Biometrika (2024), **111**, 2, *pp.* 707–714 https://doi.org/10.1093/biomet/asad049
Advance Access publication 31 August 2023

Deep Kronecker network

By LONG FENG

Department of Statistics and Actuarial Science, The University of Hong Kong, Pokfulam Road, Hong Kong
lfeng@hku.hk

and GUANG YANG

School of Data Science, City University of Hong Kong, 83 Tat Chee Ave, Kowloon Tong, Hong Kong
guang.yang@my.cityu.edu.hk

Summary

We develop a novel framework for the analysis of medical imaging data, including magnetic resonance imaging, functional magnetic resonance imaging, computed tomography and more. Medical imaging data differ from general images in two main aspects: (i) the sample size is often considerably smaller and (ii) the interpretation of the model is usually more crucial than predicting the outcome. As a result, standard methods such as convolutional neural networks cannot be directly applied to medical imaging analysis. Therefore, we propose the *deep Kronecker network*, which can adapt to the low sample size constraint and offer the desired model interpretation. Our approach is versatile, as it works for both matrix- and tensor-represented image data and can be applied to discrete and continuous outcomes. The deep Kronecker network is built upon a Kronecker product structure, which implicitly enforces a piecewise smooth property on coefficients. Moreover, our approach resembles a fully convolutional network as the Kronecker structure can be expressed in a convolutional form. Interestingly, our approach also has strong connections to the tensor regression framework proposed by Zhou et al. (2013), which imposes a canonical low-rank structure on tensor coefficients. We conduct both classification and regression analyses using real magnetic resonance imaging data from the Alzheimer's Disease Neuroimaging Initiative to demonstrate the effectiveness of our approach.

Some key words: Brain imaging; Convolutional neural network; Kronecker product; Tensor decomposition.

1. Introduction

Medical imaging analysis plays a central role in modern medicine. With the progression of imaging technologies, such as computed tomography, magnetic resonance imaging, MRI and functional magnetic resonance imaging, fMRI, the diagnosis and treatment of diseases have experienced significant improvements.

Although image analysis has been intensively studied over the past years, medical image data differs from general images in two main aspects. First, medical imaging typically has a much smaller sample size, but with higher order and higher dimension. For example, datasets in MRI analysis frequently consist of only a few hundred or at most a few thousand patients, each with an MRI scan that contains millions of voxels. In contrast, sample sizes in general image recognition problems can easily reach millions, surpassing the image dimensions significantly. Second, while predicting

the outcome is a top priority in many image recognition problems, medical imaging analysis places greater importance on interpreting the model.

Because of the unique nature of medical imaging data, it is challenging to directly apply general image methods. In recent years, convolutional neural networks (Fukushima & Miyake, 1982; LeCun et al., 1998) have emerged as the most successful method for image recognition. However, their training requires large amounts of samples, which are rarely available in medical imaging analysis. Moreover, a convolutional neural network typically consists of thousands of unknown parameters within a black box, rendering it difficult to interpret and unable to meet the needs of medical imaging analysis.

In the statistics community, numerous efforts have been made to develop methodologies for medical imaging analysis. One common strategy is to vectorize the images and use the resulting vectors as independent predictors. Built on this strategy, various methods have been developed in the literature, including total variation and fused lasso-based approaches (Rudin et al., 1992; Tibshirani et al., 2005; Wang et al., 2017), Bayesian methods (Goldsmith et al., 2014; Kang et al., 2018) and more. Despite their effectiveness in different applications, vectorizing images is clearly not an optimal strategy. Apart from the loss of spatial information, the resulting ultra-high-dimensional vectors could also face significant computational issues. When image data are represented as tensors, Zhou et al. (2013) proposed a tensor regression framework that imposes a canonical low-rank structure on the tensor coefficients, thereby significantly reducing the number of unknown parameters. Furthermore, Feng et al. (2021) proposed a new internal variation penalization to mimic the effects of total variation and promote smoothness in tensor image regression. While the tensor regression framework is appealing for general tensors, it does not fully capitalize the special nature of image data. Recently, Wu & Feng (2023) proposed an innovative framework named *sparse Kronecker product decomposition* for identifying signal regions in image regression. As this approach is designed for sparse signal detection, it is not well suited for analysing images with dense signals.

To this end, we aim to develop an approach for medical imaging analysis that can (i) adapt to low sample size limitation, (ii) enjoy good interpretability and (iii) achieve desired prediction power. In this paper, we develop a novel framework called the deep Kronecker network that is able to achieve all three goals. The deep Kronecker network is built upon a Kronecker product structure, which inherently imposes a piecewise smooth property on coefficients. Moreover, the deep Kronecker network allows us to locate the most influential regions for the outcome, facilitating model interpretation. Our approach works for both matrix- and high-order tensor-represented image data, and thus MRI and fMRI can be addressed. In addition, the deep Kronecker network is embedded in a generalized linear model; therefore, it is applicable to both discrete and continuous outcomes.

We refer to the deep Kronecker network as a network because it resembles a convolutional neural network, particularly, a fully convolutional network. While the deep Kronecker network originates from a Kronecker structure, it can also be represented in a convolutional form. Unlike a classical convolutional neural network, the convolutions in the deep Kronecker network do not overlap. This design allows us to achieve maximized dimension reduction and at the same time enjoy desired model interpretability. Interestingly, the deep Kronecker network is also connected to the tensor regression framework of Zhou et al. (2013). We show that the deep Kronecker network, not only includes tensor regression as a special case, but can also be easily implemented by applying tensor regression on certain reshaped images. Consequently, three seemingly irrelevant methods, deep Kronecker network, convolutional neural network and tensor regression, are connected. Finally, we conduct a real MRI analysis from the Alzheimer's Disease Neuroimaging Initiative to further demonstrate the effectiveness of our approach.

2. Deep Kronecker network

Suppose that we observe n samples, each consisting of a tensor-represented image $\mathcal{X}_i \in \mathbb{R}^{d\times p\times q}$ and a scalar response y_i for $i = 1, 2, \ldots, n$. Assume that y_i follows a generalized linear model:

$$y_i \mid \mathcal{X}_i \sim \mathbb{P}(y_i \mid \mathcal{X}_i) = \rho(y_i)\exp\{y_i\langle\mathcal{X}_i, \mathcal{C}\rangle - \psi(\langle\mathcal{X}_i, \mathcal{C}\rangle)\} \tag{1}$$

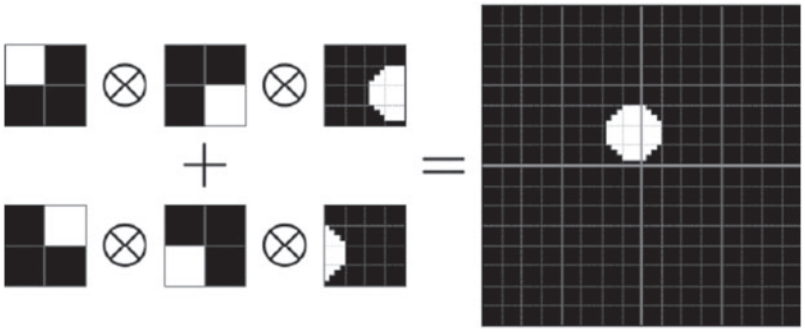

Fig. 1. An illustration of (2) with $L = 3$, $R = 2$, $\mathcal{B}_3^r, \mathcal{B}_2^r \in \mathbb{R}^{2\times 2}$, $\mathcal{B}_1^r \in \mathbb{R}^{4\times 4}$, $r = 1, 2$.

with $\mathcal{C} \in \mathbb{R}^{d\times p\times q}$ the target unknown coefficient tensor, $\langle\cdot,\cdot\rangle$ the inner product and $\rho(\cdot)$ and $\psi(\cdot)$ certain known univariate functions. Here we focus on the image data and omit other possible design variables, such as age, sex, etc. These variables can be added back into the model easily if needed. Given model (1), we have, for a certain known link function $g(\cdot)$,

$$g\{\mathbb{E}(y_i)\} = \langle \mathcal{X}_i, \mathcal{C}\rangle.$$

Now we introduce the Kronecker product for K-order tensors.

Definition 1 (Tensor Kronecker product). *Let $\mathcal{A} \in \mathbb{R}^{p_1\times\cdots\times p_K}$ and $\mathcal{B} \in \mathbb{R}^{q_1\times\cdots\times q_K}$ be two K-order tensors with entries denoted by $\mathcal{A}_{i_1,\ldots,i_K}$ and $\mathcal{B}_{j_1,\ldots,j_K}$, respectively. The tensor Kronecker product $\mathcal{C} = \mathcal{A}\otimes\mathcal{B}$ is defined by $\mathcal{C}_{[j_1i_1],\ldots,[j_Ki_K]} = \mathcal{A}_{i_1,\ldots,i_K}\mathcal{B}_{j_1,\ldots,j_K}$ for all possible values of $(i_1,\ldots,i_K)$ and $(j_1,\ldots,j_K)$, where $[j_ki_k] = j_k + (i_k-1)q_k$ for all $k = 1, 2, \ldots, K$.*

Our deep Kronecker network models the tensor coefficient $\mathcal{C}$ using a rank-R Kronecker product decomposition with $L(\geqslant 2)$ factors:

$$\mathcal{C} = \sum_{r=1}^{R} \mathcal{B}_L^r \otimes \mathcal{B}_{L-1}^r \otimes \cdots \otimes \mathcal{B}_1^r. \tag{2}$$

Here $\mathcal{B}_l^r \in \mathbb{R}^{d_l\times p_l\times q_l}$, $l \in [L], r \in [R]$, are unknown tensors and referred to as Kronecker factors. The sizes of $\mathcal{B}_l^r$ are unknown, but are assumed to satisfy $d = \prod_{l=1}^L d_l$, $p = \prod_{l=1}^L p_l$ and $q = \prod_{l=1}^L q_l$. For ease of notation, we also write (2) in the form $\mathcal{C} = \sum_{r=1}^R \bigotimes_{l=L}^1 \mathcal{B}_l^r$.

Figure 1 illustrates a Kronecker product decomposition with rank $R = 2$ and factor number $L = 3$ for a sparse matrix where the signal takes the form of a circle. In general, the Kronecker product decomposition (2) is able to approximate arbitrary matrices with a sufficiently large rank R. This can be seen by connecting (2) to the canonical tensor decomposition; see § 4.

The deep Kronecker network is designed for analysing medical images with low sample sizes and high dimensions. For three-order tensors, the deep Kronecker network could reduce the parameter number from $\prod_{l=1}^L d_lp_lq_l$ to $R\sum_{l=1}^L d_lp_lq_l$. Such a dimension reduction is particularly crucial in medical imaging analysis, where sample sizes are often limited.

The deep Kronecker network could be solved using maximum likelihood estimation. Under models (1) and (2), the negative loglikelihood function with regard to factors $[\mathcal{B}_1^1, \ldots, \mathcal{B}_L^R]$ is proportional to

$$\ell(\mathcal{B}_1^1, \ldots, \mathcal{B}_L^R) = \sum_{i=1}^{n} \left\{ \psi\left(\left\langle \mathcal{X}_i, \sum_{r=1}^{R} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\rangle\right) - y_i \left\langle \mathcal{X}_i, \sum_{r=1}^{R} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\rangle \right\}. \tag{3}$$

When the outcome y_i is Gaussian, the maximum likelihood estimation reduces to ordinary least squares. To minimize (3), we apply an alternating minimization algorithm to iteratively update the

blocked factors $[\mathcal{B}_l^1, \mathcal{B}_l^2, \ldots, \mathcal{B}_l^R]$, with $[\mathcal{B}_{l'}^1, \mathcal{B}_{l'}^2, \ldots, \mathcal{B}_{l'}^R]$, $l' \neq l$, fixed. We defer the computation details to the Supplementary Material.

In the literature, Kronecker product decomposition has become a powerful tool for matrix approximation and dimension reduction. In particular, Kronecker product singular value decomposition is referred to as the problem of recovering B_l^r from a given matrix $C = \sum_{r=1}^R \bigotimes_{l=L}^1 B_l^r$. Such a problem was mostly studied under $L = 2$; see, e.g., Cai et al. (2020). It becomes much more challenging when $L \geqslant 3$ (Hackbusch et al., 2005). More recently, Batselier & Wong (2017) considered Kronecker product singular value decomposition with $L \geqslant 3$ and proposed an algorithm to convert it to the canonical tensor decomposition. Kronecker product decomposition has also been studied in other contexts, e.g., correlation matrix estimation (Hafner et al., 2020), matrix autoregressive model (Chen et al., 2020) and sparse signal detection (Wu & Feng, 2023).

3. Deep Kronecker network in convolutional form and the nonlinear deep Kronecker network

We refer to the deep Kronecker network as a network because it resembles a convolutional neural network. To illustrate the connections between the deep Kronecker and convolutional neural network, we first introduce a non-overlapping convolution operator. Given two tensors $\mathcal{X} \in \mathbb{R}^{d_0 \times p_0 \times q_0}$ and $\mathcal{B} \in \mathbb{R}^{d' \times p' \times q'}$, we define the non-overlapping convolution between $\mathcal{X}$ and $\mathcal{B}$ as

$$\mathcal{X} * \mathcal{B} \in \mathbb{R}^{d'' \times p'' \times q''}, \qquad d'' = d_0/d',\ p'' = p_0/p',\ q'' = q_0/q',$$

where the (h, j, k)th component is

$$(\mathcal{X} * \mathcal{B})_{h,j,k} = \langle \mathcal{X}_{h,j,k}^{d' \times p' \times q'}, \mathcal{B} \rangle, \qquad 1 \leqslant h \leqslant d'',\ 1 \leqslant j \leqslant p'',\ 1 \leqslant k \leqslant q''.$$

Here $\mathcal{X}_{h,j,k}^{d' \times p' \times q'}$ is the (h, j, k)th block of $\mathcal{X}$ and is of size $d' \times p' \times q'$. Building on this convolution operator, the following theorem establishes the connections between the deep Kronecker network and convolutional neural network.

Theorem 1. *The deep Kronecker network can be written in convolutional form:*

$$g\{\mathbb{E}(y_i)\} = \left\langle \mathcal{X}_i, \sum_{r=1}^R \bigotimes_{l=L}^1 \mathcal{B}_l^r \right\rangle \quad \Longleftrightarrow \quad g\{\mathbb{E}(y_i)\} = \sum_{r=1}^R \mathcal{X}_i * \mathcal{B}_1^r * \mathcal{B}_2^r * \cdots * \mathcal{B}_{L-1}^r * \mathcal{B}_L^r.$$

Theorem 1 suggests that response y_i could also be modelled by a summation of consecutive convolutions between image $\mathcal{X}_i$ and factors $\mathcal{B}_l^r$. In other words, the deep Kronecker network can be considered as a network consisting solely of convolution layers, that is, a fully convolutional network. More specifically, we may regard L as the depth of a deep Kronecker network, R as the width and $\mathcal{B}_l^r$ as the convolution filters. But here the convolutions do not overlap with each other, meaning that the stride sizes are equal to the filter sizes. On the one hand, the non-overlapping design allows the deep Kronecker network to achieve maximized dimension reduction, thus eliminating the need for pooling layers. On the other hand, it enables the explicit formulation of the coefficient tensor, allowing us to locate the significant regions and achieve desired model interpretability. Both aspects are crucial in medical imaging analysis. Figure 2 illustrates the deep Kronecker network in a convolutional form.

The activation function in the deep Kronecker network is taken as an identity function. By introducing a nonlinear function $h(\cdot)$, we can generalize the deep Kronecker network to its nonlinear version

$$g\{\mathbb{E}(y_i)\} = \sum_{r=1}^R h[\cdots h\{h(\mathcal{X}_i * \mathcal{B}_1^r) * \mathcal{B}_2^r\} \cdots * \mathcal{B}_{L-1}^r] * \mathcal{B}_L^r.$$

Standard activation functions, such as the rectified linear unit or sigmoid, can be used for the deep Kronecker network. This nonlinear network can be easily solved by employing common deep learning frameworks like Pytorch.

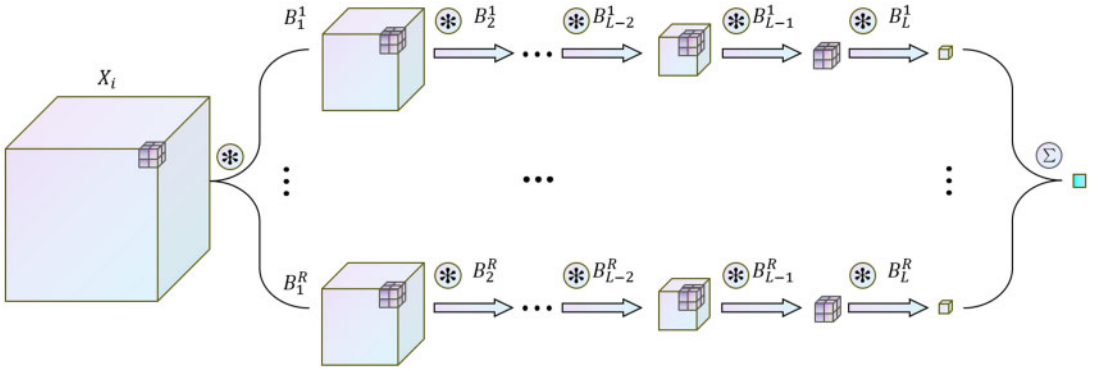

Fig. 2. An illustration of the deep Kronecker network in a convolutional form.

4. Deep Kronecker network and tensor regression

In this section, we demonstrate the connections between the deep Kronecker network and the tensor regression of Zhou et al. (2013). Specifically, we show that the deep Kronecker network, not only includes tensor regression as a special case, but it can also be implemented by applying tensor regression on certain reshaped images.

Suppose that a tensor $\mathcal{C} \in \mathbb{R}^{d\times p\times q}$ can be decomposed as $\mathcal{C} = \bigotimes_{l=L}^{1} \mathcal{B}_l$. Then the entries of $\mathcal{C}$ are characterized by $\mathcal{C}_{[h_1\cdots h_L],[j_1\cdots j_L],[k_1\cdots k_L]} = \prod_{l=1}^{L} [\mathcal{B}_l]_{h_l,j_l,k_l}$. Here the square brackets indicate grouping of indices. For example, the grouped index $[h_1 \cdots h_L]$ refers to the linear index $h_1 + (h_2 - 1)d_1 + \cdots + (h_L - 1)\prod_{l=1}^{L} d_l$. Now we define $\mathcal{T}\colon \mathbb{R}^{d\times p\times q} \to \mathbb{R}^{(d_1p_1q_1)\times\cdots\times(d_Lp_Lq_L)}$ as a reshaping operator from $\mathcal{C}$ to an L-order tensor $\mathcal{T}(\mathcal{C})$ with the entries characterized by

$$[\mathcal{T}(\mathcal{C})]_{[h_1j_1k_1],\ldots,[h_Lj_Lk_L]} = \mathcal{C}_{[h_1\cdots h_L],[j_1\cdots j_L],[k_1\cdots k_L]}.$$

By this operator, Batselier & Wong (2017) provided the following lemma to connect the Kronecker product and canonical tensor decomposition.

Lemma 1 (Batselier & Wong, 2017). *Given a tensor $\mathcal{C} \in \mathbb{R}^{d\times p\times q}$, if $\mathcal{C} = \sum_{r=1}^{R} \bigotimes_{l=L}^{1} \mathcal{B}_l^r$ then we have $\mathcal{T}(\mathcal{C}) = \sum_{r=1}^{R} b_1^r \circ \cdots \circ b_L^r$, where $b_l^r = \mathrm{vec}(\mathcal{B}_l^r)$, $l \in [L]$ and $r \in [R]$.*

Canonical tensor decomposition is frequently used to approximate tensors and is employed by Zhou et al. (2013) for image data analysis. As the reshaping operator $\mathcal{T}(\cdot)$ is one to one, Lemma 1 suggests that Kronecker product decomposition (2) is also able to approximate arbitrary tensors. Furthermore, we have the following theorem to connect the deep Kronecker network and tensor regression.

Theorem 2. *The low Kronecker rank in the deep Kronecker network is equivalent to a low-canonical-rank assumption on the reshaped tensors $\mathcal{T}(\mathcal{X}_i)$. Let $b_l^r = \mathrm{vec}(\mathcal{B}_l^r)$; we have*

$$g\{\mathbb{E}(y_i)\} = \left\langle \mathcal{X}_i, \sum_{r=1}^{R} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\rangle \quad \Longleftrightarrow \quad g\{\mathbb{E}(y_i)\} = \left\langle \mathcal{T}(\mathcal{X}_i), \sum_{r=1}^{R} b_1^r \circ \cdots \circ b_L^r \right\rangle.$$

Remark 1. Theorem 2 implies that the deep Kronecker network could be solved by a two-step procedure: (i) reshape the original images and (ii) apply a tensor regression algorithm, such as the block relaxation in Zhou et al. (2013), on the reshaped images. The reshaping step is crucial and results in different performances of the deep Kronecker network and tensor regression.

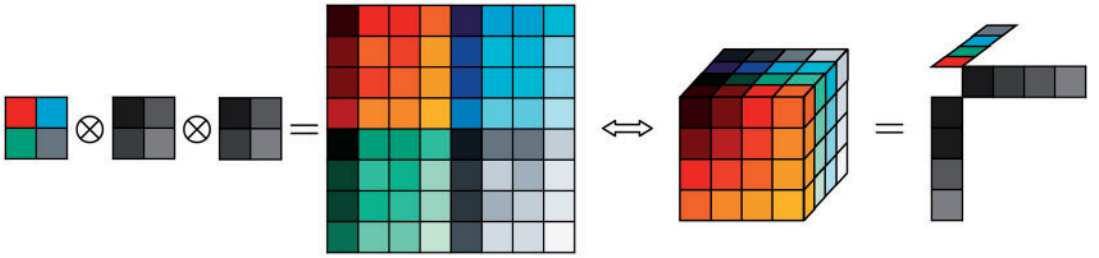

Fig. 3. An illustration of the Kronecker product and canonical tensor decomposition.

Remark 2. The deep Kronecker network includes tensor regression as a special case. Suppose that images are of size $D_1 \times D_2 \times D_3$. Then tensor regression could be viewed as a special deep Kronecker network with factor number $L = 3$ and factors $\mathcal{B}_1^r \in \mathbb{R}^{D_1 \times 1 \times 1}$, $\mathcal{B}_2^r \in \mathbb{R}^{1 \times D_2 \times 1}$, $\mathcal{B}_3^r \in \mathbb{R}^{1 \times 1 \times D_3}$ for $r \in [R]$. Under such a case, $\mathcal{T}(\mathcal{X}_i) = \mathcal{X}_i$. Thus, the deep Kronecker network is a more flexible and adaptive framework for allowing different sizes of factors.

Remark 3. The size of factors $\mathcal{B}_l^r$ and the number of layers L_r are allowed to be different across r. Under such a case, we can apply different reshaping operations $\mathcal{T}_r(\mathcal{X}_i)$ and obtain

$$g\{\mathbb{E}(y_i)\} = \sum_{r=1}^{R} \langle \mathcal{T}_r(\mathcal{X}_i), b_1^r \circ \cdots \circ b_{L_r}^r \rangle. \tag{4}$$

Model (4) is no longer a tensor regression model. However, it could still be solved by an alternating minimization algorithm with b_l^r iteratively updated by fixing $b_{l'}^{r'}$, $l' \neq l$, $r' \neq r$.

Remark 4. The deep Kronecker network implicitly imposes a blockwise smoothness structure on the coefficients, a property particularly suitable for image analysis. Figure 3 illustrates Kronecker product decomposition and its relation to canonical tensor decomposition. It is clear that the matrix created by the Kronecker product exhibits a blockwise smooth pattern.

5. Theoretical analysis

In this section, we show that the solution computed by an alternating minimization algorithm is guaranteed to converge to the truth, despite the problem being highly nonconvex. Our target is to bound the distance between the estimated coefficient $\hat{\mathcal{C}}$ and its true counterpart $\mathcal{C}$ when the network structure is correctly specified. In this context, the distance refers to the tensor angles. For two tensors $\mathcal{U}, \mathcal{V}$ of the same shape, define the distance, angle between, $\mathcal{U}$ and $\mathcal{V}$ as $\text{dist}^2(\mathcal{U}, \mathcal{V}) = 1 - \langle \mathcal{U}, \mathcal{V} \rangle^2 / (\|\mathcal{U}\|_F^2 \|\mathcal{V}\|_F^2)$, where $\|\cdot\|_F$ is the Frobenius norm. Here we focus on a rank-1 deep Kronecker network under the linear model, but our results can be extended to general cases.

Condition 1 (*Restricted isometry property*). Let $\mathcal{X}_i$ be the observed image tensors. Suppose that, for all $\mathcal{B}_l^r \in \mathbb{R}^{d_l \times p_l \times q_l}$, all $l \in [L]$ and $r = 1, 2$, there exists a constant $\delta \in (0, 1)$ such that

$$(1-\delta) \left\| \sum_{r=1}^{2} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\|_F^2 \leqslant \frac{1}{n} \sum_{i=1}^{n} \left\langle \mathcal{X}_i, \sum_{r=1}^{2} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\rangle^2 \leqslant (1+\delta) \left\| \sum_{r=1}^{2} \bigotimes_{l=L}^{1} \mathcal{B}_l^r \right\|_F^2.$$

Now we present an overview of our main theorem; comprehensive details can be found in the Supplementary Material.

Theorem 3. *Suppose that model* $y_i = \langle \mathcal{X}_i, \mathcal{C} \rangle + \epsilon_i$ *holds with* $\mathcal{C} = \bigotimes_{l=L}^{1} \mathcal{B}_l$. *Assume that Condition* 1 *holds with a small enough constant* δ *and* $\|\boldsymbol{\varepsilon}\|_2 \leqslant c(1-\delta)\|\mathcal{C}\|_F/2$ *for a certain constant* c. *Suppose that the likelihood function* (3) *is solved using an alternating minimization algorithm with a correctly*

Table 1. *Results of the ADNI analysis. The best-performing method is marked with an asterisk*

Task	Criterion	DKN	TR	TRlasso	CNN
Regression	RMSE	*0.2258	0.2627	0.2557	0.2909
Classification	Accuracy	*79.25%	66.80%	76.76%	78.01%

DKN, deep Kronecker network; TR, tensor regression; TRlasso, tensor regression with lasso penalty; CNN, convolutional neural network; RMSE, root-mean-square error.

specified network structure and a spectral initialization. Let $\kappa < 1$ *be a contraction parameter,* μ *be the initialization error and* $\tau = \sqrt{(1/n)}\log n$*. Then, after* t *iterations, the distance between estimates* $\hat{\mathcal{C}}^{(t)}$ *and* $\mathcal{C}$ *is bounded with high probability:*

$$\mathrm{dist}(\hat{\mathcal{C}}^{(t)}, \mathcal{C}) \leqslant c_1\kappa^t\mu + c_2\tau. \tag{5}$$

Remark 5. The first term on the right-hand side of (5) can be viewed as the optimization error, and the second term is the statistical error. Theorem 3 suggests that the optimization error decays geometrically, even if the objective function (3) is highly nonconvex. After $t \geqslant t_0 + \log(n^{-1}\log n)/2\log(\kappa)$ iterations, $\mathrm{dist}(\hat{\mathcal{C}}^{(t)}, \mathcal{C}) \asymp \sqrt{(1/n)}\log n$ holds with high probability.

Remark 6. Because of the connection between the deep Kronecker network and tensor regression, Theorem 3 also works for a tensor regression solved by the block relaxation algorithm. The spectral initialization required by Theorem 3 is essential, as it can be proved to be not too far away from the truth.

6. Alzheimer's Disease Neuroimaging Initiative analysis

In this section, we analyse Alzheimer's disease with data collected from the Alzheimer's Disease Neuroimaging Initiative (ADNI, `https://adni.loni.usc.edu/`), a study designed to detect and track Alzheimer's disease with clinical, genetic and imaging data. In the ADNI analysis, we use T1-weighted MRI scans with two types of outcome:

(i) binary outcomes for classification, indicating whether or not participants have Alzheimer's disease,
(ii) continuous outcomes for regression, indicating the mini-mental state examination score, a commonly used reference for the diagnosis of Alzheimer's disease.

The images, after pre-processing, are represented as tensors with dimension 64^3. We use the first two phases of ADNI (ADNI-1 and ADNI-GO) as training and the third phase ADNI-3 as testing, resulting in 417 subjects for training and 241 for testing. The deep Kronecker network is implemented with six layers, factors of size 2^3 and ranks tuned by the Bayesian information criterion. We compare the performance of the deep Kronecker network with three competing methods: the convolutional neural network, tensor regression and tensor regression with lasso penalty. We report the prediction results of the four methods in Table 1 and plot the estimated coefficients in Fig. 4.

From Table 1 and Fig. 4, we can see that the deep Kronecker network, not only delivers the most accurate prediction, but also detects the most precise regions. Moreover, the regions detected by the deep Kronecker network in classification and regression are consistent, focusing primarily around the hippocampus. This is in line with medical literature, which has established a connection between the hippocampus and Alzheimer's disease; see, e.g., Ball et al. (1985).

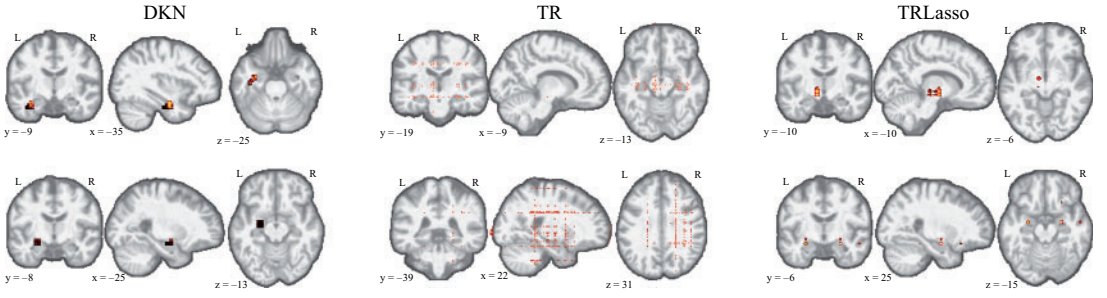

Fig. 4. Detected regions in regression (top row) and classification (bottom row).

ACKNOWLEDGEMENT

This work was funded in part by the Hong Kong RGC Grant ECS 21313922 and GRF 17301123.

SUPPLEMENTARY MATERIAL

The Supplementary Material includes additional theorems, numerical studies and proofs.

REFERENCES

BALL, M., HACHINSKI, V., FOX, A., KIRSHEN, A., FISMAN, M., BLUME, W., KRAL, V., FOX, H. & MERSKEY, H. (1985). A new definition of Alzheimer's disease: a hippocampal dementia. *Lancet* **325**, 14–16.

BATSELIER, K. & WONG, N. (2017). A constructive arbitrary-degree Kronecker product decomposition of tensors. *Numer. Lin. Algeb. Appl.* **24**, e2097.

CAI, C., CHEN, R. & XIAO, H. (2020). Kopa: automated Kronecker product approximation. *arXiv:* 1912.02392v3.

CHEN, E. Y., TSAY, R. S. & CHEN, R. (2020). Constrained factor models for high-dimensional matrix-variate time series. *J. Am. Statist. Assoc.* **115**, 775–93.

FENG, L., BI, X. & ZHANG, H. (2021). Brain regions identified as being associated with verbal reasoning through the use of imaging regression via internal variation. *J. Am. Statist. Assoc.* **116**, 144–58.

FUKUSHIMA, K. & MIYAKE, S. (1982). Neocognitron: a self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and Cooperation in Neural Nets*, S. Amari & M. A. Arbib, eds. Berlin: Springer, pp. 267–85.

GOLDSMITH, J., HUANG, L. & CRAINICEANU, C. M. (2014). Smooth scalar-on-image regression via spatial Bayesian variable selection. *J. Comp. Graph. Statist.* **23**, 46–64.

HACKBUSCH, W., KHOROMSKIJ, B. N. & TYRTYSHNIKOV, E. E. (2005). Hierarchical Kronecker tensor-product approximations. *J. Numer. Math.* **13**, 119–56.

HAFNER, C. M., LINTON, O. B. & TANG, H. (2020). Estimation of a multiplicative correlation structure in the large dimensional case. *J. Economet.* **217**, 431–70.

KANG, J., REICH, B. J. & STAICU, A.-M. (2018). Scalar-on-image regression via the soft-thresholded Gaussian process. *Biometrika* **105**, 165–84.

LECUN, Y., BOTTOU, L., BENGIO, Y. & HAFFNER, P. (1998). Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–324.

RUDIN, L. I., OSHER, S. & FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D* **60**, 259–68.

TIBSHIRANI, R., SAUNDERS, M., ROSSET, S., ZHU, J. & KNIGHT, K. (2005). Sparsity and smoothness via the fused lasso. *J. R. Statist. Soc.* B **67**, 91–108.

WANG, X., ZHU, H. & INITIATIVE, A. D. N. (2017). Generalized scalar-on-image regression models via total variation. *J. Am. Statist. Assoc.* **112**, 1156–68.

WU, S. & FENG, L. (2023). Sparse Kronecker product decomposition: a general framework of signal region detection in image regression. *J. R. Statist. Soc.* B **85**, 783–809.

ZHOU, H., LI, L. & ZHU, H. (2013). Tensor regression with applications in neuroimaging data analysis. *J. Am. Statist. Assoc.* **108**, 540–52.

[*Received on* 3 *October* 2022. *Editorial decision on* 9 *August* 2023]