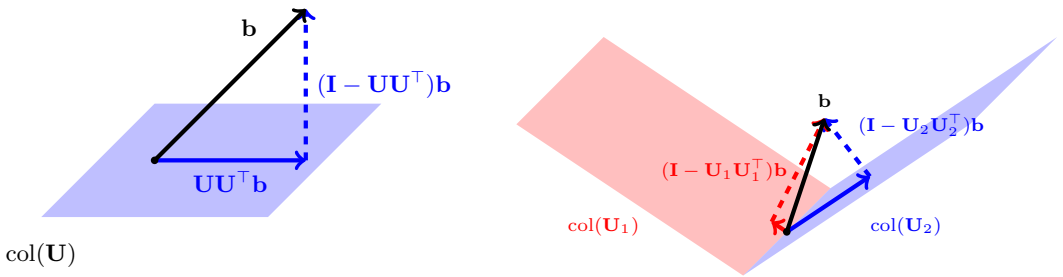


Figure 6: Illustration of a single orthogonal projection in $\mathbb{R}^3$ (left) and using orthogonal projections in $\mathbb{R}^3$ to classify (right). In the left illustration, assume $\mathbf{U}$ is a matrix with orthonormal columns. The projection (blue solid), $\mathbf{U}\mathbf{U}^\top\mathbf{b}$, lies in the column space of $\mathbf{U}$ and the error (blue dashed), $(\mathbf{I}-\mathbf{U}\mathbf{U}^\top)\mathbf{b}$, is orthogonal to the projection. In the right illustration, the vector $\mathbf{b}$ is orthogonally projected onto the column spaces of $\mathbf{U}_1$ and $\mathbf{U}_2$ respectively. The vector $\mathbf{b}$ lies more in col($\mathbf{U}_2$) than in col($\mathbf{U}_1$). Hence, $\mathbf{b}$ would be classified as belonging to the subspace spanned by $\mathbf{U}_2$.

is finding a representative basis for each class that simultaneously captures the fundamental features of the class and distinguishes between distinct classes.

Due to its provably optimal representation and the flexible choice of transformation, we propose forming local bases and projections using the t-SVDM. Specifically, let $\boldsymbol{\mathcal{A}}_i$ contain the fMRI data for the i-th class as lateral slices of $\boldsymbol{\mathcal{A}}_i$; that is, $\boldsymbol{\mathcal{A}}_i$ is a fifth-order tensor of size $(x, \text{trials}_i, y, z, \text{time})$ where x, y, and z correspond to the spatial dimensions[1]. We compute its t-SVDM

$$\boldsymbol{\mathcal{A}}_i = \boldsymbol{\mathcal{U}}_i \star_{\mathrm{M}} \boldsymbol{\mathcal{S}}_i \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}_i^\top.$$

We choose the second dimension to be the trials so that each lateral slice of $\boldsymbol{\mathcal{A}}_i$ is one fMRI image. Hence, the lateral slices of $\boldsymbol{\mathcal{U}}_i$ form an orthonormal basis which contain the most important features of the fMRI data across the trials.

To obtain features that are representative of class i, but distinguishable from other classes, we truncate the t-SVDM to k terms[2]

$$\boldsymbol{\mathcal{A}}_i \approx \boldsymbol{\mathcal{U}}_{i,k} \star_{\mathrm{M}} \boldsymbol{\mathcal{S}}_{i,k} \star_{\mathrm{M}} \boldsymbol{\mathcal{V}}_{i,k}^\top$$

where $\boldsymbol{\mathcal{U}}_{i,k}$ contains the first k lateral slices and $\boldsymbol{\mathcal{S}}_{i,k}$ and $\boldsymbol{\mathcal{V}}_{i,k}$ are truncated accordingly. The truncated t-SVDM will be the best t-rank-k approximation to $\boldsymbol{\mathcal{A}}_i$ (see Theorem 2.17) and hence the local basis $\boldsymbol{\mathcal{U}}_{i,k}$ is the best set of k lateral slices to describe class i.

Suppose we have c classes. We form class tensors $\boldsymbol{\mathcal{A}}_0, \ldots, \boldsymbol{\mathcal{A}}_{c-1}$ and local bases $\boldsymbol{\mathcal{U}}_{0,k_1}, \ldots, \boldsymbol{\mathcal{U}}_{c-1,k_{c-1}}$ where each class can have its own truncation and choice of transformation. We orthogonally project a test fMRI image (i.e., an image not contained in the class

[1] Note we can permute the dimensions of the class tensor freely as long as the second dimension corresponds to the number of images.

[2] Note that we can also select a different truncation parameter k for each class i.

tensors) stored as a lateral slice $(x, 1, y, z, \text{time})$ onto each of the class spaces via

$$\vec{\mathcal{P}}_i = \mathcal{U}_{i,k_i} \star_{\mathrm{M}} \mathcal{U}_{i,k_i}^{\top} \star_{\mathrm{M}} \vec{\mathcal{T}}.$$

The projection $\vec{\mathcal{P}}_i$ is a t-linear combination of the lateral slices contained in $\mathcal{U}_{i,k_i}$, and hence lies in the span of $\mathcal{U}_{i,k_i}$; see [9, 8] for details. Analogous to the matrix case, $\vec{\mathcal{P}}_i$ is the closest image lying in the span of $\mathcal{U}_{i,k_i}$ to the original image $\vec{\mathcal{T}}$. Here, closeness is measured in the Frobenius norm, although other norms can also be utilized.

After projecting the test image onto each of the spaces spanned by the local bases, we classify based on the projection that was closest to the original image; that is,

$$i^* = \underset{i=0,\ldots,c-1}{\arg\min} \|\vec{\mathcal{T}} - \vec{\mathcal{P}}_i\|_F.$$

Here, i^* is the predicted class.

The t-SVDM projection-based classification algorithm offers several advantages. First, the projection-based method is simple and efficient to implement. The local bases are pre-computed and can be formed in parallel because the class tensors are distinct. Second, the proposed algorithm is a direct method. We do not form a parameterized decision boundary and hence there is no training process to adjust parameters. Third, our method is flexible. We extend the original work in [19] to a more general family of tensor-tensor products based on the choice of $\mathbf{M}$. In doing so, we offer many choices of transformations that, when well-chosen, can improve the classification results. Fourth, this algorithm is based upon a rigorous tensor algebraic framework created by the $\star_{\mathrm{M}}$-product and is therefore mathematically justified. As seen in Theorem 2.17, the t-SVDM satisfies an Eckart-Young-like Theorem, hence the local bases we form are in some sense optimal. This also gives us a natural analog to projections in multidimensional space.

3.2. Intuition Example for Algorithm. This following example is intended to serve as a stepping stone towards understanding this t-SVDM classification algorithm before we proceed to describing our application to fMRI data. The MNIST database consists of 70000 28×28 grayscale images where each contains one digit between 0 and 9, resulting in 10 possible classes [13]. For illustrative purposes, we apply the local t-SVDM algorithm using only the first two classes (digits 0 and 1).

Figure 7 provides an illustration of how classification via local tensor SVD (Subsection 3.1) is accomplished using MNIST data. Here, we select a small truncation value $k = 2$ to build a basis and utilize the t-product (or Discrete Foureir Transform) as our transformation. $\mathcal{U}_{0,2}$ and $\mathcal{U}_{1,2}$ both look very similar to the original digits, which capture the features of digits. $\mathcal{U}_{0,2}$ exhibits the roundness of digit 0, and $\mathcal{U}_{1,2}$ shows the vertical characteristics of digit 1. Shift of digits is also caught in the truncated basis because of our specific transformation choice. Since $\mathcal{P}_0$ is the projection of the test image $\mathcal{T}$ onto the space spanned by the lateral slices of $\mathcal{U}_{0,2}$, we visually observe that the projection is blurred, and it seemed to have characteristics of both digit 0 and digit 1. Conversely, $\mathcal{P}_1$ is the projection of $\mathcal{T}$ to $\mathcal{U}_{1,2}$ (the true class to which it belongs), and we see that $\mathcal{P}_1$ only retains the characteristics of digit 1. Consequently,

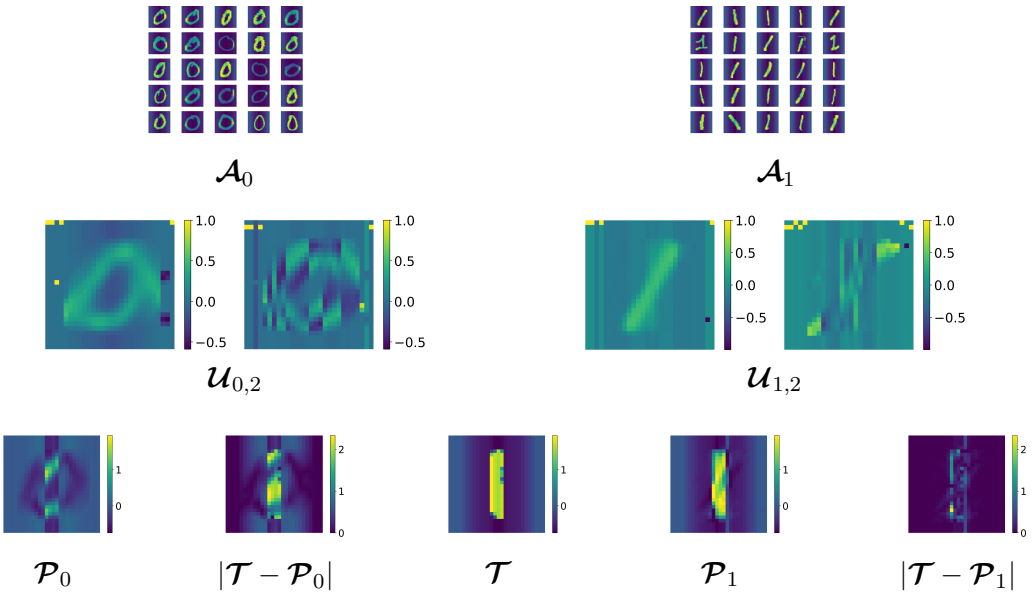


Figure 7: Illustration of applying local t-SVDM classification algorithm on MNIST database. We compute the t-SVDM of two class tensors $\mathcal{A}_0$ (representing digits consisting of 0) and $\mathcal{A}_1$ (representing digits consisting of 1), both with dimension $(\mathrm{x}, \mathrm{trials}, \mathrm{y}) = (28, 100, 28)$. Bases $\mathcal{U}_{0,2}$ and $\mathcal{U}_{1,2}$ are generated by class 0 and class 1 respectively. $\mathcal{T}$ represents the test image, which belongs to class $\mathcal{A}_1$. We project $\mathcal{T}$ onto the spaces spanned by $\mathcal{U}_{0,2}$ and $\mathcal{U}_{1,2}$ and obtain projections $\mathcal{P}_0$ and $\mathcal{P}_1$ respectively. Absolute difference images $|\mathcal{T}-\mathcal{P}_0|$ and $|\mathcal{T}-\mathcal{P}_1|$ are generated by the absolute pixel difference between $\mathcal{T}$ and $\mathcal{P}_0$, and $\mathcal{T}$ and $\mathcal{P}_1$.

the test image looks the most similar to $\mathcal{P}_1$, which also means that the underlying formation of the test image comes from the $\mathcal{U}_{1,2}$ basis, and thus can be classified as digit 1.

Numerically, using Definition 2.6, the classification procedure makes the following calculation: $\|\mathcal{T}-\mathcal{P}_0\|_F \approx 0.0263 > \|\mathcal{T}-\mathcal{P}_1\|_F \approx 0.0089$. From this, we would categorize the test image as a digit 1. One can also make similar qualitative conclusions by visually observing $|\mathcal{T}-\mathcal{P}_0|$ and $|\mathcal{T}-\mathcal{P}_1|$. The bright pixels seen in $|\mathcal{T}-\mathcal{P}_0|$ illustrate stark differences in pixel values, whereas the more consistent dark coloring in $|\mathcal{T}-\mathcal{P}_1|$ indicates that the pixels are more similar.

4. Numerical Experiments.

Our numerical experiments aim to understand how our algorithm performs for our classification task under different choices of transformations and different truncations of the basis elements. We include an overview of the choices of transformation $\mathbf{M}$ that we experiment with in Subsection 4.1 and describe our application of the algorithm and its performance on fMRI data in Subsection 4.2. Related code can be found at https://github.com/elizabethnewman/tensor-fmri.

4.1. Choices of M.

The choice of $\star_{\mathrm{M}}$-product transformations can significantly impact the extracted features of the tensor. In our experiments, we implement three invertible transformation matrices to three dimensions when computing the $\star_{\mathrm{M}}$-product in t-SVDM.

- **Discrete Fourier Transform (t-product):** Based on the work in [11], we apply the one-dimensional fast Fourier transform [4] along a mode of $\boldsymbol{\mathcal{A}}$. This specific choice of transformation can decompose signals to several separate frequencies and capture the sensitive shift of matrices in the spatial dimension.
- **Discrete Cosine Transform (c-product):** Based on the work in [8], we apply the one-dimensional discrete cosine transform [16] along a mode of $\boldsymbol{\mathcal{A}}$. This is a more efficient implementation than explicitly forming the full discrete cosine transform matrix. By transforming the signal from the spatial and temporal domain to the frequency domain, the DCT helps to separate data into parts of different importance. For this reason, the DCT is also often used in image compression domains [23][20].
- **Haar Matrix:** Introduced in [6], the Haar wavelet transformation is adept at dealing with abrupt transitions of signals [14]. Our implementation adopts the normalized version to ensure the matrix is orthogonal. One limitation of our implementation is that the dimensions of our matrix must be a power of two, i.e., $2^n \times 2^n$ for $n = 1, 2, \ldots$.
- **Banded Matrix:** We apply the lower-triangular banded matrix defined in [17]. This matrix was originally proposed for dynamic graphs, or graphs in which the nodes are fixed but edges and features can change in time. Since fMRI images are related at adjacent time points, the banded matrix could be an appropriate $\mathbf{M}$ for the temporal dimension of our data.
- **Random Orthogonal Matrix:** We construct the random orthogonal matrix by retrieving the orthogonal matrix from the QR decomposition of a $n \times n$ matrix with entries sampled from a univariate Gaussian distribution of random floats with mean 0 and variance 1. No data structure is assumed when applying the random orthogonal matrix, giving it little advantage over other transformations that do assume and incorporate some kind of structure. However, we still choose to incorporate this matrix into our experiments and compare its performance to that of other choices of $\mathbf{M}$.
- **Data Dependent Matrix:** For a transformation along the k-th dimension, the data dependent matrix $\mathbf{M}_k$ is computed via the following:

$$\boldsymbol{\mathcal{A}}_{(k)} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \text{ and } \mathbf{M}_k = \mathbf{U}^\top$$

  The advantage of this $\mathbf{M}_k$ is that it can capture structure specific to the data, despite not having the same physical features with other choices of $\mathbf{M}$. Such matrices are often used for efficient representations in transformations including Higher Order Singular Value Decomposition (HOSVD) [12].

4.2. fMRI Results. In this section, we provide our results when applying the t-SVDM classification procedure using various combinations of parameters to a classification task utilizing the StarPlus fMRI dataset. These results demonstrate the superiority of the t-SVDM method compared to the best possible equivalent matrix-based method, and also illustrate how incorporating knowledge about the data (such as time series information or annotated regions of interest) into our choice of transformation can impact performance.

4.2.1. Data Setup. For our experiments, we use the StarPlus fMRI dataset [7], which is a publicly available dataset from Carnegie Mellon University's Center for Cognitive Brain Imaging. The StarPlus fMRI dataset is organized as follows: for a single human subject, 80

trials are completed where each trial corresponds to the subject either reading a sentence or viewing a picture. Each trial is composed of a series of fMRI scans over a period of 16 time intervals spaced out over 500 milliseconds. The fMRI scan taken at each time point is three-dimensional, with 8 axial slices that are 64 by 64 pixels. For additional information about the conditions under which the trials were obtained, see [7]. The three spatial dimensions, time dimension, and trials are concatenated to form a single five-dimensional tensor, which is visualized in Figure 8.

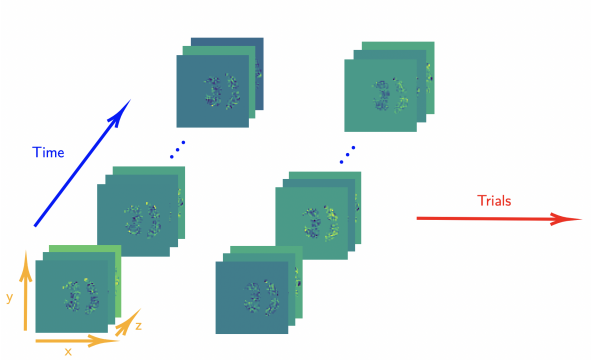


Figure 8: Visualization of data with dimensional shape (x, y, z, time, trials)

We orient the tensor such that the the trials are indexed in the second dimension, giving us the following new dimensions: $(\mathrm{x}, \mathrm{trials}, \mathrm{y}, \mathrm{z}, \mathrm{time}) = (64, 480, 64, 8, 16)$. Similar to how a sample of data is usually represented as a vector, we permute the tensor such that the second dimension contains trial information, allowing each trial to be stored as a lateral slice (analogous to vectors or columns) of the data tensor.

The comparable matrix-based approach would be to vectorize (Definition 2.3) our high-dimensional data into a matrix. This is accomplished by unraveling all of the data corresponding to a single trial into one long column. These columns are then placed side-by-side to form a two-dimensional matrix where each column contains all of the spatial and time information for a single trial. The vectorized matrix shape is $(\mathrm{x} \times \mathrm{y} \times \mathrm{z} \times \mathrm{time}, \mathrm{trials}) = (64 \times 64 \times 8 \times 16, 480) = (524288, 480)$

4.2.2. Test Accuracy Results. In this experiment, we use data from all six human subjects provided in the StarPlus dataset, resulting in a total of 480 trials. From these trials, we split the data such that 67% of the trials are used for training and the remaining 33% of the trials are used as test data. We divide the training trials based on labels so that we can construct two class tensors and compute local bases as described in Subsection 3.1. Test data is stored as a tensor, and we project each lateral slice to local bases we produced on the last step.

The matrix method would involve vectorizing the images as described earlier, computing the local SVD, and using the matrix version of projection and distance metric to make classification. We monitor the performance of our classification procedure by measuring how many

test trials it was able to correctly classify. To compute this test accuracy, we calculate

$$\text{test accuracy} = \frac{\text{number of correctly classified images}}{\text{number of images}}.$$

Figure 9 illustrates the relationship between the number of basis elements and the test accuracy for various $\mathbf{M}$. The main takeaways, as described below, are that our tensor-based method demonstrates superior performance over the traditional matrix-based approach; different transformation matrices provides different accuracies; the optimal truncation parameter also impacts results.

Tensor or Matrix. We know from Subsection 2.3 that representing high-dimensional data as a tensor is provably optimal to its vectorized matrix form. Our results quantitatively illustrate that this optimal tensor representation also carries over into classification performance. The tensor method outperforms the matrix method in terms of test accuracy in the following ways. First, with appropriate choice of transformation as will be described in the following point, the test accuracy is consistently higher than the matrix method for all choices of k. Second, we observe that tensor-based methods tend to start off well with a limited number of basis elements, despite having the same storage cost for both matrix and tensor methods. Another computational benefit can be seen in Algorithm 2.2, which shows that the t-SVDM offers potential for parallelization as well as only computing the SVD on relatively small frontal slice.

Specific choices of $\mathbf{M}$. Selected choices of $\mathbf{M}$ are discussed in detail here to demonstrate how they each influence the test accuracy.

1. **Facewise Product:** The facewise product multiplies frontal slices of the data and is not designed to account for spatial and temporal change within the fMRI images. This suggests why its test accuracy is among the lowest.
2. **Haar-Banded:** When we use the banded matrix for the temporal dimension and the Haar matrix for all other transformations, we observe a consistently higher test accuracy than using the Haar transformation for all dimensions. This makes sense since there is a time relationship in fMRI data that we can exploit. This illustrates the advantages of choosing different $\mathbf{M}$ that are optimal for the nature of information being stored in a specific dimension.
3. **Discrete Fourier Transform (t-product):** Since our fMRI data have similar slices among adjacent temporal dimensions as well as spatial dimensions, the t-product leads to the highest accuracy.

Low vs. High Representation Power. The larger the number of basis elements, the more expressive our truncated basis $\boldsymbol{\mathcal{U}}_{i,k}$ will be. However, an overly expressive basis may represent classes equally well. Likewise, keeping too few basis elements could result in our basis $\boldsymbol{\mathcal{U}}_{i,k}$ failing to properly represent the most important features for a particular class. For example, as shown in Figure 9, some choices of transformations including Haar matrix and c-product demonstrate a trend of a brief increase followed by a decrease once the number of basis elements becomes too high.

4.2.3. Utilizing Regions of Interest. The StarPlus dataset is marked with 25-30 regions of interest, or ROIs, corresponding to anatomically defined regions of the brain. Figure 10

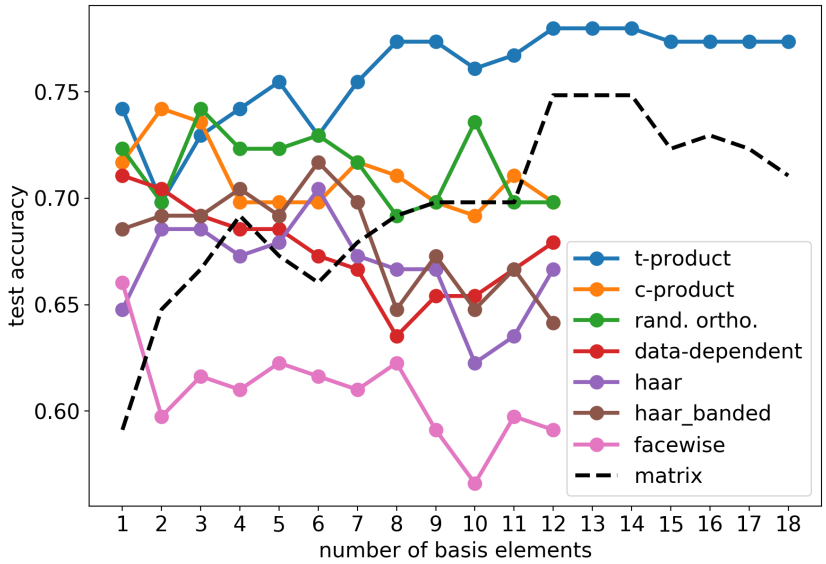


Figure 9: Test accuracy with respect to number of basis elements with varying $\mathbf{M}$. All methods are implemented for $k = 1, \ldots, 12$. We see a rise of the accuracy for the matrix method when k approaches 12, so we extend k to 18 for the matrix method. To give a complete comparison, the t-product is also run for $k = 1, \ldots, 18$. Most tensor-based methods implement the same $\mathbf{M}$ for all three transformed dimensions, except for the Haar-banded tensor method which implements banded matrix in the temporal dimension, and Haar matrix in the other two dimensions.

shows an example of how the ROIs are provided for each image, with points on the colorbar associated with specific ROIs in the brain. For more information on the meanings of these abbreviated ROIs, please see Appendix A.2.

To determine if incorporating knowledge about these anatomically-defined regions of interests impacts classification performance, we develop a transformation matrix that is formulated using the provided ROI data markups. To form this transformation, we use the following process. Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ and let $\boldsymbol{\mathcal{R}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$. Each entry of $\boldsymbol{\mathcal{R}}$ is an integer that indicates the region of interest at that voxel.

Let $m_k = n_1 n_2 \cdots n_{k-1} n_{k+1} \cdots n_p$ be the number of columns of the unfolded tensor $\boldsymbol{\mathcal{R}}_{(k)} \in \mathbb{R}^{n_k \times m_k}$. Let $1 \leq j_1 < j_2 < \cdots < j_q \leq m_k$ be the set of indices of columns of $\boldsymbol{\mathcal{R}}_{(k)}$ that contain a particular ROI label. Recall that since the columns of $\boldsymbol{\mathcal{R}}_{(k)}$ are vectorized mode-k fibers, these columns contain fibers that cross through the desired region of interest. Form the ROI selection matrix $\mathbf{P}_k^{\text{ROI}} \in \mathbb{R}^{m_k \times q}$ that selects the columns of $\boldsymbol{\mathcal{R}}_{(k)}$ that contain the desired region of interest via matrix multiplication from the right; that is, $\boldsymbol{\mathcal{R}}_{(k)} \mathbf{P}_k^{\text{ROI}} \in \mathbb{R}^{m_k \times q}$. Here, each column of $\mathbf{P}_k^{\text{ROI}}$ contains columns from an $m_k \times m_k$ identity matrix:

$$\mathbf{P}_k^{\text{ROI}}(:, \ell) = \mathbf{e}_{j_\ell} \quad \text{for} \quad \ell = 1, \ldots, q.$$

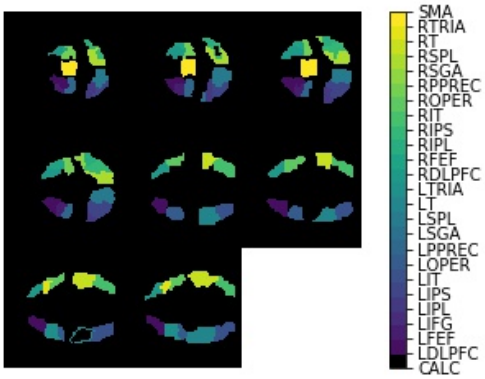
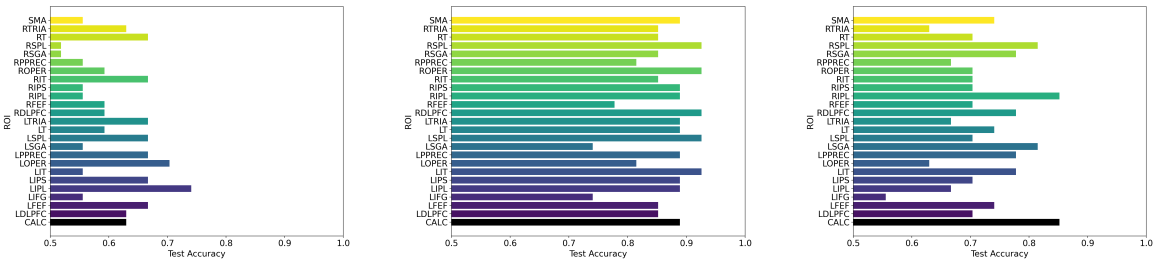


Figure 10: Regions of interest for 3D fMRI scan at single time point

To form the ROI data-dependent matrices, we take the following steps for $k = n_3, n_4, \ldots, n_p$:

$$\boldsymbol{\mathcal{A}}_{(k)}^{\text{ROI}} = \boldsymbol{\mathcal{A}}_{(k)}\mathbf{P}_k^{\text{ROI}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \qquad\qquad \mathbf{M}_k^{\text{ROI}} = \mathbf{U}^\top.$$

We use the ROI-dependent transformation to transform the data into a domain that is created from the anatomical structure according to a specific brain region. We hypothesize that the ROIs from which a better-performing transformation is created might also be regions that are known to be associated with vision or language. We repeat each of these experiments for three human subjects. Figure 11 displays results obtained when using an ROI-dependent $\mathbf{M}$.



(a) Subject 1 (b) Subject 2 (c) Subject 3

Figure 11: Results when using ROI-dependent $\mathbf{M}$ for $\mathbf{M}_3, \mathbf{M}_4$, and $\mathbf{M}_5$ with $k = 4$ for all dimensions

From these results, we can see that there do exist choices of ROI's from which we can

construct a $\mathbf{M}$, it is possible that offers improved classification accuracy over some of the choices of $\mathbf{M}$ illustrated in Figure 9. For example, for Subjects 2 and 3, there are multiple ROIs for which test accuracy exceeds 90%, an accuracy which was never reached in any of our earlier experiments. However, it is also clear that there does not exist a specific ROI that consistently improves the accuracy across all subjects. Note that we only construct the ROI-dependent $\mathbf{M}$ from the data of a single subject. Therefore, it makes sense that the performance using that $\mathbf{M}$ will vary drastically from subject to subject.

Our interpretation of these results is that the regions that are most impactful for classification must be different depending on the human subject. Since individuals might cognitively experience the tasks of reading a sentence or viewing a picture differently, making generalizations about how all humans process these kinds of information is difficult. This observation gives insight into the challenges of constructing a t-SVDM approach that is sufficiently generalized to work for various brains but specific enough to produce accurate predictions.

It is also possible that our dataset, while large in the sense that numerous trials are conducted for each subject, provides information from too few subjects. This would make it difficult to identify brain regions that would be universally impactful in the classification process. After all, six human subjects are hardly a representative sample of how all human brains work. There may actually be underlying universal similarities that could be detected by utilizing a dataset with more subjects. On the other hand, the variability introduced through the inclusion of more human subjects may make it more difficult for our multilinear approach to construct a good local basis.

Another potential culprit for the dramatic differences in accuracy among subjects could be some registration issues inherent in the StarPlus data. For example, if the MRI machine is oriented slightly differently the day that data was being collected for Subject 2 than it was for Subject 1, then our framework would not be able to fully execute the classification task to its best potential. This could perhaps be remedied by utilizing image registration techniques [25] to standardize our data.

We also recognize that our inability to construct an ROI-dependent transformation that produces consistently improved results across all subjects may simply represent an intrinsic limitation of our method. Brain activity is inherently nonlinear, produced through the discontinuous firing of neurons, and so the data that is obtained through monitoring this brain activity using fMRI would be representing nonlinear structure. Our methods in this study rely only on t-linear transformations and are not able to fully capture the nonlinear data structure.

5. Conclusions and Future Work. Based on tensor notations and $\star_{\mathrm{M}}$-product, we extend the t-SVDM framework to p-dimensional tensors and use this to describe a local truncated t-SVDM approach for image classification, which can theoretically be applied to any high-dimensional labeled datasets. In our numerical experiments, we have been able to show that there does exist a t-SVDM approach that outperforms the best equivalent matrix-based SVD approach in terms of test accuracy, which quantitatively demonstrates the advantage of using tensor methods that preserve multilinear structure. Moreover, we find drastic differences in accuracy depending on the choices of transformation matrices, which encourages intentional product selections and requires understanding of the data. We also explore the implications

of region of interests in the brain. Our success could further the development of future tensor-based approaches for classification that are better able to accommodate the complexity of high-dimensional data.

We acknowledge that while we have been able to achieve success with applying the t-SVDM to the StarPlus dataset, there are some intrinsic mathematical limitations that may have inhibited its performance. First, we only extract multilinear features via the t-SVDM. While this is an improvement over extracting linear features via the matrix SVD, fMRI data may have more complex relationships (e.g., nonlinear) that our framework cannot easily exploit. This could be remedied by combining our approach with a nonlinear classification method, such as neural networks. Second, our method is orientation-dependent: we treat the frontal slices differently than the other slices, hence the way we orient the data is crucial. However, while other tensor-based approaches are orientation-independent, they do not have a natural analogy to projections like the $\star_{\mathrm{M}}$-framework does. This motivates future methodological work of defining an orientation-independent approach in the $\star_{\mathrm{M}}$-framework (see [10, Sec. 7]) and a local bases classification approach for other tensor frameworks (e.g., Higher-Order SVD [12] and Tensor-Train [24]) that are less dependent on orientation. Third, we focus on a binary classification task, only paying attention to the differences captured between how human brains perceive picture or sentences. Our proposed classification procedure is defined in Subsection 3.1 for any number of classes, and so we are also interested in exploring how classification using the t-SVDM framework could be applied to tasks with more than two classes (e.g. more complex dataset described in [22]). Fourth, we hope to further develop these results to identify an approach that would be useful for medical diagnostic classification tasks. Using the local t-SVDM classification algorithm, we could analyze fMRI data for more significant medical challenges such as disease prediction and prevention.

Acknowledgments. This work was supported by the US National Science Foundation award DMS 2051019 and was completed during the "Computational Mathematics for Data Science" REU/RET program in Summer 2021. We would like to thank Dr. Newman and the rest of the faculty supervising this REU/RET program for their guidance and input throughout this project.

REFERENCES

[1] D. Adolf, S. Baecke, W. Kahle, J. Bernarding, and S. Kropf, *Applying multivariate techniques to high-dimensional temporally correlated fmri data*, Journal of Statistical Planning and Inference, 141 (2011), pp. 3760–3770, https://doi.org/https://doi.org/10.1016/j.jspi.2011.06.012, https://www.sciencedirect.com/science/article/pii/S0378375811002370.

[2] C. Chatzichristos, E. Kofidis, M. Morante, and S. Theodoridis, *Blind fmri source unmixing via higher-order tensor decompositions*, Journal of Neuroscience Methods, 315 (2019), pp. 17–47, https://doi.org/https://doi.org/10.1016/j.jneumeth.2018.12.007, https://www.sciencedirect.com/science/article/pii/S0165027018303996.

[3] Cleveland Clinic, *Brain: Temporal lobe, vagal nerve, & frontal lobe*, https://my.clevelandclinic.org/health/diseases/16799-brain-temporal-lobe-vagal-nerve--frontal-lobe.

[4] J. W. Cooley and J. W. Tukey, *An algorithm for the machine calculation of complex fourier series*, Mathematics of Computation, 19 (1965), pp. 297–301, http://www.jstor.org/stable/2003354.

[5] G. H. Glover, *Overview of functional magnetic resonance imaging*, (2012), https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3073717/.

[6] A. Haar, *Zur theorie der orthogonalen funktionensysteme*, Mathematische Annalen, 69 (1910), pp. 331–371, https://doi.org/10.1007/BF01456326, https://doi.org/10.1007/BF01456326.

[7] M. Just, *Starplus fmri data*. http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-81/www/. Center for Cognitive Brain Imaging at Carnegie Mellon University.

[8] E. Kernfeld, M. Kilmer, and S. Aeron, *Tensor–tensor products with invertible linear transforms*, Linear Algebra and its Applications, 485 (2015), pp. 545–570, https://doi.org/10.1016/j.laa.2015.07.021.

[9] M. Kilmer, K. Braman, N. Hao, and R. Hoover, *Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging*, SIAM Journal on Matrix Analysis and Applications, 34 (2013), https://doi.org/10.1137/110837711.

[10] M. E. Kilmer, L. Horesh, H. Avron, and E. Newman, *Tensor-tensor algebra for optimal representation and compression of multiway data*, Proceedings of the National Academy of Sciences, 118 (2021), https://doi.org/10.1073/pnas.2015851118, https://www.pnas.org/content/118/28/e2015851118, https://arxiv.org/abs/https://www.pnas.org/content/118/28/e2015851118.full.pdf.

[11] M. E. Kilmer and C. D. Martin, *Factorization strategies for third-order tensors*, Linear Algebra and its Applications, 435 (2011), pp. 641–658, https://doi.org/https://doi.org/10.1016/j.laa.2010.09.020, https://www.sciencedirect.com/science/article/pii/S0024379510004830. Special Issue: Dedication to Pete Stewart on the occasion of his 70th birthday.

[12] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Review, 51 (2009), pp. 455–500, https://doi.org/10.1137/07070111X.

[13] Y. LeCun and C. Cortes, *MNIST handwritten digit database*, (2010), http://yann.lecun.com/exdb/mnist/.

[14] B. Y. Lee and Y. S, *Application of the discrete wavelet transform to the monitoring of tool failure in end milling using the spindle motor current*, The International Journal of Advanced Manufacturing Technology, 15 (1999), pp. 238–243, https://doi.org/10.1007/s001700050062, https://doi.org/10.1007/s001700050062.

[15] G. Ma, L. He, C.-T. Lu, P. S. Yu, L. Shen, and A. B. Ragin, *Spatio-Temporal Tensor Analysis for Whole-Brain fMRI Classification*, pp. 819–827, https://doi.org/10.1137/1.9781611974348.92, https://epubs.siam.org/doi/abs/10.1137/1.9781611974348.92, https://arxiv.org/abs/https://epubs.siam.org/doi/pdf/10.1137/1.9781611974348.92.

[16] J. Makhoul, *A fast cosine transform in one and two dimensions*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 28 (1980), pp. 27–34, https://doi.org/10.1109/TASSP.1980.1163351.

[17] O. A. Malik, S. Ubaru, L. Horesh, M. E. Kilmer, and H. Avron, *Tensor graph convolutional networks for prediction on dynamic graphs*, 2020, https://openreview.net/forum?id=rylVTTVtvH.

[18] E. Newman, M. Kilmer, and L. Horesh, *Image classification using local tensor singular value decompositions*, in 2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2017, pp. 1–5, https://doi.org/10.1109/CAMSAP.2017.8313137.

[19] E. Newman, M. Kilmer, and L. Horesh, *Image classification using local tensor singular value decompositions*, in 2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2017, pp. 1–5, https://doi.org/10.1109/CAMSAP.2017.8313137.

[20] A. M. Raid, W. M. Khedr, M. A. El-Dosuky, and W. Ahmed, *Jpeg image compression using discrete cosine transform - A survey*, CoRR, abs/1405.6147 (2014), http://arxiv.org/abs/1405.6147, https://arxiv.org/abs/1405.6147.

[21] G. Strang, *Linear Algebra and Learning from Data*, Wellesley-Cambridge Press, 2019, https://books.google.com/books?id=L0Y_wQEACAAJ.

[22] D. C. Van Essen, K. Ugurbil, E. Auerbach, D. Barch, T. E. Behrens, R. Bucholz, A. Chang, L. Chen, M. Corbetta, S. W. Curtiss, S. Della Penna, D. Feinberg, M. F. Glasser, N. Harel, A. C. Heath, L. Larson-Prior, D. Marcus, G. Michalareas, S. Moeller, R. Oostenveld, S. E. Petersen, F. Prior, B. L. Schlaggar, S. M. Smith, A. Z. Snyder, J. Xu, and E. Yacoub, *The human connectome project: a data acquisition perspective*, Neuroimage, 62 (2012), pp. 2222–31, https://doi.org/10.1016/j.neuroimage.2012.02.018.

[23] A. Watson, *Image compression using the discrete cosine transform*, Mathematica Journal, 4 (1994), https://doi.org/10.1007/978-3-322-96658-2_5.

[24] X. Xu, Q. Wu, S. Wang, J. Liu, J. Sun, and A. Cichocki, *Whole brain fmri pattern analysis based on*

tensor neural network, IEEE Access, 6 (2018), pp. 29297–29305, https://doi.org/10.1109/ACCESS.2018.2815770.

[25] X. Zhang, Y. Feng, W. Chen, X. Li, A. V. Faria, Q. Feng, and S. Mori, *Linear registration of brain mri using knowledge-based multiple intermediator libraries*, Frontiers in Neuroscience, 13 (2019), p. 909, https://doi.org/10.3389/fnins.2019.00909, https://www.frontiersin.org/article/10.3389/fnins.2019.00909.

Appendix A. Appendix.

A.1. Proofs for the Eckart-Young Theorem for Order-p Tensors.

Lemma A.1 (Tensor Orthogonal Invariance under $\star_{\mathrm{M}}$-product). *Let $\mathbf{M}_3 \in \mathbb{R}^{n_3 \times n_3}, ..., \mathbf{M}_p \in \mathbb{R}^{n_p \times n_p}$ such that each $\mathbf{M}_i$ is an invertible non-zero scalar multiple of an orthogonal matrix $\mathbf{W}_i$ with scalars $c_i \in \mathbb{R}$ and $\star_{\mathrm{M}}$-orthogonal $\mathcal{Q} \in \mathbb{R}^{n \times n \times \cdots \times n_p}$. Then, for $\mathcal{A} \in \mathbb{R}^{n \times \ell \times \cdots \times n_p}$, we have $\|\mathcal{Q} \star_{\mathrm{M}} \mathcal{A}\|_F = c\|\mathcal{A}\|_F$, with $c \in \mathbb{R}$. Likewise, if $\mathcal{A} \in \mathbb{R}^{\ell \times n \times \cdots \times n_p}$, $\|\mathcal{A} \star_{\mathrm{M}} \mathcal{Q}\|_F = c\|\mathcal{A}\|_F$.*

Proof. Let $\mathcal{A} \in \mathbb{R}^{n \times \ell \times \cdots \times n_p}$. First, we show that the norm of $\mathcal{A}$ is preserved (up to scalar multiplication) in the transform domain. From [10], we know that $\|\mathcal{A} \times_3 \mathbf{M}_3\|_F = c_3\|\mathcal{A}\|_F$. This generalizes to the mode-k product as follows. Assume $\mathbf{M}_k = c_k \mathbf{W}_k$ where $\mathbf{W}_k \in \mathbb{R}^{n_k \times n_k}$ is orthogonal. Then,

$$\|\mathcal{A} \times_k \mathbf{M}_k\|_F = \|\mathbf{M}_k \mathcal{A}_{(k)}\|_F = \|c_k \mathbf{W}_k \mathcal{A}_{(k)}\|_F = c_k\|\mathcal{A}_{(k)}\|_F = c_k\|\mathcal{A}\|_F.$$

If we apply multiple transformation matrices to each dimension from 3 to p, we obtain the following:

$$\begin{aligned}
\|\hat{\mathcal{A}}\|_F &= \|\mathcal{A} \times_3 \mathbf{M}_3 \cdots \times_p \mathbf{M}_p\|_F \\
&= \|\mathbf{M}_p(\mathcal{A} \times_3 \mathbf{M}_3 \cdots \times_{p-1} \mathbf{M}_{p-1})_{(p)}\|_F \\
&= \|c_p \mathbf{W}_p(\mathcal{A} \times_3 \mathbf{M}_3 \cdots \times_{p-1} \mathbf{M}_{p-1})_{(p)}\|_F \\
&= c_p\|\mathcal{A} \times_3 \mathbf{M}_3 \cdots \times_{p-1} \mathbf{M}_{p-1}\|_F \\
&\vdots \\
&= c\|\mathcal{A}\|_F \quad \text{where } c = c_3 c_4 \cdots c_p.
\end{aligned}$$

We now show the norm-invariance of the $\star_{\mathrm{M}}$-product when multiplying by an orthogonal tensor. Let $\mathcal{Q}$ be orthogonal and $\mathcal{C} = \mathcal{Q} \star_{\mathrm{M}} \mathcal{A}$. Then,

$$\begin{aligned}
\|\mathcal{A}\|_F^2 &= \frac{1}{c^2}\|\hat{\mathcal{A}}\|_F^2 \\
&= \frac{1}{c^2} \sum_{i_3=1}^{n_3} \cdots \sum_{i_p=1}^{n_p} \|\mathcal{A}_{:,:,i_3,\ldots,i_p}\|_F^2 \\
&= \frac{1}{c^2} \sum_{i_3=1}^{n_3} \cdots \sum_{i_p=1}^{n_p} \|\hat{\mathcal{Q}}_{:,:,i_3,\ldots,i_p} \hat{\mathcal{A}}_{:,:,i_3,\ldots,i_p}\|_F^2 \text{ since frontal slices of } \hat{\mathcal{Q}} \text{ are orthogonal} \\
&= \frac{1}{c^2}\|\hat{\mathcal{C}}\|_F \\
&= \|\mathcal{C}\|_F.
\end{aligned}$$

The other direction is similar. ∎

Lemma A.2 (Ordering of Singular Tubes). *Given the t-SVDM of $\boldsymbol{\mathcal{A}}$ where $\boldsymbol{\mathcal{A}}$ is of t-rank-r, we have $\|\boldsymbol{\mathcal{A}}\|_F^2 = \|\boldsymbol{\mathcal{S}}\|_F^2 = \sum_{i=1}^r \|\mathbf{s}_i\|_F^2$ where $\mathbf{s}_i$ is the (i,i)-tube of $\mathbf{S}$. Moreover, $\|\mathbf{s}_1\|_F \geq \|\mathbf{s}_2\|_F \geq \cdots \geq \|\mathbf{s}_r\|_F > 0$.*

Proof. We know that $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}^\top$ are $\star_{\mathrm{M}}$-orthogonal and the only entries of $\boldsymbol{\mathcal{S}}$ lie along the diagonal of its frontal slices. By results shown in [10], we have

$$\|\boldsymbol{\mathcal{A}}\|_F^2 = \|\boldsymbol{\mathcal{U}}\boldsymbol{\mathcal{S}}\boldsymbol{\mathcal{V}}^\top\|_F^2 = \|\boldsymbol{\mathcal{S}}\|_F^2 = \sum_{i=1}^r \|\mathbf{s}_i\|_F^2.$$

We now prove the second part. From Lemma A.1, we have

$$\|\mathbf{s}_i\|_F^2 = \frac{1}{c^2}\|\hat{\mathbf{s}}_i\|_F^2 = \frac{1}{c^2}\sum_{i_3=1}^{n_3}\cdots\sum_{i_p=1}^{n_p}\hat{\sigma}_{i,i,i_3,\ldots,i_p}^2$$

where $\hat{\sigma}_i^{(i_3,\ldots,i_p)}$ is the i-th largest singular value in the transform domain located at the frontal slice along fixed indices $i_3,\ldots,i_p$. By the definition of the matrix SVD, we know that $\hat{\sigma}_i^{(i_3,\ldots,i_p)} \geq \hat{\sigma}_{i+1}^{(i_3,\ldots,i_p)}$, or that the singular values lie along the diagonal of the singular value matrix in descending order of magnitude. Thus, we have

$$\|\mathbf{s}_i\|_F \geq \|\mathbf{s}_{i+1}\|_F.$$

Hence, the norm of the singular tubes is ordered. ∎

The proof of the Eckart-Young-like Theorem 2.17 is provided below.

Proof. We first obtain a formulation for the error. We know that for the matrix SVD, we have $\|\mathbf{A}-\mathbf{A}_k\|_F^2 = \sum_{i=k+1}^r \sigma_i^2$ [21]. Thus, we have

$$\begin{aligned}
\|\boldsymbol{\mathcal{A}}-\boldsymbol{\mathcal{A}}_k\|_F^2 &= \frac{1}{c^2}\|\hat{\boldsymbol{\mathcal{A}}}-\hat{\boldsymbol{\mathcal{A}}}_k\|_F^2 \\
&= \frac{1}{c^2}\sum_{i_3=1}^{n_3}\cdots\sum_{i_p=1}^{n_p}\|\hat{\boldsymbol{\mathcal{A}}}_{:,:,i_3,\ldots,i_p}-\hat{\boldsymbol{\mathcal{A}}}_{:,1:k,i_3,\ldots,i_p}\|_F^2 \\
&= \frac{1}{c^2}\sum_{i_3=1}^{n_3}\cdots\sum_{i_p=1}^{n_p}\sum_{i=k+1}^{r}\hat{\sigma}_{i,i,i_3,\ldots,i_p}^2 \quad \text{(Lemma A.2)} \\
&= \frac{1}{c^2}\sum_{i=k+1}^{r}\|\mathbf{s}_i\|_F^2.
\end{aligned}$$

Let $\boldsymbol{\mathcal{B}}$ be a tensor of t-rank-k. Then, using the fact that the SVD produces the best possible rank-k approximation of a matrix, we have

$$\begin{aligned}
\|\boldsymbol{\mathcal{A}}-\boldsymbol{\mathcal{B}}\|_F^2 &= \frac{1}{c^2}\|\hat{\boldsymbol{\mathcal{A}}}-\hat{\boldsymbol{\mathcal{B}}}\|_F^2 \quad \text{(Lemma A.1)} \\
&= \frac{1}{c^2}\|\operatorname{reshape}(\hat{\boldsymbol{\mathcal{A}}},[n_1,n_2,n_3n_4\cdots n_p]) - \operatorname{reshape}(\hat{\boldsymbol{\mathcal{B}}},[n_1,n_2,n_3n_4\cdots n_p])\|_F^2 \\
&\geq \frac{1}{c^2}\|\operatorname{reshape}(\hat{\boldsymbol{\mathcal{A}}},[n_1,n_2,n_3n_4\cdots n_p]) - \operatorname{reshape}(\hat{\boldsymbol{\mathcal{A}}}_k,[n_1,n_2,n_3n_4\cdots n_p])\|_F^2 \\
&= \|\boldsymbol{\mathcal{A}}-\boldsymbol{\mathcal{A}}_k\|_F^2.
\end{aligned}$$

∎

When we compute reshape($\hat{\boldsymbol{\mathcal{B}}}, [n_1, n_2, n_3 n_4 \cdots n_p]$), we are concatenating the decoupled frontal slices in the transform domain into a third-order tensor. Thus, we can use the results from [10] and immediately obtain the inequality above.

A.2. Additional Descriptions for Regions of Interest. To clarify some of the abbreviated ROIs referenced in Subsection 4.2.3, we provide the full names of each of the ROIs in the below table.

Abbreviation	Name
LTRIA/RTRIA	Left/Right Triangularis
LIT/RIT	Left/Right Inferior Temporal Lobe
LIPS/RIPS	Left/Right Intraparietal Sulcus
LSGA/RSGA	Left/Right Supramarginal Gyrus
LT/RT	Left/Right Temporal Lobe
LOPER/ROBER	Left/Right Opercularis
LSPL/RSPL	Left Superior Parietal Lobe
LPPREC/RPPREC	Left/Right Posterior Precentral Sulcus
LDLPFC/RDLPFC	Left/Right Dorsolateral Prefrontal Cortex
CALC	Calcarine Sulcus

The above terms, along with more detailed explanations of each of these regions, can be found in [3]. Our initial hypothesis was that regions known to be associated with vision and language (e.g. the left temporal lobe, which is associated with language skills) might have a greater role in improving accuracy when utilized as an ROI-dependent $\mathbf{M}$ as described in Subsection 4.2.3.