Submitted to the Annals of Statistics

NONPARAMETRIC TRACE REGRESSION IN HIGH DIMENSIONS VIA SIGN SERIES REPRESENTATION

BY CHANWOO LEE[1], LEXIN LI[2], HAO HELEN ZHANG[3], AND MIAOYAN WANG[1]

[1]*Department of Statistics, University of Wisconsin-Madison, chanwoo.lee@wisc.edu; miaoyan.wang@wisc.edu*

[2]*Department of Biostatistics and Epidemiology, University of California-Berkeley, lexinli@berkeley.edu*

[3]*Department of Mathematics, University of Arizona, hzhang@math.arizona.edu*

Learning of matrix-valued data has recently surged in a range of scientific and business applications. Trace regression is a widely used method to model effects of matrix predictors and has shown great success in matrix learning. However, nearly all existing trace regression solutions rely on two assumptions: (i) a known functional form of the conditional mean, and (ii) a global low-rank structure in the entire range of the regression function, both of which may be violated in practice. In this article, we relax these assumptions by developing a general framework for nonparametric trace regression models via structured sign series representations of high dimensional functions. The new model embraces both linear and nonlinear trace effects, and enjoys rank invariance to order-preserving transformations of the response. In the context of matrix completion, our framework leads to a substantially richer model based on what we coin as the "sign rank" of a matrix. We show that the sign series can be statistically characterized by weighted classification tasks. Based on this connection, we propose a learning reduction approach to learn the regression model via a series of classifiers, and develop a parallelable computation algorithm to implement sign series aggregations. We establish the excess risk bounds, estimation error rates, and sample complexities. Our proposal provides a broad nonparametric paradigm to many important matrix learning problems, including matrix regression, matrix completion, multi-task learning, and compressed sensing. We demonstrate the advantages of our method through simulations and two applications, one on brain connectivity study and the other on high-rank image completion.

1. Introduction. Matrix-valued data are rising ubiquitously in modern data science applications, for instance, brain neuroimaging analysis, integrative genomics, and sensor network localization. Trace regression is one of the most commonly used approaches for modeling matrix data [11, 17]. The model characterizes the relationship between a scalar response Y and a high dimensional matrix predictor $\boldsymbol{X} \in \mathcal{X} \subset \mathbb{R}^{d_1 \times d_2}$ as

$$Y = \langle \boldsymbol{X}, \boldsymbol{B} \rangle + \varepsilon, \text{ with } \boldsymbol{B} \in \mathbb{R}^{d_1 \times d_2} \text{ and } \operatorname{rank}(\boldsymbol{B}) \le r, \tag{1}$$

where ε is a zero-mean sub-Gaussian noise, and $r \in \mathbb{N}_+$ is the matrix rank typically assumed fixed and much smaller than $\min(d_1, d_2)$. The function $\boldsymbol{X} \mapsto \langle \boldsymbol{X}, \boldsymbol{B} \rangle = \operatorname{tr}(\boldsymbol{X}\boldsymbol{B}^T)$ is called the trace effect, where $\operatorname{tr}(\cdot)$ denotes the matrix trace. Over the last decade, the low-rank trace regression (1) has been studied intensively in numerous contexts, including matrix predictor regression, matrix completion, multi-task learning, and compressed sensing.

- **Matrix predictor regression.** Linear trace regression (1) was first proposed to model a matrix-valued predictor [44, 38], and was later generalized to model an exponential family response with a known link function [37, 11].

MSC2020 subject classifications: Primary 62G05; secondary 62H30.

Keywords and phrases: Matrix trace model, Matrix completion, Matrix predictor regression, Nonparametric regression, Sparsity.

- **Matrix completion.** In addition to the usual regression setting, another application of trace regression (1) is matrix completion, where the goal is to fill in the missing entries of a partially observed matrix [2]. Suppose the predictor space $\mathcal{X}$ consists of basis matrices $\boldsymbol{a}_i^T\boldsymbol{b}_j$ in $\mathbb{R}^{d_1\times d_2}$, with $\boldsymbol{a}_i \in \mathbb{R}^{d_1}$ (respectively, $\boldsymbol{b}_j \in \mathbb{R}^{d_2}$) being the basis vector with 1 at the i-th (respectively, jth) position and 0 elsewhere. Let $\mathbb{P}_{\boldsymbol{X}}$ be a uniform distribution over $\mathcal{X}$. Then model (1) reduces to a matrix completion problem, $Y_{ij} = \langle \boldsymbol{a}_i^T\boldsymbol{b}_j, \boldsymbol{B}\rangle + \varepsilon_{ij} = B_{ij} + \varepsilon_{ij}$, where $Y_{ij}, B_{ij} \in \mathbb{R}$ denotes the (i,j)-th entry of the data matrix $\boldsymbol{Y}$ and the signal matrix $\boldsymbol{B}$, respectively, for $(i,j) \in \Omega \subset \{1,\dots,d_1\}\times\{1,\dots,d_2\}$ in the observed index set. Moreover, the model becomes a matrix denosing problem [40] when the observation set is complete, i.e, $\Omega = \{1,\dots,d_1\}\times\{1,\dots,d_2\}$.
- **Multi-task learning.** Another application of trace regression is multi-task learning, where the goal is to predict one task response by leveraging the structural similarities among multiple tasks. Here the predictor space $\mathcal{X}$ consists of only matrices that have a single non-zero row. The multi-task problem collects n observations from d_1 different supervised learning tasks. Each task is modeled as a linear regression with an unknown d_2-dimensional parameter $\boldsymbol{b}_i, i = 1,\dots,d_1$, and the collection of $\boldsymbol{b}_i$ forms the rows of $\boldsymbol{B}$. The model exploits similarities among multiple tasks to predict the response of the i-th task [4, 11].
- **Compressed sensing.** Compressed sensing is also a special application of trace regression, where the goal is to recover the structured matrix $\boldsymbol{B}$ from multiple linear combinations of the entry observations. The space $\mathcal{X}$ is the family of measurement matrices given the sampling schemes. For example, Gaussian ensembles use random matrices $\boldsymbol{X}$ with i.i.d. entries from a standard normal distribution [3], while factorized ensembles use rank-1 matrices $\boldsymbol{X} = \boldsymbol{u}\boldsymbol{v}^T$ for two random vectors $\boldsymbol{u} \in \mathbb{R}^{d_1}, \boldsymbol{v} \in \mathbb{R}^{d_2}$ [26].

In this article, we propose and study a nonparametric extension of the trace regression model (1), which encompasses all above matrix learning problems. Particularly, we illustrate our method with two common problems, i.e., matrix predictor regression and matrix completion.

1.1. *Inadequacy of low-rank trace regression.* The existing trace regression model (1) and its variants rely on two key assumptions: the relationship between $\mathbb{E}(Y|\boldsymbol{X})$ and the trace effect is known a priori through some link function, and the matrix effect is encoded by a global low-rank matrix $\boldsymbol{B}$ in the entire function range. However, despite the popularity of trace regressions, these assumptions are stringent and may often be violated in practice. Next, we use two examples to illustrate the limitations of the classical low-rank trace regression. We present the pitfall in the context of matrix completion, and similar phenomena also occur in general matrix predictor regression.

In the first example, we show the sensitivity of low-rank matrix models to order-preserving transformations. Let $\boldsymbol{B} = \boldsymbol{U}^T\boldsymbol{V} \in \mathbb{R}^{d\times d}$ be a rank-5 matrix, where $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{R}^{d\times 5}$ consists of i.i.d. standard normal entries and $d = 50$. Now suppose a monotonic transformation $g(b) = (1+\exp(-cb))^{-1}$ is applied to $\boldsymbol{B}$ entry-wise, and we let $g(\boldsymbol{B})$ be the signal matrix prior to measurements. A small c implies an approximate linearity $b \mapsto -cb$, whereas a large c implies a high nonlinearity $b \mapsto \{0,1\}$. Fig 1(a) shows that the numerical rank of $g(\boldsymbol{B})$ increases rapidly with c, rendering the classical low-rank model ineffective. In genomic signal processing and other applications, the matrix of interest often undergoes unknown transformation prior to measurements. The sensitivity makes low-rank models less desirable as the global low-rank structure fails to be preserved through monotonic transformations.

In the second example, we show the failure of the classical low-rank model in representing a structured but high-rank effect. We again consider the matrix completion for simplicity, but this time, from a full-rank signal matrix $\boldsymbol{B} \in \mathbb{R}^{d\times d}$, where the (i,j)-th entry is $\log(1+$

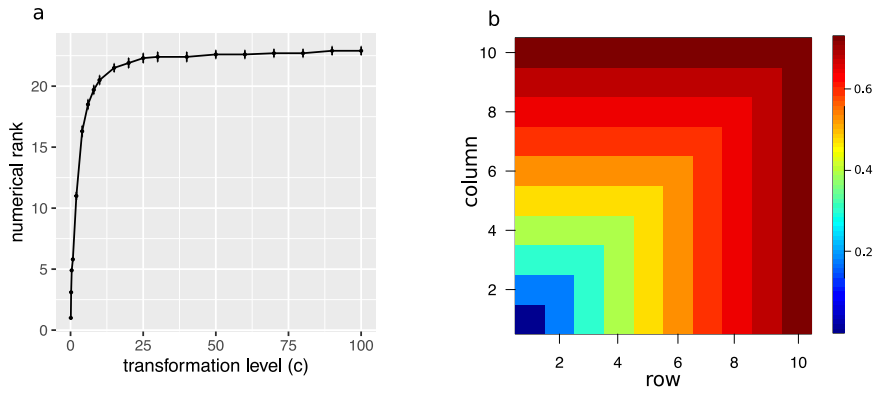

FIG 1. *Two examples of high-rank matrix trace models. (a) The numerical rank of the matrix* $g(\boldsymbol{B})$ *versus* c *in the transformation, where the numerical rank is defined by* $\text{rank}(g(\boldsymbol{B})) = \min\{\text{rank}(\boldsymbol{C}) : \|\boldsymbol{C} - g(\boldsymbol{B})\|_F \leq 0.01\|g(\boldsymbol{B})\|_F\}$. *The error bar represents standard errors from 10 realizations of* $\boldsymbol{B}$. *(b) Heatmap of a full-rank matrix* $\boldsymbol{B} \in \mathbb{R}^{d\times d}$ *with the* (i,j)*-th entry equal to* $\log(1+\max(i,j))$. *In (a),* $d = 50$*, and in (b),* $d = 10$.

$\max(i,j)/d)$ and $d = 10$. Fig 1(b) shows that $\boldsymbol{B}$ is clearly structured, but is of full-rank that $\text{rank}(\boldsymbol{B}) = d$. The classical low-rank model is again ineffective in this case.

These examples reveal the inadequacy of the conventional low-rank trace model (1) in capturing important yet complex matrix effects. This has motivated us to develop a flexible class of nonparametric trace regression for modeling and estimating nonlinear, local, and possibly high-rank effects for high dimensional matrices. We later revisit these two examples in Section 2, and show how those limitations can be overcome using a richer class of matrix models based on a new concept what we coin as the matrix "sign rank".

1.2. *Our proposal and contributions.* In this article, we first propose a new notion of low-rank sign representable function, then develop a flexible class of nonparametric trace regression models based on this representation, as well as relevant theory and computational algorithms. Our proposal makes useful contributions on multiple fronts.

First, the proposed work fills a crucial gap between a global parametric model and a local nonparametric model in the literature of matrix modeling. We develop a new nonparametric regression paradigm – structured sign representations – to address the challenges previously difficult or infeasible in trace regressions, especially in the high dimensional regime where $d_1 d_2 \gg n$. Existing literature on matrix regressions almost exclusively focuses on low-rank trace effects in the global scale. However, such a premise often fails, where the rank of global effects may grow with the matrix dimension. By contrast, our proposed model enjoys rank invariance under monotonic transformations, and permits both low-rank and high-rank effects through aggregations of sign representation functions. We show that the low-rank sign functions not only preserve all information for conventional low-rank models, but also provide powerful tools for extracting nonlinear, high-rank trace effects and estimating them accurately. Our framework is flexible and applicable to high-rank matrix learning problems, and it greatly expands the horizon of conventional low-rank matrix models.

Second, we show that the sign function series can be statistically characterized by classification tasks with carefully specified weights. This characterization converts a complex and hard regression problem, *"what is the value of the nonparametric regression function?"* to a series of simpler and easier classification problems, *"does the regression function fall below a threshold?"* Correspondingly, we develop a learning reduction approach to estimate the regression function via a series of classifiers, by leveraging classification solutions from existing state-of-art computational algorithms. Theoretically, we establish the excess

risk bounds, estimation error rates, and sample complexities. Particularly, our error bound reveals the well-controlled complexity from sign estimation to regression, where

$$\text{sign function error} \lesssim \underbrace{t_n^{\alpha/(2+\alpha)}}_{\text{classification error}},$$

$$\text{regression error} \lesssim \underbrace{t_n^{\alpha/(2+\alpha)} \log H}_{\text{estimation error inherited from classification}} + \underbrace{\tfrac{1}{H}}_{\text{reduction bias}} + \underbrace{t_n H \log H}_{\text{reduction variance}},$$

in which $\alpha \geq 0$ quantifies the smoothness of the nonparametric regression function, $H \in \mathbb{N}_+$ is a resolution parameter that specifies the total number $(2H+1)$ of sign functions to aggregate in our algorithm, $t_n = t_n(d,n) \to 0$ quantifies the convergence rate depending on the specific model, and $d = d_1 = d_2$ for simplicity. In particular, we establish $t_n \asymp n^{-1} \log d$ under a two-way sparse non-parametric trace regression model (see Section 4.1), and $t_n \asymp n^{-1} d$ under a low sign rank non-parametric matrix completion model (see Section 4.2). These results imply that a low sample complexity with respect to the matrix dimension. Note that the sign function estimation reaches a faster $\mathcal{O}(n^{-1})$ rate compared to the $\mathcal{O}(n^{-1/2})$ regression rate when $\alpha = \infty$, which confirms our premise that sign estimation is easier than regression. To our knowledge, these statistical guarantees are among the first for the learning reduction approach in the context of nonparametric matrix regression.

Lastly, we develop an alternating direction method of multipliers (ADMM) algorithm for optimization with a family of large-margin loss functions. From the computational and learning perspectives, the proposed method can be characterized as the **A**ggregation of **S**tructured **SI**gn **S**eries for **T**race regression (**ASSIST**). We show that the **ASSIST** algorithm leverages recent advances in large-margin solvers as well as non-convex optimization for low-rank, two-way sparse matrix learning. As demonstrated in our simulations and real data applications, the **ASSIST** method contributes a new matrix modeling tool of easy interpretability and accurate prediction.

1.3. *Related work.* Nonparametric learning for matrix data is much more challenging than standard multivariate data. Naively turning a matrix into a vector followed by a classical vector based nonparametric method can destroy rich structural information encoded in the matrix data. Moreover, most nonparametric methods rely on some notion of smoothness in a local neighborhood of the predictors. In the context of matrix regressions, however, the predictor space is huge, rendering the "local smoothness" assumption less practical, which is partially why the topic is barely explored by data with a limited sample size.

Our work is related to but also clearly distinctive from several lines of existing research. The first line is the classical trace regression [11, 17]. The key difference is that the existing solutions all adopt a parametric model with a global low-rank structure. By contrast, our method is nonparametric and embraces nonlinear, local, and possibly high-rank effects for high dimensional matrices.

The second line is the recent development of nonparametric methods with matrix-valued or tensor-valued data. In imaging analysis, convolution neural networks (CNNs) have been widely adopted as a nonparametric tool for prediction given matrix-valued images [16]. In contrast, our proposal studies not only prediction, but also estimation and interpretability, with the theoretical guarantees. We also numerically compare our method with CNNs. Hao et al. [18] proposed a sparse additive model with tensor predictors by extending the usual spline basis functions. Zhou et al. [45] studied tensor predictors and proposed a broadcasting operation to introduce nonlinearity to individual tensor entries. Our nonparametric solution has broader implications than those approaches in estimating local low-rank effects. Our sign series representation of function bridges the gap between regression and classification in high

dimensions, and naturally lends the problem to a learning reduction type solution. Moreover, although a matrix can be viewed as a two-dimensional tensor, the problem of nonparametric learning for matrix data itself is more parsimonious and deserves a full investigation. We leave the counterpart problem for nonparametric tensor regression as future research.

The third line is function sign estimation, which is in turn related to classification, or more generally, the level set estimation. The latter problem has a long history in statistics [32] and computational mathematics [15]. Particularly, Wang et al. [35] proposed a conditional probability estimation method based on support vector machines (SVMs), but their results were restricted to a fixed number of features and vector predictors only. Singh et al. [31] proposed a tree based method for multiple sets extraction, but their goal was level set estimation instead of function estimation. None of these methods address the regression problem or high dimensional matrix predictors. By contrast, we bridge the problems of level set estimation and nonparametric regression using low-rank sign series representations. Instead of constructing a point-wise function in the domain space, the sign representation partitions the domain space based on the function range. The benefit bears the analogy of Lebesgue versus Riemann integrals in functional analysis, in the sense that the neighborhood is determined by the range space instead of the domain space. The former approach is especially appealing for matrix regressions, where the range space is determined by a simple scalar response, whereas the domain space is huge and high dimensional.

1.4. *Notation and organization.* We adopt the following notation throughout this article. Let $\mathcal{X} \subset \mathbb{R}^{d_1 \times d_2}$ denote the feature space equipped by some measure $\mathbb{P}_{\boldsymbol{X}}$. For a function $f\colon \mathcal{X} \to \mathbb{R}$, let $\operatorname{sgn} f$ denote its sign function, i.e., $\operatorname{sgn} f(\boldsymbol{X}) = 1$ if $f(\boldsymbol{X}) > 0$ and $\operatorname{sgn} f(\boldsymbol{X}) = -1$ otherwise. Let $\|f\|_1$ denote its L_1 norm, where we define $\|f\|_1 = \mathbb{E}|f(\boldsymbol{X})|$ with the expectation taken with respect to $\boldsymbol{X} \sim \mathbb{P}_{\boldsymbol{X}}$. For a set $A \subset \mathcal{X}$, let $\operatorname{sgn}(\boldsymbol{X} \in A)$ denote the sign function induced by A, i.e., a function taking value 1 on the event $\{\boldsymbol{X} \in A\}$ and -1 otherwise. Let $[n] = \{1, \ldots, n\}$, and $|\cdot|$ denote the cardinality. Let $\|\cdot\|_p$ denote the vector p-norm for $p \geq 0$. For a matrix $\boldsymbol{B} \in \mathbb{R}^{d_1 \times d_2}$, let $\boldsymbol{B}_i$ denote its i-th row and B_{ij} its (i,j)-th entry. Let $\|\boldsymbol{B}\|_{p,q}$ denote the matrix (p,q)-norm such that $\|\boldsymbol{B}\|_{p,q} = \|\boldsymbol{b}\|_q$, where $\boldsymbol{b} = (\|\boldsymbol{B}_1\|_p, \ldots, \|\boldsymbol{B}_{d_1}\|_p)^T \in \mathbb{R}^{d_1}$ consists of the p-norms for each row of $\boldsymbol{B}$. In particular, let $\|\boldsymbol{B}\|_{1,0} = |\{i \in [d_1]\colon \boldsymbol{B}_i \neq 0\}|$ denote the number of non-zero rows in $\boldsymbol{B}$. Let $\|\boldsymbol{B}\|_F = \sqrt{\langle \boldsymbol{B}, \boldsymbol{B} \rangle}$ denote the matrix Frobenius norm, and $\|\boldsymbol{B}\|_\infty = \max_{(i,j)} |B_{ij}|$ the matrix maximum norm. Denote $a_n \asymp b_n$ if $c_1 \leq \lim_{n\to\infty} a_n/b_n \leq c_2$ for some constants $c_1, c_2 > 0$, and denote $a_n \lesssim b_n$ if $\lim_{n\to\infty} a_n/b_n \leq c$ for some constant $c \geq 0$. Let $\mathcal{O}(\cdot)$ denote the big-O notation, $\widetilde{\mathcal{O}}(\cdot)$ the variant that hides the logarithmic factors, and $\mathbb{1}(\cdot)$ the indicator function. Whenever applicable, the basic arithmetic operators are applied to a matrix in an element-wise manner.

The rest of the article is organized as follows. Section 2 presents the low-rank sign representable functions and our nonparametric trace regression model. Section 3 develops the learning reduction approach through weighted classifications, and establishes the corresponding statistical guarantees. Section 4 specializes the general theory to two concrete learning problems, the low-rank sparse matrix predictor regression and the high-rank matrix completion. Section 5 studies the large-margin based estimation and develops an optimization algorithm. Section 6 presents the simulations, and Section 7 two real data applications. Section 8 concludes with a discussion. All technical proofs and additional results are relegated to the Supplementary Appendix.

2. Nonparametric trace regression model. In this section, we present our nonparametric trace regression model. Let $\boldsymbol{X} \in \mathcal{X} \subset \mathbb{R}^{d_1 \times d_2}$ denote the matrix predictor, $Y \in \mathbb{R}$ the scalar response, and $\mathbb{P}_{\boldsymbol{X},Y}$ the joint probability distribution. We consider the model,

$$Y = f(\boldsymbol{X}) + \varepsilon, \tag{2}$$

where $f\colon \mathcal{X} \mapsto \mathbb{R}$ is an unknown regression function of interest, and ε is a mean-zero noise. For a cleaner exposition, we assume the noise is bounded and the range of Y is in $[-1, 1]$; the extension to a sub-Gaussian noise is provided in Section A.2 of the Appendix. In addition, we allow a heterogeneous noise such that ε may depend on $\boldsymbol{X}$. Model (2) therefore incorporates both continuous and binary-valued responses. For instance, we allow the binary regression problem where Y is a $\{0, 1\}$-label from a Bernoulli distribution, in which case, the noise variance depends on the mean, and f represents the conditional probability, $f(\boldsymbol{X}) = \mathbb{P}(Y = 1|\boldsymbol{X})$. Our goal is to estimate the regression function $f(\boldsymbol{X}) = \mathbb{E}(Y|\boldsymbol{X})$ based on n i.i.d. training samples $(\boldsymbol{X}_i, Y_i)_{i=1,\dots,n}$.

We next introduce the notion of low-rank sign representable function, which is essential to bridge the usual global low-rank trace models to nonparametric local low-rank trace models.

DEFINITION 1 (Rank-r sign representable function). A function $f\colon \mathcal{X} \mapsto [-1, 1]$ is called (r, π)-sign representable, for a given level $\pi \in [-1, 1]$ and a rank $r \in \mathbb{N}_+$, if the function $(f - \pi)$ has the same sign as a rank-r trace function; that is,

$$\operatorname{sgn}(f(\boldsymbol{X}) - \pi) = \operatorname{sgn}(\langle \boldsymbol{X}, \boldsymbol{B}\rangle + b), \quad \text{for all } \boldsymbol{X} \in \mathcal{X}, \tag{3}$$

where $\boldsymbol{B} = \boldsymbol{B}(\pi)$ is a rank-r matrix, and $b = b(\pi)$ is the intercept. A function f is called globally rank-r sign representable, if f is (r, π)-sign representable for all $\pi \in [-1, 1]$. Let $\mathcal{F}_{\text{sgn}}(r)$ denote the rank-r sign representable function family, and let $\Phi(r) = \{\phi\colon \boldsymbol{X} \mapsto \langle \boldsymbol{X}, \boldsymbol{B}\rangle + b \,\big|\, \operatorname{rank}(\boldsymbol{B}) \le r, (\boldsymbol{B}, b) \in \mathbb{R}^{d_1 \times d_2} \times \mathbb{R}\}$ denote the rank-r trace function family.

Next, we show that (2) and (3) together form a very general family of models that incorporate most existing matrix regression models, including the low-rank trace regression, single index models, and high-rank matrix completion model.

EXAMPLE 1 (Generalized trace regression). The linear and generalized trace regression [44, 37, 11] imposes that $f(\boldsymbol{X}) = g(\langle \boldsymbol{X}, \boldsymbol{B}\rangle)$ with a known link function g and a rank-r coefficient matrix $\boldsymbol{B}$. By definition, $\operatorname{sgn}(f(\boldsymbol{X}) - \pi) = \operatorname{sgn}(\langle \boldsymbol{X}, \boldsymbol{B}\rangle - g^{-1}(\pi))$ holds for every π in the function range. Therefore, our model includes the generalized trace regression, i.e, $f \in \mathcal{F}_{\text{sgn}}(r)$. In particular, the usual trace model corresponds to the identity link g. More generally, any monotonic g is allowed as the link function, e.g., the logistic function $g(z) = (1 + \exp(-z))^{-1}$, the arctangent function $g(z) = 1/\pi \arctan(z) + 1/2$, the rectified linear unit (ReLU) function $g(z) = \max(0, z)$, and any inverse cumulative distribution function.

EXAMPLE 2 (Single index regression model). The monotonic matrix predictor single index model [1, 12] assumes a similar form of regression function $f(\boldsymbol{X}) = g(\langle \boldsymbol{X}, \boldsymbol{B}\rangle)$ with a low-rank $\boldsymbol{B}$ and a monotonic g, but the form of g is unknown. By definition, our model family $\mathcal{F}_{\text{sgn}}(r)$ incorporates the single index model and does not require to know g a priori.

EXAMPLE 3 (Multivariate normal mixture). The prospective model from matrix linear discriminant analysis [21] considers a binary response $Y = \{0, 1\}$, and assumes the matrix $\boldsymbol{X}|Y$ follows a Gaussian mixture distribution, $\boldsymbol{X}|\{Y = i\} = \boldsymbol{B}_0 + \boldsymbol{B} \times i + \boldsymbol{E}_i, i = 0, 1$, where $\boldsymbol{B}_0$ is an arbitrary baseline matrix, $\boldsymbol{B}$ is a rank-r matrix, and $(\boldsymbol{E}_i)_{i=0,1}$ are two mutually independent noise matrices with i.i.d. standard normal entries. Our model incorporates this model, by noting that $f(\boldsymbol{X}) = \mathbb{E}(Y|\boldsymbol{X}) = \text{logistic}(\langle \boldsymbol{B}, \boldsymbol{X}\rangle + b)$ for some $b \in \mathbb{R}$, and thus $f \in \mathcal{F}_{\text{sgn}}(r)$.

Definition 1 leads to another notion, the matrix sign rank, which is important for applying our proposed model for matrix completion as a special nonparametric trace regression. Specifically, for a given matrix $\boldsymbol{\Theta} \in \mathbb{R}^{d_1 \times d_2}$, define its sign rank as:

$$\operatorname{srank}(\boldsymbol{\Theta}) = \min\left\{\operatorname{rank}(\boldsymbol{\Theta}')\colon \operatorname{sgn}(\boldsymbol{\Theta}') = \operatorname{sgn}(\boldsymbol{\Theta}),\ \boldsymbol{\Theta}' \in \mathbb{R}^{d_1 \times d_2}\right\}.$$

This concept is important in areas such as combinatorics [8] and quantum mechanics [9], and, to our knowledge, we are the first to exploit this notion for nonparametric learning. To better understand its relation to the proposed nonparametric trace regression, we consider model (3) with the predictor space $\mathcal{X} = \{\boldsymbol{a}_i^T \boldsymbol{b}_j \colon (i,j) \in [d_1] \times [d_2]\}$, and $\boldsymbol{a}_i \in \mathbb{R}^{d_1}, \boldsymbol{b}_j \in \mathbb{R}^{d_2}$ are the basis vectors. For matrix completion, a function f over $\mathcal{X}$ is equivalently represented by a d_1-by-d_2 signal matrix $\boldsymbol{\Theta} = [\![f(\boldsymbol{e}_i^T \boldsymbol{e}_j)]\!]$. Our proposed function family $\mathcal{F}_{\text{sgn}}(r)$ essentially defines a new family of structured matrices with a low sign rank, as shown in the next proposition.

PROPOSITION 1 (Sign-representable function over basis matrices). Consider the predictor space $\mathcal{X} = \{\boldsymbol{a}_i^T \boldsymbol{b}_j \colon (i,j) \in [d_1] \times [d_2]\}$. We represent a bounded function $f \colon \mathcal{X} \to [-1,1]$ by its function values organized as a matrix $\boldsymbol{\Theta} = [\![f(\boldsymbol{a}_i^T \boldsymbol{b}_j)]\!] \in [-1,1]^{d_1 \times d_2}$, for basis vectors $\boldsymbol{a}_i \in \mathbb{R}^{d_1}, \boldsymbol{b}_j \in \mathbb{R}^{d_2}$. If f is rank-r sign representable, then $\max_{\pi \in [-1,1]} \text{srank}(\boldsymbol{\Theta} - \pi) \le r + 1$ (the constant 1 is due to the intercept in (3)). Conversely, if $\max_{\pi \in [-1,1]} \text{srank}(\boldsymbol{\Theta} - \pi) \le r$, then $\boldsymbol{\Theta}$ defines a rank-r sign representable function f.

Define the sign-r representable family for the signal matrix in matrix completion.

$$\mathcal{M}_{\text{sgn}}(r) = \{\boldsymbol{\Theta} \colon \max_{\pi \in [-1,1]} \text{srank}(\boldsymbol{\Theta} - \pi) \le r, \ \|\boldsymbol{\Theta}\|_\infty \le 1\}.$$

The family $\mathcal{M}_{\text{sgn}}(r)$ is a special case of the function family $\mathcal{F}_{\text{sgn}}(r)$ in Definition 1 with $b = 0$ and the predictor space $\mathcal{X} = \{\boldsymbol{a}_i^T \boldsymbol{b}_j \colon (i,j) \in [d_1] \times [d_2]\}$. We next further compare the sign rank with the matrix rank in this setting.

PROPOSITION 2 (Sign-rank vs. matrix rank). Consider the setting in Proposition 1. Then,

(a) $\max_{\pi \in [-1,1]} \text{srank}(\boldsymbol{\Theta} - \pi) \le \text{rank}(\boldsymbol{\Theta}) + 1$.
(b) If $\boldsymbol{\Theta} \in \mathcal{M}_{\text{sgn}}(r)$, then $g(\boldsymbol{\Theta})/\|g(\boldsymbol{\Theta})\|_\infty \in \mathcal{M}_{\text{sgn}}(r+1)$ for any strictly monotonic function $g \colon \mathbb{R} \to \mathbb{R}$. Here $g(\boldsymbol{\Theta})$ denotes the matrix by applying $g(\cdot)$ to $\boldsymbol{\Theta}$ entry-wise.
(c) For every dimension d, there exists a d-by-d matrix $\boldsymbol{\Theta} \in \mathcal{M}_{\text{sgn}}(2)$ such that $\text{rank}(\boldsymbol{\Theta}) = d$.

Proposition 2 highlights the advantages of using the sign rank in the high dimensional matrix analysis. The first property implies that classical low-rank matrix model is a special case of our low sign rank model. The second property shows that, compared to the matrix rank, the sign rank remains nearly invariant under monotonic transformations, since $\text{srank}(g(\boldsymbol{\Theta})) \le 1 + \text{srank}(\boldsymbol{\Theta})$ for all monotonic functions g. The last property shows that the sign rank can be dramatically smaller than the conventional matrix rank. Therefore, our model $\mathcal{M}_{\text{sgn}}(r)$ is strictly richer than the usual low-rank model.

A key advantage about the sign rank concept is that the low sign rank assumption is more relaxed and hence more realistic than the classical low matrix rank assumption. We next revisit the high-rank matrix model in Fig 1(a) to show that $\boldsymbol{B}$ is of a high matrix rank but a low sign rank. Meanwhile, we provide some additional examples of low sign rank matrices in Section A.1 of the Appendix, including matrices with repeating patterns [5], banded matrices, and the identity matrix.

EXAMPLE 4 (Single index model based matrix completion). For the model in Fig 1(a), $g(\boldsymbol{B})$ is a low sign rank matrix because $\text{srank}(g(\boldsymbol{B}) - \pi) \le 1 + \text{rank}(\boldsymbol{B}) = 6$ for all π in the function range. However, $g(\boldsymbol{B})$ itself is often high-rank as shown in Fig 1(a).

EXAMPLE 5 (High-rank matrix completion model). For the model in Fig 1(b), the matrix $\boldsymbol{B} = [\![\log(1 + \max(i,j)/d)]\!]$ is full-rank. Remarkably, this high-rank matrix belongs to our

sign representable function with rank 2, i.e., $\boldsymbol{B} \in \mathcal{M}_{\text{sgn}}(2)$. This is because srank$(\boldsymbol{B} - \pi) =$ srank$(\bar{\boldsymbol{B}})$, where $\bar{\boldsymbol{B}} = [\![\text{sgn}(\max(i,j) - e^{\pi} + 1)]\!]$ is a block matrix with rank at most 2. More generally, matrices of the type $\boldsymbol{B} = [\![g(\max(i,j)/d)]\!]$ belong to $\mathcal{M}_{\text{sgn}}(2r)$, where $g(\cdot)$ is a polynomial of degree r. See Section A.1 of the Appendix.

Our proposed nonparametric matrix regression model $\mathcal{F}_{\text{sgn}}(r)$ therefore implies a new matrix completion model in $\mathcal{M}_{\text{sgn}}(r)$. In next sections, we establish the general theory for $\mathcal{F}_{\text{sgn}}(r)$ first, then specialize the results to the high-rank completion problems in Section 4.2.

3. From classification to regression: a learning reduction approach. In this section, we present a learning reduction approach to estimate f from the model as specified in (2) and (3). Our main crux is to provably convert the regression estimation problem into a series of sign function estimation problems, which are in turn solved by weighted classifications.

More specifically, we dichotomize the response Y_i into a series of binary observations, $\text{sgn}(Y_i - \pi)$, for $\pi \in \mathcal{H} = \{-1, \ldots, -1/H, 0, 1/H, \ldots, 1\}$, where $H \in \mathbb{N}_+$ is a resolution parameter that controls the total number of sign functions to estimate. Then, for each π, we estimate the sign function $\text{sgn}(f - \pi)$ by performing a classification task,

$$\hat{\phi}_\pi = \underset{\phi \in \Phi(r)}{\arg\min} \frac{1}{2n} \sum_{i=1}^{n} \text{weighted-classification}(\text{sgn}(Y_i - \pi),\ \text{sgn}\phi(\boldsymbol{X}_i)), \tag{4}$$

where $\Phi(r)$ is the collection of rank-r trace functions, and the weighted classification$(\cdot, \cdot)$ denotes a classification objective function with a response-specific weight to each sample point. The weight in the objective function is crucial in our method, and we will detail the form in next section. Our final regression function estimate takes the form,

$$\hat{f} = \frac{1}{2H+1} \sum_{\pi \in \mathcal{H}} \text{sgn}\hat{\phi}_\pi. \tag{5}$$

We comment that the $(2H+1)$ estimation tasks of the sign functions are fully separable, leading naturally to a parallel type computation. Moreover, the sign functions bridge the problems of level set estimation and Bayes classification, as we will detail in Section 3.2. Fig 2 illustrates our main idea graphically. We refer to our method as the **A**ggegration of **S**tructured **SI**gn **S**eries for **T**race regression, and abbreviate it as **ASSIST**.

Next, we describe the specific form of weighted classification, the uniqueness of the classification optimizer, as well as the accuracy guarantee of the estimator.

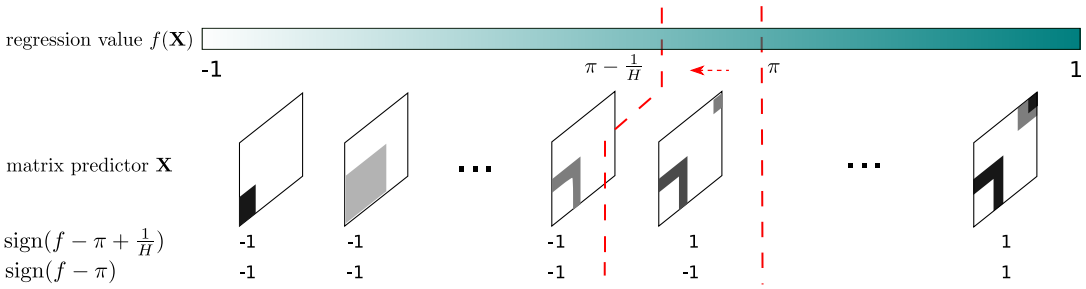

FIG 2. *Nonparametric matrix regression via sign function series estimation. We use a series of weighted classifications to estimate the sign functions, then obtain the regression function estimate via sign aggregations. Here,* $\boldsymbol{X} \in \mathcal{X}$ *denotes matrix-valued predictor,* $f\colon \mathcal{X} \to \mathbb{R}$ *denotes regression function, and* $\text{sgn}(f - \pi) \in \{-1, 1\}$ *is the sign function, where* $\pi \in \{-1, \ldots, -1/H, 0, 1/H, \ldots, 1\}$ *is the series of levels to aggregate in our algorithm.*

3.1. *Statistical characterization of sign functions via weighted classification.* For a given level $\pi \in [-1,1]$, define the π-shifted response $\bar{Y}_{\pi,i} = Y_i - \pi$ for $i \in [n]$. We propose a weighted classification objective function in (4) using

$$L(\phi; (\boldsymbol{X}_i, \bar{Y}_{\pi,i})_{i\in[n]}) = \frac{1}{2n}\sum_{i=1}^{n} \underbrace{|\bar{Y}_{\pi,i}|}_{\text{response-specific weight}} \times \underbrace{|\operatorname{sgn}\bar{Y}_{\pi,i} - \operatorname{sgn}\phi(\boldsymbol{X}_i)|}_{\text{classification loss}}, \tag{6}$$

where $\phi \in \Phi(r)$ is the trace function to be optimized, and $|\bar{Y}_{\pi,i}|$ serves as the weight. Such a response-specific weight incorporates the magnitude information of the response into classification, in that the response values that are far away from the target level are penalized more heavily in the objective (6). In the special case of a binary response $Y_i \in \{-1,1\}$ and target level $\pi = 0$, the objective (6) reduces to the usual classification loss.

Next, define the weighted classification risk,

$$\operatorname{Risk}_\pi(\phi) = \mathbb{E}L(\phi; (\boldsymbol{X}_i, \bar{Y}_{\pi,i})_{i\in[n]}), \tag{7}$$

where the expectation is taken with respect to the joint distribution of $(\boldsymbol{X}_i, Y_i)$ i.i.d. from $\mathbb{P}_{\boldsymbol{X},Y}$. The next theorem quantifies the global optimum of (7).

THEOREM 1 (Global optimum of weighted classification risk). *For any given level $\pi \in [-1,1]$, under the model specified in (2) and (3), for all functions $\bar{f}$ that have the same sign as $\operatorname{sgn}(f-\pi)$, it holds that $\operatorname{Risk}_\pi(\bar{f}) = \inf\{\operatorname{Risk}_\pi(\phi)\colon \phi \in \Phi(r)\}$.*

Theorem 1 suggests a practical procedure to estimate $\operatorname{sgn}(f-\pi)$ through weighted classifications. The result ensures that the sign function $\operatorname{sgn}(f-\pi)$ minimizes the weighted classification risk. The inverse, however, may not hold true, due to possible multiple global optimizers of $\operatorname{Risk}_\pi(\cdot)$. A simple example is a constant regression $f(\boldsymbol{X}) = \mathbb{E}(Y|\boldsymbol{X}) = c$, in which case, every function $\phi \in \Phi(r)$ minimizes $\operatorname{Risk}_\pi(\cdot)$ at the level $\pi = c$. The next section resolves this issue by characterizing the uniqueness of the risk optimizer.

3.2. *Identifiability.* To establish the statistical guarantee of the minimizer of $\operatorname{Risk}(\cdot)$, we first address its uniqueness, up to some sign equivalence. It turns out the local behavior of the regression function f around π plays a key role to establish the identifiability of sign function series from weighted classifications.

We introduce some additional notation. We call $S_{\text{bayes}}(\pi) = \{\boldsymbol{X} \in \mathcal{X}\colon f(\boldsymbol{X}) \geq \pi\}$ the Bayes set at level π, and $\partial S_{\text{bayes}}(\pi) = \{\boldsymbol{X} \in \mathcal{X}\colon f(\boldsymbol{X}) = \pi\}$ the level set boundary. Note that there is a one-to-one correspondence between the sign function $\operatorname{sgn}(f-\pi)$ and the Bayes set $S_{\text{bayes}}(\pi)$. We choose to present the results in terms of $S_{\text{bayes}}(\pi)$ for easier comparison with the existing classification literature [33, 31]. We call a level $\pi \in [0,1]$ a mass point if the level set boundary $\partial S_{\text{bayes}}(\pi)$ has a non-zero measure under $\mathbb{P}_{\boldsymbol{X}}$. Let $\mathcal{N} = \{\pi \in [-1,1]\colon \mathbb{P}_{\boldsymbol{X}}[f(\boldsymbol{X}) = \pi] \neq 0\}$ denote the collection of all mass points in f. Assume there exists a constant $c > 0$, independent of the feature space dimension, such that $|\mathcal{N}| \leq c < \infty$. We introduce a notion of smoothness for the cumulative distribution function (CDF) of $f(\boldsymbol{X})$ under measure $\mathbb{P}_{\boldsymbol{X}}$.

DEFINITION 2 (α-smoothness). Suppose $\mathbb{P}_{\boldsymbol{X}}$ is a continuous distribution, and denote the CDF $G(\pi) = \mathbb{P}_{\boldsymbol{X}}[f(\boldsymbol{X}) \leq \pi]$. A function f is called (α,π)-locally smooth, for a given $\pi \notin \mathcal{N}$, if there exist constants $C = C(\pi) > 0$ and $\alpha = \alpha(\pi) \geq 0$, such that

$$\sup_{0\leq t<\rho(\pi,\mathcal{N})} \frac{G(\pi+t) - G(\pi-t)}{t^\alpha} \leq C, \tag{8}$$

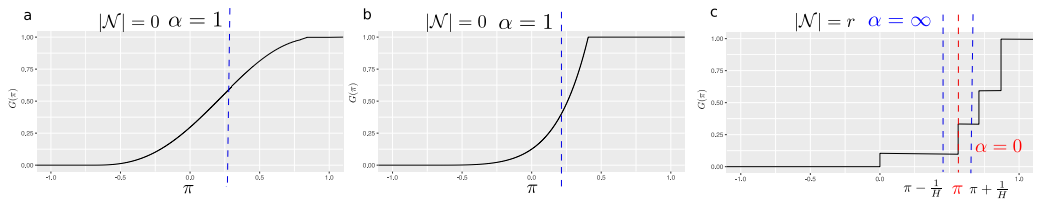

FIG 3. *Three examples of CDF, $G(\pi) = \mathbb{P}_{\boldsymbol{X}}(f(\boldsymbol{X}) \le \pi)$, with local smoothness index α at π depicted in dashed line. (a) and (b). Function $G(\pi)$ $\alpha = 1$ because the $G(\pi)$ has finite sub-derivatives in the range of π; (c). Function $G(\pi)$ with $\alpha = \infty$ at most π (in blue), except for a total number of $|\mathcal{N}| = r$ jump points (in red). Here $|\mathcal{N}|$ denotes the number of jump points.*

where $\rho(\pi, \mathcal{N}) = \min_{\pi' \in \mathcal{N}} |\pi - \pi'|$ denotes the distance from π to the nearest point in $\mathcal{N}$. We make the convention that $\rho(\pi, \mathcal{N}) = 2$ (which equals the range of $\pi \in [-1, 1]$) when $\mathcal{N}$ is empty, and $\alpha = \infty$ when the numerator in (8) is zero. The largest possible $\alpha = \alpha(\pi)$ in (8) is called the smoothness index at level π. The function f is called α-globally smooth, if (8) holds with a global constant C for all $\pi \in [-1, 1]$ except for a finite number of levels.

Fig 3 shows three examples of the CDF with various levels of smoothness. A small value of $\alpha < 1$ indicates the infinite density at level π, or equivalently, when $G(\pi)$ jumps at π. A large value of $\alpha > 1$ corresponds to the case of no point mass around π, or equivalently, when $G(\pi)$ remains flat. An intermediate case is $\alpha = 1$ when $G(\pi)$ has a finite non-zero sub-derivative in the vicinity of π. The global smoothness index is the minimal α over all π's; meanwhile, we allow exceptions for a finite number of levels.

Next, we show that the α-smoothness with $\alpha \neq 0$ implies the uniqueness of $S_{\text{bayes}}(\pi)$ for the optimizer of $\text{Risk}(\cdot)$. For two sets $S_1, S_2 \in \mathcal{X}$, define the probabilistic set difference,

$$d_\Delta(S_1, S_2) = \mathbb{P}_{\boldsymbol{X}}(S_1 \Delta S_2) = \mathbb{P}_{\boldsymbol{X}}\{\boldsymbol{X} : \boldsymbol{X} \in S_1 \setminus S_2 \text{ or } S_2 \setminus S_1\},$$

and the risk difference,

$$d_\pi(S_1, S_2) = \text{Risk}_\pi(\text{sgn}(S_1)) - \text{Risk}_\pi(\text{sgn}(S_2)).$$

THEOREM 2 (Identifiability). *Suppose f is α-globally smooth over $\mathcal{X}$. Then,*

$$d_\Delta(S, S_{\text{bayes}}(\pi)) \lesssim [d_\pi(S, S_{\text{bayes}}(\pi))]^{\frac{\alpha}{1+\alpha}} + \frac{1}{\rho(\pi, \mathcal{N})} d_\pi(S, S_{\text{bayes}}(\pi)), \tag{9}$$

for all sets $S \in \mathcal{X}$ and all levels $\pi \in [-1, 1]$ except for a finite number of levels.

We make two remarks. First, the bound (9) controls the worst-case perturbation of the classifiers under the measure $\mathbb{P}_{\boldsymbol{X}}$ with respect to the weighted classification risks. When $\alpha \neq 0$, the inequality (9) immediately implies the uniqueness, up to a measure-zero set in $\mathbb{P}_{\boldsymbol{X}}$, of $S_{\text{bayes}}(\pi)$ in minimizing $\text{Risk}_\pi(\cdot)$. Second, our identifiability improves the earlier results for a single level set estimation to multiple level set estimations. Existing work [31, 39] considered only a finite number of π's, and provided only the first term in the bound (9). In contrast, our bound quantifies the full dependence on the level π, and establishes the recovery condition of $S_{\text{bayes}}(\pi)$ uniformly over all possible π's. It turns out both terms in the bound (9) are crucial for our regression function estimation. The first term contributes to the classification error, and the second term contributes to the variance in sign series aggregations.

3.3. *Regression risk bound.* In this section, we provide the statistical accuracy guarantee for the learning reduction based estimators (4) and (5). Our theory consists of three main ingredients. We first leverage the α-smoothness to provide a sharp rate for $\hat{\phi}_\pi$'s classification risk faster than the usual root-n convergence. The improvement stems from the fact that, under the given assumptions, the variance of the excess classification loss is bounded in terms of its expectation. Because the variance decreases as we approach the optimal $\operatorname{sgn}(f-\pi)$, the risk of $\hat{\phi}_\pi$ converges more quickly to the optimal risk than the simple uniform converge results would suggest. The second step is to convert the risk error into the probability set error by Theorem 2. The last step is to aggregate the set error into the final nonparametric function estimation. A careful error analysis reveals the joint contribution from both sign aggregations and variance-bias trade-off.

The next result establishes the estimation accuracy for sign function estimator (4).

THEOREM 3 (Sign function estimation). *Suppose the regression function $f \in \mathcal{F}_{\text{sgn}}(r)$ is α-globally smooth over $\mathcal{X}$, and let $d_{\max} = \max(d_1, d_2)$. Then, for all $\pi \in [-1,1]$ except for a finite number of levels, with high probability at least $1-\exp(-rd_{\max})$ over the training data $(\boldsymbol{X}_i, Y_i)_{i\in[n]}$, we have,*

$$\|\operatorname{sgn}\hat{\phi}_\pi - \operatorname{sgn}(f-\pi)\|_1 \lesssim \left(\frac{rd_{\max}}{n}\right)^{\frac{\alpha}{2+\alpha}} + \frac{1}{\rho^2(\pi,\mathcal{N})}\left(\frac{rd_{\max}}{n}\right), \tag{10}$$

where the L_1 norm is taken with respect to the measure $\boldsymbol{X} \sim \mathbb{P}_{\boldsymbol{X}}$.

Theorem 3 quantifies the statistical convergence of the sign function estimation. For a fixed π, the second term in (10) is absorbed into the first term, leading to the rate $O(n^{-\alpha/(2+\alpha)})$. We find that the sign estimation reaches a fast rate $1/n$ when $\alpha = \infty$, and reaches a slow rate $1/\sqrt{n}$ when the point mass concentrates with $\alpha = 0$. This is consistent with our intuition, because best rate $\alpha = \infty$ corresponds to a clear separation with no point mass at the Bayes set boundary $\partial S_{\text{bayes}}(\pi)$, whereas the worst rate $\alpha = 0$ corresponds to a heavy mass around $\partial S_{\text{bayes}}(\pi)$. Furthermore, the sign function estimation achieves consistency in the high dimensional region as long as $n \gg d_{\max} \to \infty$ and $\alpha \neq 0$. Combining the sign representability of the regression function and the uniform sign estimation accuracy, we obtain our main theoretical result on the nonparametric trace regression.

THEOREM 4 (Regression function estimation). *Suppose the same conditions in Theorem 3 hold. With high probability at least $1-\exp(-rd_{\max})$ over the training data $(\boldsymbol{X}_i, Y_i)_{i\in[n]}$, we have*

$$\|\hat{f} - f\|_1 \lesssim \underbrace{\left(\frac{rd_{\max}\log H}{n}\right)^{\frac{\alpha}{2+\alpha}}}_{\text{estimation error from sign functions}} + \underbrace{\frac{1}{H}}_{\text{reduction bias}} + \underbrace{\left(\frac{rd_{\max}}{n}\right) H\log H}_{\text{reduction variance}}, \tag{11}$$

for any resolution parameter $H \in \mathbb{N}_+$. In particular, setting $H \asymp \left(\frac{n}{rd_{\max}}\right)^{1/2}$ gives

$$\|\hat{f} - f\|_1 \lesssim \left(\frac{rd_{\max}\log n}{n}\right)^{\min\left(\frac{\alpha}{2+\alpha}, \frac{1}{2}\right)}, \tag{12}$$

where the L_1 norm is taken with respect to the measure $\boldsymbol{X} \sim \mathbb{P}_{\boldsymbol{X}}$

Theorem 4 establishes the convergence rate of the proposed learning reduction estimator for the nonparametric trace regression. We make three remarks. First, the bound (11) reveals

three sources of errors: the estimation error from sign functions, the bias due to sign series representations, and the variance thereof. Recall that H determines the number of sign functions in sign series representations. It controls the bias-variance tradeoff here. Second, the regression is robust to a few off-target classifications, as long as the majorities are accurate. This can also be seen in Fig 3(a) where the classification is nonidentifiable at some mass point (red line). Nevertheless, the regression estimation is still possible because the nearby classifications provide the sign signal (blue lines). This fact shows the benefit of sign aggregations, and also explains the trade-off in choosing H. Intuitively, a larger value of H increases the approximation accuracy, but meanwhile renders the classification harder near the mass points. Third, the final regression error is generally no better than the sign error, as we compare the bounds in (12) with (10). This confirms our premise that classification is easier than regression. On the other hand, our sign representation approach allows us to disentangle the complexity and achieve the theoretical guarantee from classification to regression.

4. Two applications of nonparametric matrix learning. In this section, we apply the general theory in Theorem 4 to two specific nonparametric matrix learning problems, the low-rank sparse matrix predictor regression, and the high-rank matrix completion.

4.1. *Low-rank sparse matrix predictor regression.* The first problem we consider is matrix predictor regression. In addition to the low sign rank structure, we also introduce a two-way sparsity structure. That is, we impose that some rows and columns of $\boldsymbol{B}$ are zeros, where $\boldsymbol{B}$ is as defined in (3). We comment that sparsity is a commonly used structure in matrix data modeling [44], and it is scientifically plausible in numerous applications [42].

Specifically, we extend the notation $\Phi(r)$ and $\mathcal{F}_{\text{sgn}}(r)$ introduced in Definition 1 to incorporate the sparsity. Let $\Phi(r, s_1, s_2)$ denote the collection of trace functions,

$$\Phi(r, s_1, s_2) = \{\phi\colon \boldsymbol{X} \mapsto \langle \boldsymbol{X}, \boldsymbol{B}\rangle + b \,\big|\, \text{rank}(\boldsymbol{B}) \le r, \text{supp}(\boldsymbol{B}) \le (s_1, s_2), (\boldsymbol{B}, b) \in \mathbb{R}^{d_1\times d_2} \times \mathbb{R}\},$$

where $\text{supp}(\boldsymbol{B})$ denotes the support of $\boldsymbol{B}$, with the sparsity parameters, $s_1 = \|\boldsymbol{B}\|_{1,0} = |\{i \in [d_1]\colon \boldsymbol{B}_i \neq \boldsymbol{0}\}|$, and $s_2 = \|\boldsymbol{B}^T\|_{1,0} = |\{j \in [d_2]\colon \boldsymbol{B}_j^T \neq \boldsymbol{0}\}|$, denoting the number of non-zero rows and non-zero columns of $\boldsymbol{B}$, respectively. Similarly, let $\mathcal{F}_{\text{sgn}}(r, s_1, s_2)$ denote a family of rank-r, support-(s_1, s_2) sign representable functions based on (3). We have the following result.

THEOREM 5 (Nonparametric low-rank two-way sparse regression). *Consider the same setup as in Theorem 4, except that we replace $\mathcal{F}_{\text{sgn}}(r)$ and $\Phi(r)$ with $\mathcal{F}_{\text{sgn}}(r, s_1, s_2)$ and $\Phi(r, s_1, s_2)$, respectively. Set $H \asymp \left(\frac{n}{r(s_1+s_2)\log d_{\max}}\right)^{1/2}$ in (5). With high probability at least $1 - d_{\max}^{-r(s_1+s_2)}$ over the training data $(\boldsymbol{X}_i, Y_i)_{i\in[n]}$, the estimate (5) is bounded by*

$$\|\hat{f} - f\|_1 \lesssim \left(\frac{r(s_1+s_2)\log d_{\max}\log n}{n}\right)^{\min\left(\frac{\alpha}{2+\alpha}, \frac{1}{2}\right)}. \tag{13}$$

We make two remarks. First, the bound (13) suggests that the estimator remains consistent in the high dimensional regime as $d_{\max}$ and $n \to \infty$, as long as $d_{\max}$ grows sub-exponentially in the sample size n. Such a sample complexity shows the pronounced advantage of the low-rank two-way sparse structural model, by comparing (13) and (12). Second, the two-way sparsity structure facilitates the interpretability, which we further demonstrate through numerical examples in Section 6.2.

4.2. *High-rank matrix completion.* The second problem we consider is matrix completion. Let $\boldsymbol{Y} \in \mathbb{R}^{d_1 \times d_2}$ be a data matrix generated from the model,

$$\boldsymbol{Y} = \boldsymbol{\Theta} + \boldsymbol{E}, \tag{14}$$

where $\boldsymbol{\Theta} \in \mathcal{M}_{\text{sgn}}(r)$ denotes an unknown signal matrix, and $\boldsymbol{E}$ is an error matrix consisting of zero-mean, independent but not necessarily identically distributed entries. For simplicity, we assume $d_1 = d_2 = d$. Model (14) can be viewed as a special case of model (2), where the predictor space consists of the basis matrices in $\mathbb{R}^{d\times d}$, and the data matrix $\boldsymbol{Y} = [\![Y_{ij}]\!]$ collects the scalar response $Y_{ij} \in \mathbb{R}$. In this case, the problem of regression estimation becomes the estimation of $\boldsymbol{\Theta}$. What is observed is an incomplete data matrix $\boldsymbol{Y}_\Omega$ from (14), where $\Omega \subset [d]^2$ represents the index set of the observed entries. We allow both uniform and non-uniform sampling schemes for Ω. Let $\Pi = \{p_\omega\}$ be an arbitrarily predefined probability distribution over the full index set with $\sum_{\omega\in[d]^2} p_\omega = 1$. Assume the entries ω in Ω are i.i.d. draws with replacement from the full index set following the distribution Π. Denote the sampling rule as $\omega \sim \Pi$, and $\boldsymbol{Y}(\omega)$ the matrix entry indexed by ω.

Now applying our learning reduction approach to the matrix completion problem (14) yields the signal matrix estimate

$$\hat{\boldsymbol{\Theta}} = \frac{1}{2H+1} \sum_{\pi\in\mathcal{H}} \text{sgn}(\hat{\boldsymbol{Z}}_\pi), \tag{15}$$

where, for every $\pi \in \{-1, \ldots, -1/H, 0, 1/H, \ldots, 1\}$, the matrix $\hat{\boldsymbol{Z}}_\pi$ is the solution to the weighted classification

$$\hat{\boldsymbol{Z}}_\pi = \underset{\boldsymbol{Z}:\ \text{rank}(\boldsymbol{Z})\le r}{\arg\min} \sum_{\omega\in\Omega} \underbrace{|\boldsymbol{Y}(\omega) - \pi|}_{\text{weight}} \underbrace{|\text{sgn}(\boldsymbol{Y}(\omega) - \pi) - \text{sgn}(\boldsymbol{Z}(\omega))|}_{\text{classification loss}}.$$

To assess the accuracy of the estimate $\hat{\boldsymbol{\Theta}} = \hat{\boldsymbol{\Theta}}_{d\times d}$ in the high dimensional regime $d \to \infty$, we need to put the model in the nonparametric context of Definition 2. We next extend the notion of α-smoothness to a discrete feature space as follows. Let $\Delta s = 1/d^2$ denote a small tolerance, where d^2 represents the number of elements in the feature space. We quantify the distribution of the entries in matrix $\boldsymbol{\Theta}$ using a pseudo density, i.e., histogram with bin width $2\Delta s$. Specifically, let $G(\pi) = \mathbb{P}_{\omega\sim\Pi}[\boldsymbol{\Theta}(\omega) \le \pi]$ denote the CDF of $\boldsymbol{\Theta}(\omega)$ under $\omega \sim \Pi$. We partition $[-1, 1] = \mathcal{N} \cup \mathcal{N}^c$, where $\mathcal{N}$ consists of levels whose pseudo density based on $2\Delta s$-bin is asymptotically unbounded; i.e,

$$\mathcal{N} = \left\{ \pi \in [-1, 1]\colon \frac{G(\pi + \Delta s) - G(\pi - \Delta s)}{\Delta s} \ge c_1 \right\}, \text{ for some universal constant } c_1 > 0,$$

and $\mathcal{N}^c$ otherwise. Let $|\mathcal{N}|_{\text{cover}}$ be the covering number of $\mathcal{N}$ with $2\Delta s$-bin's; i.e, $|\mathcal{N}|_{\text{cover}} = \text{Leb}(\mathcal{N})/2\Delta s$, where $\text{Leb}(\cdot)$ denotes the Lebesgue measure. The following assumption is a discrete analogy of Definition 2.

DEFINITION 3 (α-smoothness for discrete distribution). Let Π be the sampling distribution over $[d^2]$. We say the signal matrix $\boldsymbol{\Theta}(\omega)$ is α-globally smooth under $\omega \sim \Pi$, if there exist constants $c_2, c_3 > 0$, such that $|\mathcal{N}|_{\text{cover}} \le c_2$, and for all $\pi \in \mathcal{N}^c$,

$$\sup_{\Delta s \le t < \rho(\pi,\mathcal{N})} \frac{G(\pi + t) - G(\pi - t)}{t^\alpha} \le c_3, \quad \text{with } \rho(\pi, \mathcal{N}) = \min_{\pi'\in\mathcal{N}} |\pi - \pi'| + \Delta s$$

and $\rho(\pi, \mathcal{N})$ denotes the adjusted distance from π to the nearest point in $\mathcal{N}$.

We assess the estimation error of (15) using the mean absolute error (MAE), $\mathrm{MAE}(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta}) = \mathbb{E}|\hat{\boldsymbol{\Theta}}(\omega) - \boldsymbol{\Theta}(\omega)|$, where the expectation is with respect to a future observation $\boldsymbol{\Theta}(\omega)$ from the distribution G. We have the following result.

THEOREM 6 (Nonparametric matrix completion). *Consider the matrix model* (14) *with* α*-smooth signal matrix* $\boldsymbol{\Theta} \in \mathcal{M}_{\mathrm{sgn}}(r)$. *Set* $H \asymp \left(\frac{|\Omega|}{dr}\right)^{1/2}$. *With high probability at least* $1 - \exp(-dr)$ *over* $\boldsymbol{Y}_\Omega$, *the estimate* (15) *satisfies that*

$$\mathrm{MAE}(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta}) \lesssim \left(\frac{dr \log |\Omega|}{|\Omega|}\right)^{\min(\frac{\alpha}{2+\alpha}, \frac{1}{2})}. \tag{16}$$

We remark that our estimation accuracy (16) applies to both low-rank and high-rank signal matrices. Moreover, the estimation rate depends on the sign complexity $\boldsymbol{\Theta} \in \mathcal{M}_{\mathrm{sgn}}(r)$, where r can be much smaller than the usual matrix rank as shown in Proposition 2. In fact, our theorem can also be relaxed for a growing $|\mathcal{N}|_{\mathrm{cover}}$ as a function of d, with a slight modification on the setup; see Appendix A.3 for such an extension. We next illustrate Theorem 6 with two matrix completion examples and compare with the existing literature.

EXAMPLE 6 (Stochastic block model based matrix completion). The stochastic block model [6] assumes a checkerboard structure under marginal row and column permutations. The signal matrix belongs to our sign representable family $\boldsymbol{\Theta} \in \mathcal{M}_{\mathrm{sgn}}(r)$, where r is the number of blocks. Besides, the block matrix is ∞-globally smooth, because $\mathcal{N}$ consists of finitely many $2\Delta s$-bin's covering the block means. Our signal estimate achieves the rate $\tilde{\mathcal{O}}(d^{-1/2})$ when $\alpha = \infty$ with no missingness. This rate agrees with the minimax root-mean-square error (RMSE) rate for stochastic block models with a fixed number of blocks [14].

EXAMPLE 7 (Single index model based matrix completion). The single index model based completion [13] admits a signal matrix $\boldsymbol{\Theta} = g(\boldsymbol{B})$, where g is an unknown monotonic function, and $\boldsymbol{B}$ is an unknown low-rank matrix. Note that $\boldsymbol{\Theta}$ itself is often of a high matrix rank as shown in Fig 1(a). Suppose the CDF of $\boldsymbol{\Theta}(\omega)$ has a bounded pseudo density with $\alpha = 1$. Applying Theorem 6 yields the estimation error rate $\tilde{\mathcal{O}}(d^{-1/3})$, which is faster compared to the RMSE rate $\tilde{\mathcal{O}}(d^{-1/4})$ obtained earlier [13].

Finally, we obtain the sample complexity of the nonparametric matrix completion, summarized in the next corollary.

COROLLARY 1 (Sample complexity for nonparametric completion). Suppose the same conditions of Theorem 6 hold. When $\alpha \neq 0$, with high probability at least $1 - \exp(-dr)$ over $\mathcal{Y}_\Omega$,

$$\mathrm{MAE}(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta}) \to 0, \quad \text{as} \quad \frac{|\Omega|}{dr \log |\Omega|} \to \infty.$$

Corollary 1 improves the earlier work [41, 22] by allowing both low-rank and high-rank signals. Moreover, the sample size requirement depends only on the sign complexity $\tilde{\mathcal{O}}(dr)$, but not the nonparametric complexity α. We also note that $\tilde{\mathcal{O}}(dr)$ roughly matches the degree of freedom of the signals, suggesting the optimality of our sample requirements.

5. Large-margin implementation and ADMM algorithm. In Section 3, we have established the methodology and theory for the nonparametric matrix trace regression under the 0-1 loss, since this is the canonical loss for classification. However, this loss may be difficult to optimize in practice. In this section, we extend it with a continuous large-margin loss, and present the corresponding optimization algorithm. We consider two loss functions: the hinge loss $F(z) = (1-z)_+$ for support vector machines, and the psi-loss $F(z) = 2\min(1, (1-z)_+)$ with $z_+ = \max(z, 0)$ [30]. These two losses are most commonly used in classification, and both satisfy the linear excess risk bound; see Section 5.4. We focus on the nonparametric low-rank sparse matrix regression problem. With some straightforward modification, the solution applies to matrix completion and other matrix learning problems as well.

5.1. *Large-margin learning.* Specifically, we generalize the 0-1 loss minimization (6) to the following continuous large-margin loss minimization problem,

$$\hat{\phi}_{\pi,F} = \mathop{\arg\min}_{\phi \in \Phi(r,s_1,s_2)} \left\{ \frac{1}{n} \sum_{i=1}^{n} |Y_i - \pi| F(\phi(\boldsymbol{X}_i) \mathrm{sgn}(Y_i - \pi)) + \lambda \|\phi\|_F^2 \right\}, \tag{17}$$

where $F(z)\colon \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$ is a continuous function of the margin $z = y\phi(\boldsymbol{X})$, $\lambda > 0$ is the penalty parameter, and $\|\phi\|_F$ is the penalty function. We set $\|\phi\|_F = \|\boldsymbol{B}\|_F$, with $\boldsymbol{B}$ being the coefficient matrix associated with $\phi \in \Phi(r, s_1, s_2)$. The use of large-margin loss in (17) allows us to leverage efficient large-margin optimization algorithms, while maintaining desirable statistical properties under mild conditions. The benefit of ridge penalization has been studied [30]. We obtain the corresponding regression function estimate as,

$$\hat{f}_F = \frac{1}{2H+1} \sum_{\pi \in \mathcal{H}} \mathrm{sgn} \hat{\phi}_{\pi,F}. \tag{18}$$

5.2. *ADMM optimization.* We next present an algorithm to solve (17) for a given $\pi \in \mathcal{H}$. We first note that the estimation problem (17) is equivalent to the optimization,

$$\min_{(\boldsymbol{B},b)\colon \mathrm{rank}(\boldsymbol{B}) \leq r, \mathrm{supp}(\boldsymbol{B}) \leq (s_1,s_2)} \frac{1}{n} \sum_{i=1}^{n} |\bar{Y}_{\pi,i}| F\big([\langle \boldsymbol{X}_i, \boldsymbol{B} \rangle + b] \mathrm{sgn} \bar{Y}_{\pi,i}\big) + \lambda \|\boldsymbol{B}\|_F^2, \tag{19}$$

where we recall $\bar{Y}_{\pi,i} = Y_i - \pi$ is the π-shifted response. The loss function F can be convex, e.g., hinge loss, or non-convex, e.g., psi-loss. Meanwhile, the optimization (19) has a non-convex feasible region because of the low-rank and sparsity constraints.

We propose an alternating direction method of multipliers (ADMM) algorithm to solve (19). We introduce a dual variable and an additional feasibility constraint to perform coordinate descent in the augmented Lagrangian function. The augmented objective of (19) is

$$L(\boldsymbol{B}, b, \boldsymbol{S}, \boldsymbol{\Lambda}, \rho) = \frac{1}{n} \sum_{i=1}^{n} |\bar{Y}_{\pi,i}| F\big([\langle \boldsymbol{X}_i, \boldsymbol{B} \rangle + b] \mathrm{sgn} \bar{Y}_{\pi,i}\big) + \lambda \|\boldsymbol{B}\|_F^2 + \rho \|\boldsymbol{B} - \boldsymbol{S}\|_F^2 + \langle \boldsymbol{\Lambda}, \boldsymbol{B} - \boldsymbol{S} \rangle,$$

where $\boldsymbol{B} \in \mathbb{R}^{d_1 \times d_2}$ is the unconstrained primal variable, $\boldsymbol{S} \in \mathbb{R}^{d_1 \times d_2}$ is the constrained dual variable satisfying $\mathrm{rank}(\boldsymbol{S}) \leq r$ and $\mathrm{supp}(\boldsymbol{S}) \leq (s_1, s_2)$, $\boldsymbol{\Lambda} \in \mathbb{R}^{d_1 \times d_2}$ is the Lagrangian multiplier, and $\rho > 0$ is the step size parameter. Note that in $L(\boldsymbol{B}, b, \boldsymbol{S}, \boldsymbol{\Lambda}, \rho)$, the non-convexity has moved from the first two terms in $\boldsymbol{B}$ to the last two simpler terms in $\boldsymbol{S}$. This separability simplifies the optimization for a wide range of loss functions and constraints.

We next minimize $L(\boldsymbol{B}, b, \boldsymbol{S}, \boldsymbol{\Lambda}, \rho)$ via coordinate descent, by iteratively updating one variable at a time while holding others fixed. Each update reduces to a simpler problem and can be efficiently solved by standard algorithms.

Algorithm 1 Nonparametric low-rank two-way sparse matrix regression via ADMM

Input: data $(\boldsymbol{X}_i, Y_{\pi,i})_{i\in[n]}$, rank r, support (s_1, s_2), ridge parameter λ, resolution parameter H.

for $\pi \in \mathcal{H} = \{-1, \ldots, -\frac{1}{H}, 0, \frac{1}{H}, \ldots, 1\}$ **do**
  initialize dual variable $\boldsymbol{S}$ randomly, Lagrangian multiplier $\boldsymbol{\Lambda} = \boldsymbol{0}$, step size $\rho = 1$, and $\bar{Y}_{\pi,i}$.
  repeat
    update $(\boldsymbol{B}, b) \leftarrow \arg\min L(\boldsymbol{B}, b|\boldsymbol{S}, \boldsymbol{\Lambda}, \rho)$.
    update $\boldsymbol{S} \leftarrow \arg\min \|\boldsymbol{S} - \frac{1}{2\rho}(2\rho\boldsymbol{B} + \boldsymbol{\Lambda})\|_F^2$ subject to $\text{rank}(\boldsymbol{S}) \le r$ and $\text{supp}(\boldsymbol{S}) \le (s_1, s_2)$.
    update $\boldsymbol{\Lambda} \leftarrow \boldsymbol{\Lambda} + 2\rho(\boldsymbol{B} - \boldsymbol{S})$.
    update $\rho \leftarrow 1.1\rho$.
  until convergence
  return trace function estimate, $\hat{\phi}_\pi : \boldsymbol{X} \mapsto \langle \hat{\boldsymbol{B}}, \boldsymbol{X}\rangle + \hat{b}$.
end for

Output: nonparametric regression function estimate, $\hat{f} = \frac{1}{2H+1}\sum_{\pi\in\mathcal{H}} \text{sgn}\hat{\phi}_\pi$.

Specifically, given variables $(\boldsymbol{S}, \boldsymbol{\Lambda}, \rho)$ and $\bar{\boldsymbol{S}} = (2\rho\boldsymbol{S} - \boldsymbol{\Lambda})/[2(\rho + \lambda)]$, the objective with respect to $(\boldsymbol{B}, b)$ is

$$L(\boldsymbol{B}, b|\boldsymbol{S}, \boldsymbol{\Lambda}, \rho) = \frac{1}{n}\sum_{i=1}^{n} |\bar{Y}_{\pi,i}| F\big([\langle \boldsymbol{X}_i, \boldsymbol{B}\rangle + b]\text{sgn}\bar{Y}_{\pi,i}\big) + (\lambda + \rho)\|\boldsymbol{B} - \bar{\boldsymbol{S}}\|_F^2.$$

Optimization with (5.2) is a standard vector based classification problem with a ridge penalty and an offset $\bar{\boldsymbol{S}}$. There are a number of state-of-art algorithms for weighted SVM [35] and psi-learning [30], which are readily available to solve this problem.

Next, given $(\boldsymbol{B}, b, \boldsymbol{\Lambda}, \rho)$, and $\bar{\boldsymbol{B}} = (2\rho\boldsymbol{B} + \boldsymbol{\Lambda})/(2\rho)$, the objective with respect to $\boldsymbol{S}$ is

$$L(\boldsymbol{S}|\boldsymbol{B}, b, \boldsymbol{\Lambda}, \rho) = \|\boldsymbol{S} - \bar{\boldsymbol{B}}\|_F^2, \quad \text{subject to } \text{rank}(\boldsymbol{S}) \le r \text{ and } \text{supp}(\boldsymbol{S}) \le (s_1, s_2). \tag{20}$$

This is equivalent to the best sparse low-rank approximation, in the least-square sense, to the matrix $\boldsymbol{B}$. Compared to the original objective (19), the least-square objective is easier to handle. A number of learning algorithms have been designed to solve this problem, e.g., sparse PCA, sparse SVD, and projection pursuit [23]. We adopt the recently developed double projection method, which has a competitive performance in the high dimensional regime [40].

Finally, the Lagrangian multiplier $\boldsymbol{\Lambda}$ is updated by $\boldsymbol{\Lambda} \leftarrow \boldsymbol{\Lambda} + 2\rho(\boldsymbol{B} - \boldsymbol{S})$. Following some common practice in matrix non-convex optimization [40], we run the optimization from multiple initializations to locate a final estimate with the lowest objective value. We summarize the above optimization procedure in Algorithm 1.

5.3. *Hyperparameter tuning.* We briefly describe the hyperparameters in Algorithm 1 and discuss their choices in practice. There are two sets of hyperparameters, one set for model specification, and the other for algorithmic stability. The model hyperparameters are (r, s_1, s_2), which determine the complexity of sign functions. We choose (r, s_1, s_2) via a grid search based on cross-validation regression error. The resolution in grid search depends on the problem size; for instance, in our brain connectivity data example with $d_1 = d_2 = 68$ in Section 7.1, we search for the optimal values of r, s_1, s_2 over $[d]$, with an increment of 5, under the natural constraint $r \le s_1 = s_2$. The algorithm hyperparameters are (H, λ, ρ). For H and λ, their optimal choices are given in Theorems 4 and 7, respectively. In practice, we default $H = \min(20, \sqrt{n})$, and $\lambda = \min(0.1, n^{-1})$, which perform well in our numerical experiments. For the step size ρ that controls the closeness between the dual and primal variables, we initialize from 1, and increase its value geometrically by 1.1 during the iterations until the relative change in the primal residual $\|\boldsymbol{B} - \bar{\boldsymbol{S}}\|_F$ falls below a threshold [25]. In our numerical analyses, we observe this scheme provides a stable optimization trajectory.

5.4. *Large-margin statistical guarantees.* We next establish the statistical accuracy for the large-margin estimators under some additional technical assumptions. Let $f_{\text{bayes},\pi} = \text{sgn}(f-\pi)$ denote the ground truth sign function at $\pi \in [-1,1]$, and let

$$\begin{aligned} \text{Risk}(\phi) &= \frac{1}{2}\mathbb{E}|Y-\pi|\big|\text{sgn}(Y-\pi) - \text{sgn}\phi(\boldsymbol{X})\big|, \\ \text{Risk}_{\pi,F}(\phi) &= \mathbb{E}|Y-\pi|F\big(\phi(\boldsymbol{X})\text{sgn}(Y-\pi)\big), \end{aligned} \tag{21}$$

denote the 0-1 risk and F-risk, respectively, where F is the surrogate continuous loss, and the expectation is taken with respect to $(\boldsymbol{X},Y) \sim \mathbb{P}_{\boldsymbol{X},Y}$ following the regression model $f \in \mathcal{F}_{\text{sgn}}(r,s_1,s_2)$. For simplicity, we assume $d_1 = d_2 = d$ and $\|\boldsymbol{X}\|_F \le 1$ with probability 1. We consider the high dimensional regime where both n and d grow, while (r,s_1,s_2) remain fixed. We need the following assumptions.

ASSUMPTION 1 (Assumptions on surrogate loss).

(a) (Approximation error) For any given $\pi \in [-1,1]$, assume there exist a sequence of functions $\phi_\pi^{(n)} \in \Phi(r,s_1,s_2)$, such that $\text{Risk}_{\pi,F}(\phi_\pi^{(n)}) - \text{Risk}_{\pi,F}(f_{\text{bayes},\pi}) \le a_n$, for some sequence $a_n \to 0$ as $n \to \infty$. Furthermore, assume $\|\phi_\pi^{(n)}\|_F \le J$ for some constant $J > 0$.
(b) (Common loss) $F(z) = (1-z)_+$ is hinge loss, or $F(z) = 2\min(1,(1-z)_+)$ is psi-loss.

Assumption 1(a) quantifies the representation capability of F and $\phi(r,s1,s2)$. We note that, although the Bayes rule $f_{\text{bayes},\pi}$ also depends on n implicitly through $d = d(n)$, we drop the dependence on n for simpler notation. Assumption 1(b) implies the Fisher consistency bound for the weighted risk [29],

$$\text{Risk}_\pi(\phi) - \text{Risk}_\pi(f_{\text{bayes},\pi}) \le C[\text{Risk}_{\pi,F}(\phi) - \text{Risk}_{\pi,F}(f_{\text{bayes},\pi})], \text{ for all } \pi \in [-1,1] \text{ and all } \phi.$$

where $C = 1$ for the 0-1 or the hinge loss, and $C = 1/2$ for the psi-loss; see Lemma 2 in Appendix. Therefore, it suffices to bound the excess F-risk in order to bound the usual 0-1 risk. Under Assumption 1, we establish the estimation accuracy guarantee for the large-margin estimators (17) and (18).

THEOREM 7 (Large-margin estimation). *Consider the same setup as in Theorem 5, and denote* $t_n = \frac{r(s_1+s_2)\log d}{n}$. *Suppose the surrogate loss* F *satisfies Assumption 1 with* $a_n \lesssim t_n^{(\alpha+1)/(\alpha+2)}$. *Set* $H \asymp t_n^{-1/2}$ *in* (18) *and* $\lambda \asymp t_n^{(\alpha+1)/(\alpha+2)} + t_n/\rho(\pi,\mathcal{N})$ *in* (17). *Then, with high probability at least* $1-\exp(-nt_n)$ *over the training data* $(\boldsymbol{X}_i,Y_i)_{i\in[n]}$, *we have:*

(a) *(Sign function estimation). For all* $\pi \in [-1,1]$ *except for a finite number of levels,*

$$\|\text{sgn}\hat{\phi}_{\pi,F} - \text{sgn}(f-\pi)\|_1 \lesssim t_n^{\frac{\alpha}{2+\alpha}} + \frac{1}{\rho^2(\pi,\mathcal{N})}t_n.$$

(b) *(Regression function estimation).*

$$\|\hat{f}_F - f\|_1 \lesssim (t_n \log n)^{\min\left(\frac{1}{2},\frac{\alpha}{2+\alpha}\right)}.$$

6. Simulations. In this section, we first evaluate the empirical performance of our method **ASSIST** through four experiments, with varying sample size, response type, matrix dimension, and model complexity. We then compare **ASSIST** with some alternative methods.

6.1. *Impacts of sample size, matrix dimension, and model complexity.* We consider a random matrix predictor $\boldsymbol{X} \in \mathbb{R}^{d\times d}$ with i.i.d. entries sampled from Uniform[0,1], and simulate two types of response, continuous and binary, through

- Continuous regression: $Y = f(\boldsymbol{X}) + \varepsilon$, where $\varepsilon \sim \text{Normal}(0, 0.1^2)$;
- Binary regression: $Y \in \{-1, 1\}$, with $\mathbb{P}(Y = 1|\boldsymbol{X}) = \frac{1}{2}(f(\boldsymbol{X}) + 1)$.

We set the regression function $f(\boldsymbol{X}) = h(z)$, where $h\colon \mathbb{R} \to [-1, 1]$ is a non-decreasing function, $z \in \mathbb{R}$ is a nonlinear predictor that $z = (G^{-1} \circ \bar{G})(\langle \boldsymbol{X}, \boldsymbol{B}\rangle)$, $\circ$ denotes function composition, $\boldsymbol{B} \in \mathbb{R}^{d\times d}$ is a fixed rank-r, supp-(s, s) matrix, $\bar{G}\colon \mathbb{R} \to [0, 1]$ is the CDF of $\langle \boldsymbol{X}, \boldsymbol{B}\rangle$ induced by $\boldsymbol{X} \sim \mathbb{P}_{\boldsymbol{X}}$ so that $\bar{G}(\langle \boldsymbol{X}, \boldsymbol{B}\rangle) \sim$ Uniform[0,1], and $G\colon \mathbb{R} \to [0, 1]$ is the CDF of some reference distribution. This construction yields a highly nonlinear function f. We set the matrix dimension $d = 20, 30, \ldots, 60$, the training sample size $n = 150, 200, \ldots, 400$, and various combinations of (r, s). In this study, we set $\lambda = 10^{-2}$, $H = 20$, and use the true (r, s) in Algorithm 1, and study parameter tuning in Section 6.2.

The first experiment assesses the impact of the sample size n for the continuous regression. We set $h(z) = [\exp(z) - 1]/[\exp(z) + 1]$, G as the CDF of a standard normal distribution, the matrix dimension $d = 20$, and the model complexity $(r, s) = (2, 2), (2, 3), (5, 5)$. Fig 4(a) summarizes the main model configurations, including the density of $z = z(\boldsymbol{X})$, the function $h = h(z)$, and the resulting density of $f(\boldsymbol{X})$. Fig 4(b) reports the prediction error, $\|\hat{f} - f\|_1$, as the sample n increases. We see that the error decays polynomially with n. We also see that a higher rank r or a higher support s leads to a larger error, as reflected by the upward shift of the curve as (r, s) increases, since it implies a higher model complexity.

The second experiment considers a binary response. Fig 4(c) reports the prediction error $\|\hat{f} - f\|_1$ as the sample size n increases. We see that the error decays polynomially with n. We also note that, in both cases, the matrix predictor has the dimension $20 \times 20 = 400$ whereas n is on the order of hundreds. Nevertheless, our nonparametric method consistently learns the function f well from limited data without specifying a priori the functional form.

The third experiment evaluates the impact of the matrix dimension d. We fix the sample size $n = 200$ and increase d. Fig 4(d) reports the prediction error. We see that the error increases slowly with d, and the growth appears well controlled by the log rate. Note that, in this example, as d increases, the number of effective entries remains unchanged, but the combinatoric complexity increases in the model space. The increasing error is an unavoidable price to pay for not knowing the positions of the s active entries. This example shows the ability of our method to effectively handle a massive number of noisy features.

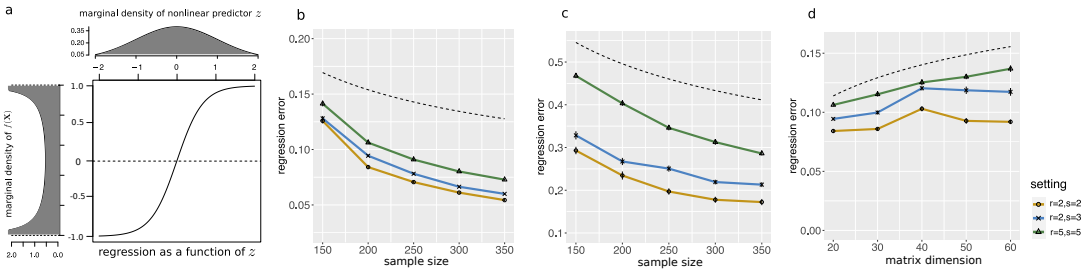

FIG 4. *Finite sample performance under a smooth function. (a) simulation setup; (b) prediction error with varying n and $d = 20$ for the continuous response; (c) for the binary response; (d) with varying d and $n = 200$. The dashed lines in panels (b)-(d) represent upper bounds $\mathcal{O}(n^{-1/3})$, $\mathcal{O}(n^{-1/3})$, and $\mathcal{O}(\log d)$, respectively. The results are based on 30 data replications.*

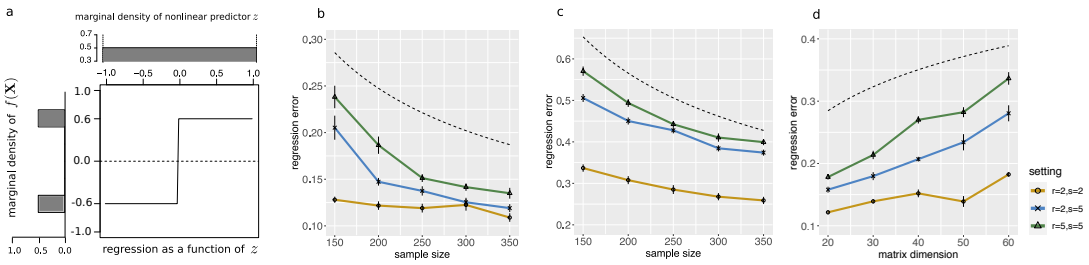

FIG 5. *Finite sample performance under a non-smooth function. The setup is similar as Fig 4. The dashed lines in panels (b)-(d) represent upper bounds* $\mathcal{O}(n^{-1/2})$*,* $\mathcal{O}(n^{-1/2})$*, and* $\mathcal{O}(\log d)$*, respectively.*

The fourth experiment investigates the impact of smoothness in regression function. In Section 2, we show that the probabilistic behavior of $f(\boldsymbol{X})$ plays a key role in our learning reduction approach. Here we assess the empirical performance by repeating all the above experiments using a model configuration with $z = z(\boldsymbol{X}) \sim \text{Uniform}[-1, 1]$, $h(z) = -0.6 + 1.2\mathbb{1}(z > 0)$, and $(r, s) = c(2, 2), (2, 5), (5, 5)$. This case falls on the other end of the spectrum in contrast to the infinity smooth function in Fig 4(a). That is, $f(\boldsymbol{X})$ now concentrates at two mass points $\pi = \pm 0.6$. This makes the π-sign function estimation challenging around $\pi = \pm 0.6$ because of the non-identifiability. Fig 5 reports the new model configurations and the corresponding results. Interestingly, we find that our method still maintains a good performance. Such a robustness may be explained by the fact that we aggregate in total $2H + 1$ sign functions, each of which incurs at most $1/(2H + 1)$ error to the regression function estimation. Therefore, our function estimate is robust against some off-target sign estimates, as long as the majority are accurate. This observation is consistent with the consistency result established in Section 3.

6.2. *Comparison with alternative methods.* Next, we compare our method with several popular alternative solutions. In this comparison, we adopt the simulation setup as in [27], but add more challenging matrix effects. Particularly, in this setup, the response is binary, and the predictor is a symmetric matrix that encodes a network. In this article, we have been targeting a general matrix predictor, which is directly applicable to a symmetric matrix, though we do not focus on symmetry. Moreover, as we show in Section A.4 of the Appendix, the data generating model falls into our general family of nonparametric trace regression when there is no noise, but no longer so when there is noise. Therefore, we also investigate the performance of our method under model misspecification when including the noise.

More specifically, we simulate from a latent variable model $(\boldsymbol{X}, Y)|\pi$, where we generate π i.i.d. from Uniform[0,1], and conditional on π, we generate $Y \sim \text{Bernoulli}(\pi)$, and

$$\boldsymbol{X} = [\![\boldsymbol{X}_{ij}]\!],\ \boldsymbol{X}_{ij} \overset{\text{indep.}}{\sim} \text{Normal}\left(g_{ij}(\pi)\mathbb{1}(\text{edge } (i,j) \text{ is active}), \sigma^2\right), \tag{22}$$

where the edge connectivity strength, denoted by $g_{ij}(\pi)$, varies depending on the location of $(i, j) \in [d]^2$, and the mean response π. Fig 6 shows the activation pattern we consider that specifies the locations of the active edges. The active region is further divided into several subregions, each of which has its own signal function $g_{ij}(\cdot) \colon [0, 1] \to \mathbb{R}$. The function form of $g_{ij}(\cdot)$ is randomly drawn from a pre-specified library consisting of common polynomial, log, and trigonometric functions. We set $d = 68$, the training sample size $n = 160$, and the testing size 80. In the noiseless case $\sigma = 0$ in (22), the cross and block patterns are low-rank with $r = 3$ and 5, respectively, whereas the star and circle patterns are nearly full-rank, with a numerical rank $r \approx 30$ on the supported submatrix.

We compare the following four estimation methods.

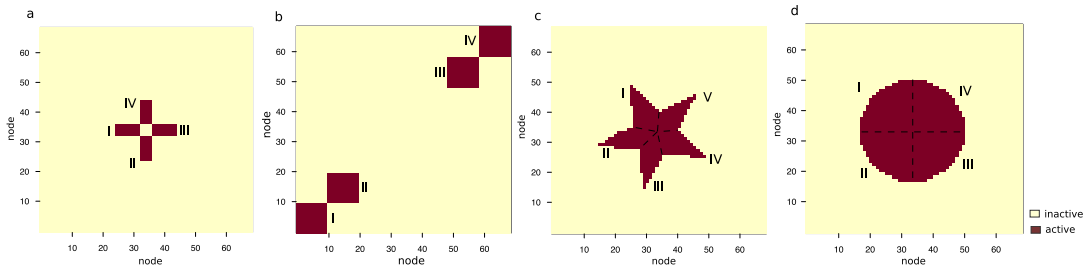

FIG 6. *Four activation patterns in simulations. The active region is divided into four or five subregions, denoted by I, II, ..., V, each of which has its own edge connectivity signal* $g_{pq}(\pi)$.

- Unstructured logistic regression for vector predictors (**LogisticV**, [46]). This method vectorizes the matrix predictor into a high dimensional vector, then employs a logistic loss with an elastic net penalty.
- Generalized trace regression for matrix predictors (**LogisticM**, [27]). This method fits a parametric trace regression model with a logistic link and a symmetric matrix predictor. It imposes a group lasso penalty to encourage two-way sparsity.
- Convolutional Neural Network (**CNN**) with two hidden layers implemented in Keras [7]. We apply 64 filters with 3×3 convolutional kernels to the matrix-valued predictor, followed by a pooling layer with size 5×5. The resulting features are fed to a fully connected layer of neural network with ReLU activation.
- **A**ggegration of **S**tructured **SI**gn **S**eries for **T**race regression (**ASSIST**), our method.

Among these methods, **LogisticV** serves as a baseline to assess the gain of modeling a matrix predictor over a vector predictor, **LogisticM** is a parametric model, whereas **CNN** and **ASSIST** are nonparametric solutions for matrix predictors. We feed each method with the binary response and the network adjacency matrix as the predictor after randomly permuting the node indices. Because **LogisticM** only supports a symmetric matrix predictor, we provide it with $(\boldsymbol{X} + \boldsymbol{X}^T)/2$ as the input. We use the default parameters of **LogisticM**, and select the tuning parameters of **LogisticV**, **CNN**, and our method **ASSIST**, including the rank r and sparsity parameters (r, s), by 5-fold cross validation.

Fig 7 reports both the prediction error $\|\hat{f} - f\|_1$ and the misclassification error at $\pi = 1/2$ of the four methods evaluated on the testing data. For prediction, we see that **ASSIST** consistently outperforms the alternatives, and the improvement is substantial. For example, the relative reduction using **ASSIST** over the next best approach, **LogisticM**, is over 20% for patterns (a) and (d), and over 15% for patterns (b) and (c). These results clearly demonstrate the benefit of our nonparametric approach. Moreover, we find that neither **LogisticV** nor **CNN**

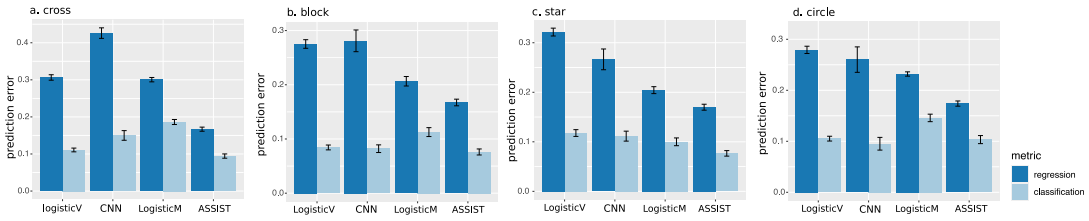

FIG 7. *Performance comparison of various methods under four different activation patterns. Reported are the prediction error* $\|\hat{f} - f\|_1$*, denoted by "regression", and the misclassification error at* $\pi = 1/2$*, denoted by "classification". The results are based on 30 data replications.*

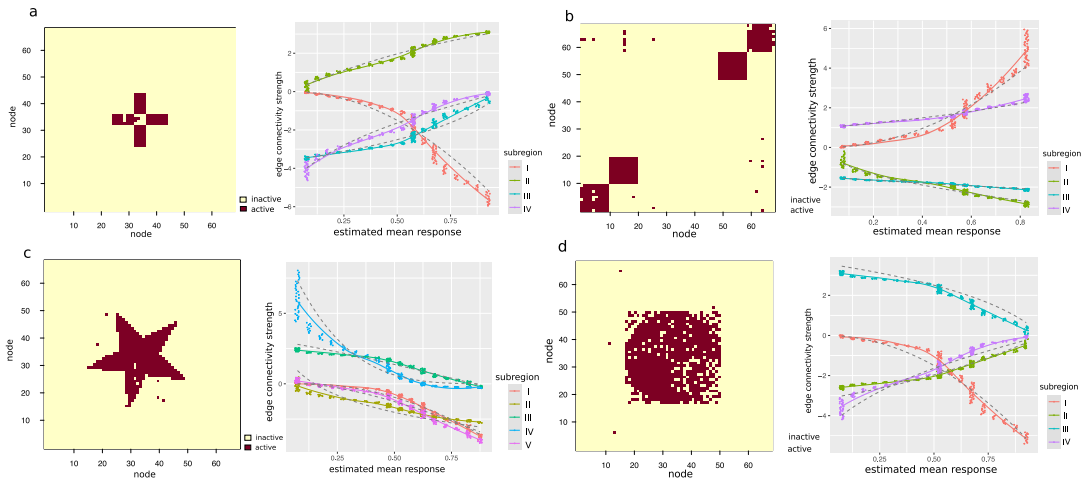

FIG 8. *Example output returned by* **ASSIST** *based on the moving average of the feature weights, and the scatter plot of the edge connectivity strength, averaged by each subregion, versus the estimated mean response. The dashed curve shows the true function.*

has a satisfactory prediction. A possible explanation is that **LogisticV** takes the vectorized matrix as the input and therefore loses the two-way pairing information. Meanwhile, **CNN** assumes spacial ordering within row and column indices. Although local similarity is important for the usual imaging analysis, the row and column indices take no particular order for a network. Actually, adjacency matrices after row or column permutation represent the same network, and thus the index-invariant methods, such as **LogisticM** and **ASSIST**, perform better. For classification, we also see that our method overall performs the best. The only exception is the circle pattern where **CNN** has a slightly lower classification error. This is perhaps due to the fact that the circle is nearly full rank and thus favors a more complicated model. Interestingly, we also find that the advantage of our method is more substantial in regression prediction than in classification, since classification is easier than regression. Moreover, with model noise included, our method still performs well even though the true model does not exactly follow our model specification.

Finally, to illustrate its capability of producing an estimate of high interpretability, Fig 8 reports the output of **ASSIST** based on the moving average of the feature weights $(\hat{\boldsymbol{B}}_\pi)_{\pi\in\Pi}$. It is observed that the identified activation region agrees well with the truth. We also investigate the relationship between the edge connectivity for individual i and the estimated mean response $\hat{\pi}_i$ for $i = 1, \ldots, n$. The trajectory accurately resembles the ground truth function in each subregion, demonstrating that our method is able to recover the pattern in the matrix predictors $\boldsymbol{X}_i$ against $\hat{\pi}_i$ on a continuous spectrum.

7. Real data applications. We present two real data applications, in parallel to the two matrix learning tasks studied in Section 4. The first task is the binary-valued trait prediction based on brain connectivity matrix regression, and the second is the continuous-valued matrix completion for imaging analysis.

7.1. *Brain connectivity analysis.* The first example is a brain connectivity data analysis, which aims to understand the relation between brain connectivity network and cognitive performance. The data is obtained from the Human Connectome Project (HCP) [34], and consists of $n = 212$ healthy subjects. For each subject, a binary connectivity network is extracted, with nodes corresponding to $d = 68$ brain regions-of-interest following the Desikan

TABLE 1

Brain connectivity analysis. (a) Comparison of prediction accuracy measured by AUC, with standard errors over 5-fold cross validation in the parentheses. For **CNN***, there is no report for node selection. (b) Top edges selected by the method* **ASSIST-p***. The letters "r" and "l" in node names indicate the right and left hemisphere, respectively. The p-value is calculated from the two-sample test of edge connection strength between two individual groups.*

a

Method	AUC	% of Active Nodes
ASSIST-p	**0.73 (0.03)**	88.2
ASSIST	**0.77 (0.04)**	97.3
LogisticM	0.72 (0.02)	100.0
LogisticV	0.68 (0.01)	89.7
CNN	0.67 (0.03)	-

b

Rank	Node	Node	p-value
1	r-inferiortemporal	r-middletemporal	0.01
2	r-parstriangularis	r-supramarginal	3e-5
3	l-posteriorcingulate	r-precentral	0.01
4	l-caudalmiddlefronta	l-isthmuscingulate	2e-5
5	l-lateralorbitofrontal	r-parstriangularis	1e-4

atlas [10], and links corresponding to the structural connectivity evaluated by diffusion tensor imaging [43]. The outcome is the dichotomized version of a visuospatial processing test score, corresponding to a high or low performance score [36]. We adjust age and gender as additional covariates in our analysis. We note that, although our model focuses on a matrix predictor, it is straightforward to incorporate additional vector-valued covariates. We use a random 60-20-20 split of the data for training, validation, and testing.

We compare our method with the same alternatives as in Section 6.2. Table 1(a) shows that our method achieves the highest accuracy, measured by the area under receiver operating characteristic (AUC). Moreover, as common in the high dimensional setting, we see the model with a good cross-validation accuracy tends to include a large number of noise variables. A useful heuristic called the "one-standard-error rule", suggested by [20], selects the most parsimonious model with cross-validation accuracy within one standard error of the best. We apply this rule and report the results as **ASSIST-p**. It is remarkable to see that **ASSIST-p** results in 12% reduction of active nodes but still achieves a comparable accuracy to the best one. Table 1(b) lists the top brain links identified by our method. The edges are ranked by their maximal values in the feature weights $(\hat{\boldsymbol{B}}_\pi)_{\pi\in\mathcal{H}}$ via moving averaging. We find that the top edges involve connections between frontal and occipital regions in the right hemisphere. This is consistent with recent findings of dysfunction in right posterior regions for deficits in visuospatial processing [36]. Fig 9(a) shows the top selected edges overlaid on a brain template. Moreover, we find the relationship between the edge connection strength and the mean response to be nonlinear. Fig 9(b) plots the edge connectivity strength versus the estimated mean response. We see that the connection between r-parstriangularis and r-supramarginal grows slowly when the mean response is small but fast when it is large. In

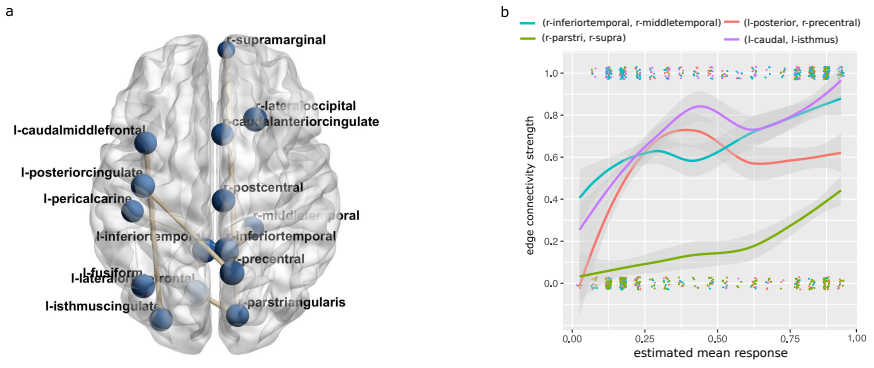

FIG 9. *Brain connectivity analysis. (a) Top edges overlaid on a brain template. (b) Edge connectivity strength versus estimated mean response. Colored curves represent the moving averages of connectivity strengths, gray bands represent one standard error, and jitter points represent the raw connectivity values (0 or 1).*

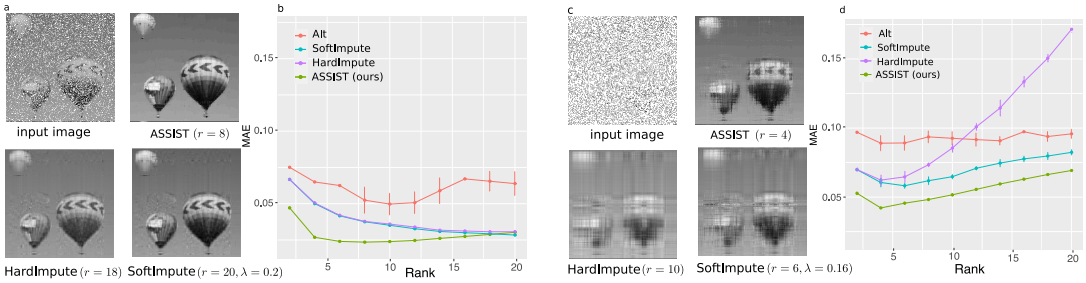

FIG 10. *Matrix completion analysis. (a)-(b) correspond to the 40% missing rate, and (c)-(d) the 80% missing rate. Error bars represent the standard error over 5-fold cross-validation. Numbers in the parentheses represent the selected tuning parameters for each method. In (a) and (c), we omit the worst method* **ALT** *for space consideration.*

contrary, the connection between r-posteriorcingulate and r-precentral grows fast initially, then reaches a plateau as the mean response increases. Such patterns suggest heterogeneous changes in brain connectivity with respect to the visuospatial processing capability.

7.2. *Imaging matrix completion.* The second application is an imaging matrix completion, where the goal is to recover and restore the partially observed gray-scaled hot air balloon image. This image is a standard benchmark in computer vision, and is organized as a 217-by-217 matrix, whose entries represent pixel values in $[0, 1]$. We randomly mask a subset of entries and perform matrix completion based on the observed entries.

We compare our method with three alternatives: a soft imputation method based on matrix nuclear norm regularization (**SoftImpute**) [19], a hard imputation method with ridge regression (**HardImpute**) [24], and a hard imputation based on alternating SVD (**ALT**) [28]. We evaluate the recovery accuracy by MAE on the unobserved entries, and we tune all the parameters based on 5-fold cross-validation.

We investigate missing percentages at 40% and 80%, and vary the rank $r = 2, 4, \ldots, 20$. Fig 10 reports the performances of the four methods. We see clearly that our method achieves the best image recovery, with the smallest MAE. Besides, the advantage of our method compared to the alternative solutions is more clear when the missing percentage increases.

8. Discussion. We have developed a nonparametric trace regression model for studying the relationship between a scalar response and a high dimensional matrix predictor. We propose a learning reduction approach, **ASSIST**, using the structured sign function series, which bridges between regression and classification. We establish the theoretical bounds, which concern the fundamental statistical errors, are independent of specific algorithms, and serve as a benchmark on how well any algorithmic procedure could perform. Our numerical results demonstrate the competitive performance of the proposed method.

Our work unlocks several possible future directions. One is nonparametric modeling of other nonconventional predictors, such as tensors, functions, and manifold data. Other directions include multi-task learning and compressed sensing. Moreover, our learning reduction approach can be coupled with more sophisticated classifiers, such as neural networks, decision trees, and boosting, for sign function estimation. Finally, the theoretical guarantees we obtain are for the global optimum. How to characterize the behavior of the actual minimizer, or relatedly, the computational error for non-convex matrix based regression remains challenging and open. All these questions are warranted for future research.

Acknowledgements. The research was supported in part by NSF DMS-1915978, NSF DMS-2023239, Wisconsin Alumni Research Foundation (to M. Wang), NIH R01 AG061303 (to L. Li), and NSF CCF-1740858 (to H. Zhang)

SUPPLEMENTARY MATERIAL

Supplementary Appendix includes all technical proofs and additional results. Our software **ASSIST** and data used in our analysis are publicly available at https://github.com/Miaoyanwang/ASSIST.

REFERENCES

[1] BALABDAOUI, F., DUROT, C. and JANKOWSKI, H. (2019). Least squares estimation in the monotone single index model. *Bernoulli* **25** 3276–3310.

[2] CAI, T., CAI, T. T. and ZHANG, A. (2016). Structured Matrix Completion with Applications to Genomic Data Integration. *Journal of the American Statistical Association* **111** 621-633.

[3] CANDES, E. J. and PLAN, Y. (2011). Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory* **57** 2342–2359.

[4] CARUANA, R. (1997). Multitask learning. *Machine learning* **28** 41–75.

[5] CHAN, S. and AIROLDI, E. (2014). A consistent histogram estimator for exchangeable graph models. In *International Conference on Machine Learning* 208–216.

[6] CHI, E. C., GAINES, B. J., SUN, W. W., ZHOU, H. and YANG, J. (2020). Provable Convex Co-clustering of Tensors. *Journal of Machine Learning Research* **21** 1–58.

[7] CHOLLET, F. and ALLAIRE, J. J. (2018). *Deep Learning mit R und Keras: Das Praxis-Handbuch von den Entwicklern von Keras und RStudio*. MITP-Verlags GmbH & Co. KG.

[8] COHN, H. and UMANS, C. (2013). Fast matrix multiplication using coherent configurations. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms* 1074–1087. SIAM.

[9] DE WOLF, R. (2003). Nondeterministic quantum query and communication complexities. *SIAM Journal on Computing* **32** 681–699.

[10] DESIKAN, R. S., SÉGONNE, F., FISCHL, B., QUINN, B. T., DICKERSON, B. C., BLACKER, D., BUCKNER, R. L., DALE, A. M., MAGUIRE, R. P., HYMAN, B. T. et al. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *Neuroimage* **31** 968–980.

[11] FAN, J., GONG, W. and ZHU, Z. (2019). Generalized high-dimensional trace regression via nuclear norm regularization. *Journal of Econometrics* **212** 177–202.

[12] GANTI, R., RAO, N., BALZANO, L., WILLETT, R. and NOWAK, R. (2017). On learning high dimensional structured single index models. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* 1898–1904.

[13] GANTI, R. S., BALZANO, L. and WILLETT, R. (2015). Matrix completion under monotonic single index models. In *Advances in Neural Information Processing Systems* **28** 1873–1881.

[14] GAO, C., LU, Y., MA, Z. and ZHOU, H. H. (2016). Optimal estimation and completion of matrices with biclustering structures. *Journal of Machine Learning Research* **17** 5602–5630.

[15] GIBOU, F., FEDKIW, R. and OSHER, S. (2018). A review of level-set methods and some recent applications. *Journal of Computational Physics* **353** 82–109.

[16] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. and BENGIO, Y. (2016). *Deep learning* **1(2)**. MIT press Cambridge.

[17] HAMIDI, N. and BAYATI, M. (2019). On low-rank trace regression under general sampling distribution. *arXiv preprint arXiv:1904.08576*.

[18] HAO, B., WANG, B., WANG, P., ZHANG, J., YANG, J. and SUN, W. W. (2019). Sparse tensor additive regression. *arXiv preprint arXiv:1904.00479*.

[19] HASTIE, T., MAZUMDER, R., LEE, J. D. and ZADEH, R. (2015). Matrix completion and low-rank SVD via fast alternating least squares. *The Journal of Machine Learning Research* **16** 3367–3402.

[20] HASTIE, T., TIBSHIRANI, R. and WAINWRIGHT, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. CRC press.

[21] HU, W., SHEN, W., ZHOU, H. and KONG, D. (2020). Matrix Linear Discriminant Analysis. *Technometrics* **62** 196–205.

[22] LEE, C. and WANG, M. (2020). Tensor denoising and completion based on ordinal observations. In *International Conference on Machine Learning* 5778–5788.

[23] MA, Z. (2013). Sparse principal component analysis and iterative thresholding. *The Annals of Statistics* **41** 772–801.

[24] MAZUMDER, R., HASTIE, T. and TIBSHIRANI, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research* **11** 2287–2322.

[25] PARIKH, N. and BOYD, S. (2014). Proximal algorithms. *Foundations and Trends in Optimization* **1** 127–239.

[26] RECHT, B., FAZEL, M. and PARRILO, P. A. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review* **52** 471–501.

[27] RELIÓN, J. D. A., KESSLER, D., LEVINA, E. and TAYLOR, S. F. (2019). Network classification with applications to brain connectomics. *The Annals of Applied Statistics* **13** 1648–1677.

[28] RENNIE, J. D. and SREBRO, N. (2005). Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning* 713–719.

[29] SCOTT, C. (2011). Surrogate losses and regret bounds for cost-sensitive classification with example-dependent costs. In *International Conference on Machine Learning*.

[30] SHEN, X., TSENG, G. C., ZHANG, X. and WONG, W. H. (2003). On ψ-learning. *Journal of the American Statistical Association* **98** 724–734.

[31] SINGH, A., SCOTT, C. and NOWAK, R. (2009). Adaptive Hausdorff estimation of density level sets. *The Annals of Statistics* **37** 2760–2782.

[32] TSYBAKOV, A. B. (1997). On nonparametric estimation of density level sets. *The Annals of Statistics* **25** 948–969.

[33] TSYBAKOV, A. B. (2004). Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics* **32** 135–166.

[34] VAN ESSEN, D. C., SMITH, S. M., BARCH, D. M., BEHRENS, T. E., YACOUB, E., UGURBIL, K. and CONSORTIUM, W.-M. H. (2013). The WU-Minn human connectome project: an overview. *Neuroimage* **80** 62–79.

[35] WANG, J., SHEN, X. and LIU, Y. (2008). Probability estimation for large-margin classifiers. *Biometrika* **95** 149–167.

[36] WANG, L., ZHANG, Z. and DUNSON, D. (2019). Common and individual structure of brain networks. *The Annals of Applied Statistics* **13** 85–112.

[37] WANG, X., ZHU, H. and INITIATIVE, A. D. N. (2017). Generalized scalar-on-image regression models via total variation. *Journal of the American Statistical Association* **112** 1156–1168.

[38] WANG, Z., CURRY, E. and MONTANA, G. (2014). Network-guided regression for detecting associations between DNA methylation and gene expression. *Bioinformatics* **30** 2693–2701.

[39] XU, Z., DAN, C., KHIM, J. and RAVIKUMAR, P. (2020). Class-Weighted Classification: Trade-offs and Robust Approaches. In *International Conference on Machine Learning*.

[40] YANG, D., MA, Z. and BUJA, A. (2016). Rate Optimal Denoising of Simultaneously Sparse and Low Rank Matrices. *Journal of Machine Learning Research* **17** 1-27.

[41] YUAN, M. and ZHANG, C.-H. (2016). On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics* **16** 1031–1068.

[42] ZHANG, T., WU, J., LI, F., CAFFO, B. and BOATMAN-REICH, D. (2015). A Dynamic Directional Model for Effective Brain Connectivity Using Electrocorticographic (ECoG) Time Series. *Journal of the American Statistical Association* **110** 93-106.

[43] ZHANG, Z., DESCOTEAUX, M., ZHANG, J., GIRARD, G., CHAMBERLAND, M., DUNSON, D., SRIVASTAVA, A. and ZHU, H. (2018). Mapping population-based structural connectomes. *NeuroImage* **172** 130–145.

[44] ZHOU, H. and LI, L. (2014). Regularized matrix regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **76** 463–483.

[45] ZHOU, Y., WONG, R. K. and HE, K. (2020). Broadcasted Nonparametric Tensor Regression. *arXiv preprint arXiv:2008.12927*.

[46] ZOU, H. and HASTIE, T. (2005). Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **67** 301–320.