Robust and Resource-Efficient Identification of Two Hidden Layer Neural Networks

Massimo Fornasier\textsuperscript{1,3} · Timo Klock\textsuperscript{2} · Michael Rauchensteiner\textsuperscript{1}

Received: 30 June 2019 / Accepted: 6 July 2020 / Published online: 30 June 2021
© The Author(s) 2021

Abstract
We address the structure identification and the uniform approximation of two fully nonlinear layer neural networks of the type $f(x) = 1^T h(B^T g(A^T x))$ on $\mathbb{R}^d$, where $g = (g_1, \ldots, g_{m_0})$, $h = (h_1, \ldots, h_{m_1})$, $A = (a_1|\ldots|a_{m_0}) \in \mathbb{R}^{d \times m_0}$ and $B = (b_1|\ldots|b_{m_1}) \in \mathbb{R}^{m_0 \times m_1}$, from a small number of query samples. The solution of the case of two hidden layers presented in this paper is crucial as it can be further generalized to deeper neural networks. We approach the problem by sampling actively finite difference approximations to Hessians of the network. Gathering several approximate Hessians allows reliably to approximate the matrix subspace $W$ spanned by symmetric tensors $a_1 \otimes a_1, \ldots, a_{m_0} \otimes a_{m_0}$ formed by weights of the first layer together with the entangled symmetric tensors $v_1 \otimes v_1, \ldots, v_{m_1} \otimes v_{m_1}$, formed by suitable combinations of the weights of the first and second layer as $v_\ell = A G_0 b_\ell / \|A G_0 b_\ell\|_2$, $\ell \in [m_1]$, for a diagonal matrix $G_0$ depending on the activation functions of the first layer. The identification of the 1-rank symmetric tensors within $W$ is then performed by the solution of a robust nonlinear program, maximizing the spectral norm of the competitors constrained over the unit Frobenius sphere. We provide guarantees of stable recovery under a posteriori verifiable conditions. Once the 1-rank symmetric tensors $\{a_i \otimes a_i, i \in [m_0]\} \cup \{v_\ell \otimes v_\ell, \ell \in [m_1]\}$ are computed, we address their
correct attribution to the first or second layer ($a_i$’s are attributed to the first layer). The attribution to the layers is currently based on a semi-heuristic reasoning, but it shows clear potential of reliable execution. Having the correct attribution of the $a_i$, $v_\ell$ to the respective layers and the consequent de-parametrization of the network, by using a suitably adapted gradient descent iteration, it is possible to estimate, up to intrinsic symmetries, the shifts of the activations functions of the first layer and compute exactly the matrix $G_0$. Eventually, from the vectors $v_\ell = A G_0 b_\ell/\| A G_0 b_\ell \|_2$’s and $a_i$’s one can disentangle the weights $b_\ell$’s, by simple algebraic manipulations. Our method of identification of the weights of the network is fully constructive, with quantifiable sample complexity and therefore contributes to dwindle the black box nature of the network training phase. We corroborate our theoretical results by extensive numerical experiments, which confirm the effectiveness and feasibility of the proposed algorithmic pipeline.

**Keywords** Deep neural networks · Active sampling · Exact identifiability · Deparametrization · Frames · Nonconvex optimization on matrix spaces

**Mathematics Subject Classification** 65D15 · 68T07 · 90C26

## Contents

1 Introduction .............................................476
1.1 Notation .............................................478
1.2 From One Artificial Neuron to Shallow and Deeper Networks .............................................478
1.2.1 Meet the Neuron .....................................478
1.2.2 Shallow Networks: The One-Layer Case .........................480
1.2.3 Deeper Networks: The Two-Layer Case .........................484
2 Approximating the Span of Tensors of Weights ...........................486
2.1 Estimating Hessians of the Network by Finite Differences ..................491
2.2 Span of Tensors of (Entangled) Network Weights: Proof of Theorem 5 ............491
3 Recovery of Individual (Entangled) Neural Network Weights ...................496
3.1 Recovery Guarantees ......................................497
3.2 A Nonlinear Program: Properties of Local Maximizers of (31) ...............500
4 Numerical Experiments About the Recovery of Network Profiles .................511
5 Open Problems ............................................514
6 Reconstruction of the Entire Network ................................516
6.1 Distinguishing First and Second Layer Weights ........................517
6.2 Reconstructing the Network Function Using Gradient Descent ...............519
Appendix ................................................524
Additional Proofs for Section 2 ...................................524
Additional Results and Proofs for Section 3 .............................528
Proof of Proposition 26 .......................................532
References ................................................533

1 Introduction

Deep learning is perhaps one of the most sensational scientific and technological developments in the industry of the last years. Despite the spectacular success of deep
neural networks (NN) outperforming other pattern recognition methods, achieving even superhuman skills in some domains \cite{12,36,57} and confirmations of empirical successes in other areas such as speech recognition \cite{27}, optical character recognition \cite{8}, games solution \cite{44,55}, the mathematical understanding of the technology of machine learning is in its infancy. This is not only unsatisfactory from a scientific, especially mathematical point of view, but it also means that deep learning currently has the character of a black box method, and its success cannot be ensured yet by a full theoretical explanation. This leads to lack of acceptance in many areas, where interpretability is a crucial issue (like security, cf. \cite{10}) or for those applications where one wants to extract new insights from data \cite{60}.

Several general mathematical results on neural networks have been available since the 1990s \cite{2,17,38,39,46–48}, but deep neural networks have special features and in particular superior properties in applications that still cannot be fully explained from the known results. In recent years, new interesting mathematical insights have been derived for understanding approximation properties (expressivity) \cite{19,53} and stability properties \cite{9,67} of deep neural networks. Several other crucial and challenging questions remain open.

A fundamental one is about the number of required training data to obtain a good neural network, i.e., achieving small generalization errors for future data. Classical statistical learning theory splits this error into bias and variance and gives general estimations by means of the so-called VC dimension or Rademacher complexity of the used class of neural networks \cite{54}. However, the currently available estimates of these parameters \cite{26} provide very pessimistic barriers in comparison to empirical success. In fact, the trade-off between bias and variance is function of the complexity of a network, which should be estimated by the number of sampling points to identify it uniquely. Thus, on the one hand, it is of interest to know which neural networks can be uniquely determined in a stable way by finitely many training points. On the other hand, the unique identifiability is clearly a form of interpretability.

The motivating problem of this paper is the robust and resource-efficient identification of feedforward neural networks. Unfortunately, it is known that identifying a very simple (but general enough) neural network is indeed NP-hard \cite{7,33}. Even without invoking fully connected neural networks, recent work \cite{22,41} showed that even the training of one single neuron (ridge function or single index model) can show any possible degree of intractability, depending on the distribution of the input. Recent results \cite{3,34,42,52,56}, on the other hand, are more encouraging and show that minimizing a square loss of a (deep) neural network does not have in general or asymptotically (for large number of neurons) poor local minima, although it may retain the presence of critical saddle points.

In this paper, we present conditions for a fully nonlinear two-layer neural network to be provably identifiable with a number of samples, which is polynomially depending on the dimension of the network. Moreover, we prove that our procedure is robust to perturbations. Our result is clearly of theoretical nature, but also fully constructive and easily implementable. To our knowledge, this work is the first, which allows provable de-parametrization of the problem of deep network identification, beyond the simpler case of shallow (one hidden) layer neural networks already considered in very recent literature \cite{3,23,32,34,42,43,52,56}. For the implementation, we do not require black
box high-dimensional optimization methods and no concerns about complex energy loss landscapes need to be addressed, but only classical and relatively simple calculus and linear algebra tools are used (mostly function differentiation and singular value decompositions). The results of this paper build upon the work [22,23], where the approximation from a finite number of sampling points has been already derived for the single neuron and one-layer neural networks. The generalization of the approach of the present paper to networks with more than two hidden layers is surprisingly simpler than one may expect, and it is in the course of finalization [21], see Sect. 5 (v) below for some details.

1.1 Notation

Let us collect here some notation used in this paper. Given any integer \( m \in \mathbb{N} \), we use the symbol \( \{m\} := \{1, 2, \ldots, m\} \) for indicating the index set of the first \( m \) integers. We denote by \( B_1^d \) the Euclidean unit ball in \( \mathbb{R}^d \), \( S_{d-1} \) the Euclidean sphere, and \( \mu_{S_{d-1}} \) its uniform probability measure. We denote \( \ell_q^d \) the \( d \)-dimensional Euclidean space endowed with the norm \( \|x\|_{\ell_q^d} = \left( \sum_{j=1}^{d} |x_j|^q \right)^{1/q} \). For \( q = 2 \), we often write indifferently \( \|x\| = \|x\|_2 = \|x\|_{\ell_2^d} \). For a matrix \( M \), we denote \( \sigma_k(M) \) its \( k^{th} \) singular value. We denote \( S \) the sphere of symmetric matrices of unit Frobenius norm \( \|\cdot\|_F \). The spectral norm of a matrix is denoted \( \|\cdot\|_F \). Given a closed convex set \( C \), we denote \( P_C \) the orthogonal projection operator onto \( C \). (Sometimes we use such operators to project onto subspaces of \( \mathbb{R}^d \) or subspaces of symmetric matrices or onto balls of such spaces.) For vectors \( x_1, \ldots, x_k \in \mathbb{R}^d \), we denote the tensor product \( x_1 \otimes \cdots \otimes x_k \) as the tensor of entries \( (x_{i_1i_2\ldots i_k})_{i_1\ldots i_k} \). For the case of \( k = 2 \), the tensor product \( x \otimes y \) of two vectors \( x, y \in \mathbb{R}^d \) equals the matrix \( xy^T = (x_i y_j)_{ij} \). For any matrix \( M \in \mathbb{R}^{m \times n} \)

\[
vec(M) := (m_{11}, m_{21}, \ldots, m_{m1}, m_{12}, m_{22}, \ldots, m_{mn})^T \in \mathbb{R}^{mn},
\]

is its vectorization, which is the vector created by the stacked columns of \( M \).

1.2 From One Artificial Neuron to Shallow and Deeper Networks

1.2.1 Meet the Neuron

The simplest artificial neural network \( f : \Omega \subset \mathbb{R}^d \to \mathbb{R} \) is a network consisting of exactly one artificial neuron, which is modeled by a ridge-function (or single-index model) \( f \) as

\[
f(x) = \phi(a^T x + \theta) = g(a^T x),
\]

where \( g : \mathbb{R} \to \mathbb{R} \) is the shifted activation function \( \phi(\cdot + \theta) \) and the vector \( a \in \mathbb{R}^d \) expresses the weight of the neuron. Since the beginning of the 1990s [30,31], there is a vast mathematical statistics literature about single-index models, which
addresses the problem of approximating \( a \) and possibly also \( g \) from a finite number of samples of \( f \) to yield an expected least-squares approximation of \( f \) on a bounded domain \( \Omega \subset \mathbb{R}^d \). Now assume for the moment that we can evaluate the network \( f \) at any point in its domain, we refer to this setting as active sampling. A straightforward way to construct uniform approximations, we adhere here to the language of recent results about the sampling complexity of ridge functions from the approximation theory literature, e.g., [13,22,41]. In those papers, the identification of the neuron is performed by using approximate differentiation. Let us clarify how this method works as it will be of inspiration for the further developments below. For any \( \epsilon > 0 \), points \( x_i, i = 1, \ldots, m \), and differentiation directions \( \phi_j, j = 1, \ldots, m \), we have

\[
\frac{f(x_i + \epsilon \phi_j) - f(x_i)}{\epsilon} \approx \frac{\partial f(x_i)}{\partial \phi_j} = g'(a^T x_i)a^T \phi_j.
\]  

(1)

Hence, differentiation exposes the weight of a neuron and allows to test it against test vectors \( \varphi_j \). The approximate relationship (1) forms for every fixed index \( i \) a linear system of dimensions \( m \times d \), whose unknown is \( x_i^* = g'(a^T x_i)a \). Solving approximately and independently, the systems for \( i = 1, \ldots, m \) yield multiple approximations \( \hat{a} = x_i^* / \| x_i^* \|_2 \approx a \) of the weight, the most stable of them with respect to the approximation error in (1) is the one for which \( \| x_i^* \|_2 \) is maximal. Once \( \hat{a} \approx a \) is learned, then one can easily construct a function \( \hat{f}(x) = \hat{g}(\hat{a}^T x) \) by approximating \( \hat{g}(t) \approx f(\hat{a}t) \) on further sampling points. Under assumptions of smoothness of the activation function \( g \in C^s([0, 1]) \), for \( s > 1 \), \( g'(0) \neq 0 \) and compressibility of the weight, i.e., \( \| a \|_{\ell_q} \) is small for \( 0 < q \leq 1 \), then by using \( L \) sampling points of the function \( f \) and the approach sketched above, one can construct a function \( \hat{f}(x) = \hat{g}(\hat{a}^T x) \) such that

\[
\| f - \hat{f} \|_{C(\Omega)} \leq c \| a \|_{\ell_q} \left\{ L^{-s} + \| g \|_{C^s([0, 1])} \left( \frac{1 + \log(d/L)}{L} \right)^{1/q - 1/2} \right\}.
\]

In particular, the result constructs the approximation of the neuron with an error, which has polynomial rate with respect to the number of samples, depending on the smoothness of the activation function and the compressibility of the weight vector \( a \). The dependence on the input dimension is only logarithmical. To take advantage of the compressibility of the weight, compressive sensing [24] is a key tool to solve the linear systems (1). In [13], such an approximation result was obtained by active and deterministic choice of the input points \( x_i \). In order to relax a bit the usage of active sampling, in the paper [22] a random sampling of the points \( x_i \) has been proposed and the resulting error estimate would hold with high probability. The assumption \( g'(0) \neq 0 \) is somehow crucial, since it was pointed out in [22,41] that any level of tractability (polynomial complexity) and intractability (super-polynomial complexity) of the problem may be exhibited otherwise.
1.2.2 Shallow Networks: The One-Layer Case

Combining several neurons leads to richer function classes \([38, 39, 46–48]\). A neural network with one hidden layer and one output is simply a weighted sum of neurons whose activation function only differs by a shift, i.e.,

\[
f(x) = \sum_{i=1}^{m} b_i \phi(a_i^T x + \theta_i) = \sum_{i=1}^{m} g_i(a_i^T x),
\]

where \(a_i \in \mathbb{R}^m\) and \(b_i, \theta_i \in \mathbb{R}\) for all \(i = 1, \ldots, m\). Sometimes, it may be convenient below the more compact writing \(f(x) = 1^T g(A^T x)\) where \(g = (g_1, \ldots, g_m)\) and \(A = [a_1 | \ldots | a_m] \in \mathbb{R}^{d \times m}\). Differently from the case of the single neuron, the use of first-order differentiation

\[
\nabla f(x) = \sum_{i=1}^{m} g_i'(a_i^T x)a_i \in A = \text{span} \{a_1, \ldots, a_m\},
\]

may furnish information about \(A = \text{span} \{a_1, \ldots, a_m\}\) (active subspace identification \([14, 15]\), see also \([22, \text{Lemma 2.1}]\)), but it does not allow yet to extract information about the single weights \(a_i\). For that higher-order information is needed. Recent work shows that the identification of a network \((2)\) can be related to tensor decompositions \([1, 23, 32, 43]\). As pointed out in Sect. 1.2.1, differentiation exposes the weights. In fact, one way to relate the network to tensors and tensor decompositions is given by higher-order differentiation. In this case, the tensor takes the form

\[
D^k f(x) = \sum_{i=1}^{m} g_i^{(k)}(x) a_i \otimes \cdots \otimes a_i,
\]

which requires that the \(g_i\)’s are sufficiently smooth. In a setting where the samples are actively chosen, it is generally possible to approximate these derivatives by finite differences. However, even for passive sampling there are ways to construct similar tensors \([23, 32]\), which rely on Stein’s lemma \([58]\) or differentiation by parts or weak differentiation. Let us explain how passive sampling in this setting may be used for obtaining tensor representations of the network. If the probability measure of the sampling points \(x_i\)’s is \(\mu_X\) with known (or approximately known \([18]\)) density \(p(x)\) with respect to the Lebesgue measure, i.e., \(d\mu_X(x) = p(x)dx\), then we can approximate the expected value of higher order derivatives by using exclusively point evaluations

\[\text{\textsuperscript{1}}\]

Below, with slight abuse of notation, we may use the symbol \(A\) also for the span of the weights \(\{a_i : i = 1, \ldots, m\}\).
of \( f \). This follows from

\[
\frac{1}{N} \sum_{i=1}^{N} f(x_i)(-1)^k \frac{\nabla^k p(x_i)}{p(x_i)} \approx \int_{\mathbb{R}^d} f(x)(-1)^k \frac{\nabla^k p(x)}{p(x)} p(x) dx
\]

\[
= \int_{\mathbb{R}^d} \nabla^k f(x) d\mu_X(x) = \mathbb{E}_{x \sim \mu_X}[\nabla^k f(x)]
\]

\[
= \sum_{i=1}^{m} \left( \int_{\mathbb{R}^d} g^{(k)}(a_i^T x) d\mu_X(x) \right) a_i \otimes \cdots \otimes a_i.\
\]

In the work [32], decompositions of third-order symmetric tensors \((k = 3)\) [1,35,51] have been used for the weights identification of one hidden layer neural networks. Instead, beyond the classical results about principal Hessian directions [37], in [23] it is shown that using second derivatives \((k = 2)\) actually suffices and the corresponding error estimates reflect positively the lower order and potential of improved stability, see e.g., [16,28,29]. The main part of the present work is an extension of the latter approach, and therefore, we will give a short summary of it with emphasis on active sampling, which will be assumed in this paper as the sampling method. The first step of the approach in [23] is taking advantage of (3) to reduce the dimensionality of the problem from \( d \) to \( m \).

**Reduction to the active subspace**  Before stating the core procedure, we want to introduce a simple and optional method, which can help to reduce the problem complexity in practice. Assume \( f : \Omega \subset \mathbb{R}^d \to \mathbb{R} \) takes the form (2), where \( d \geq m \) and that \( a_1, \ldots, a_m \in \mathbb{R}^d \) are linearly independent. From a numerical perspective the input dimension \( d \) of the network plays a relevant role in terms of complexity of the procedure. For this reason in [23], the input dimension is effectively reduced to the number of neurons in the first hidden layer. With this reasoning, in the sections that follow we also consider networks where the input dimension matches the number of neurons of the first hidden layer.

Assume for the moment that the active subspace \( A = \text{span}\{a_1, \ldots, a_m\} \) is known. Let us choose any orthonormal basis of \( A \) and arrange it as the columns of a matrix \( \hat{A} \in \mathbb{R}^{d \times m} \). Then

\[
f(x) = f(P_A x) = f(\hat{A} A^T x),
\]

which can be used to define a new network

\[
\hat{f}(y) := f(\hat{A} y) : \mathbb{R}^m \to \mathbb{R}.
\]

whose weights are \( \alpha_1 = \hat{A}^T a_1, \ldots, \alpha_m = \hat{A}^T a_m \); all the other parameters remain unchanged. Note that \( A \alpha_i = P_A a_i = a_i \), and therefore, \( a_i \) can be recovered from \( \alpha_i \). In summary, if the active subspace of \( f \) is approximately known, then we can construct \( \hat{f} \), such that the identification of \( f \) and \( \hat{f} \) are equivalent. This allows us to reduce the problem to the identification of \( \hat{f} \) instead of \( f \), under the condition
Algorithm 1: Active subspace identification [23]

\begin{algorithm}
\caption{Active subspace identification [23]}
\hspace*{1em}Input: Given a shallow neural network $f$ as in (2), step-size of finite differences $\epsilon > 0$, number of samples $m_X$
\hspace*{1em}begin
\hspace*{2em}1. Draw $x_1, \ldots, x_{m_X}$ uniformly from the unit sphere $S^{d-1}$
\hspace*{2em}2. Calculate the estimated gradients $\Delta_\epsilon f(x_1), \ldots, \Delta_\epsilon f(x_{m_X})$ by first-order finite differences with stepsize $\epsilon$.
\hspace*{2em}3. Compute the singular value decomposition
\hspace*{3em}$$(\Delta_\epsilon f(x_1)| \ldots | \Delta_\epsilon f(x_{m_X})) = (\hat{U}_1 \hat{U}_2) \left( \begin{array}{cc} \hat{\Sigma}_1 & 0 \\ 0 & \hat{\Sigma}_2 \end{array} \right) \left( \begin{array}{c} \hat{V}_1^T \\ \hat{V}_2^T \end{array} \right),$$
\hspace*{4em}where $\hat{\Sigma}_1$ contains the $m$ largest singular values. Set $P_A = \hat{U}_1 \hat{U}_1^T$.
\hspace*{1em}end
\hspace*{1em}Output: $P_A$
\end{algorithm}

that we approximate $P_A$ well enough [23, Theorem 1.1]. As recalled in (3), we can produce easily approximations to vectors in $A$ by approximate first order differentiation of the original network $f$ and, in an ideal setting, generating $m$ linear independent gradients would suffice to approximate $A$. However, in general, there is no way to ensure a priori such linear independence and we have to account for the error caused by approximating gradients by finite differences. By suitable assumptions on $f$ (see the full rank condition on the matrix $J[f]$ defined in (4) below) and using Algorithm 1, we obtain the following approximation result.

**Theorem 1** ([23],Theorem 2.2) Assume the vectors $(a_i)_{i=1}^m$ are linear independent and of unit norm. Additionally, assume that the $g_i$'s are smooth enough. Let $P_A$ be constructed as described in Algorithm 1 by sampling $m_X(d+1)$ values of $f$. Let $0 < s < 1$, and assume that the matrix

$$J[f] := \mathbb{E}_{X \sim \mu_{S^{d-1}}} \nabla f(X) \otimes \nabla f(X) = \int_{S^{d-1}} \nabla f(x) \nabla f(x)^T d\mu_{S^{d-1}}(x)$$

has full rank, i.e., its $m$-th singular value fulfills $\sigma_m(J[f]) \geq \alpha > 0$. Then

$$\|P_A - P_A\|_F \leq \frac{2C_1 \epsilon m}{\sqrt{\alpha (1 - s)} - C_1 \epsilon m},$$

with probability at least $1 - m \exp\left(-\frac{m_X \alpha s^2}{2m^2C_2^2}\right)$, where $C_1, C_2 > 0$ are absolute constants depending on the smoothness of $g_i$'s.

**Identifying the weights** As clarified in the previous section, we can assume from now on that $d = m$ without loss of generality. Let $f$ be a network of the type (2), with twice differentiable activation functions $(g_i)_{i=1, \ldots, m}$, and independent weights

\[ – Springer \]
The idea is, first of all, to modify the network by an *ad hoc* linear transformation (withening) of the input

\[ f(W^T x) = \sum_{i=1}^{m} g_i (a_i^T W^T x) \]

in such a way that \( (W a_i / \| W a_i \|_2)_{i=1,\ldots,m} \) forms an orthonormal system. The computation of \( W \) can be performed by spectral decomposition of any positive definite matrix \( \hat{A} \approx A = \text{span} \{ a_1 \otimes a_1, \ldots, a_m \otimes a_m \} \). By using (5) and a similar approach as Algorithm 1 (one simply substitutes there the approximate gradients with vectorized approximate Hessians), one can compute \( \hat{A} \) under the assumption that also the second-order matrix

\[ H[f] := \mathbb{E}_{X \sim \mu_{S_m^{-1}}} \text{vec}(\nabla^2 f(X)) \otimes \text{vec}(\nabla^2 f(X)) \]

is of full rank, where \( \text{vec}(\nabla^2 f(X)) \) is the vectorization of the Hessian \( \nabla^2 f(x) \).

After whitening one could assume without loss of generality that the vectors \( (a_i)_{i=1,\ldots,m} \in \mathbb{R}^m \) are nearly orthonormal in the first place. Hence the representation (5) would be a near spectral decomposition of the Hessian and the components \( a_i \otimes a_i \) would represent the approximate eigenvectors. However, the numerical stability of spectral decompositions is ensured only under spectral gaps [6,50]. In order to maximally stabilize the approximation of the \( a_i \)'s, one seeks for matrices \( M \in \hat{A} \) with the maximal spectral gap between the first and second largest eigenvalues. This is achieved by the maximizers of the following nonconvex program

\[ M = \arg \max_M \| M \| \quad \text{s.t.} \quad M \in \hat{A}, \quad \| M \|_F \leq 1, \]  

where \( \| \cdot \| \) and \( \| \cdot \|_F \) are the spectral and Frobenius norms, respectively. This program can be solved by a suitable projected gradient ascent, see for instance [23, Algorithm
3.4] and Algorithm 3, and any resulting maximizer has the eigenvector associated to the largest eigenvalue in absolute value close to one of the $a_i$’s. Once approximations $\hat{a}_i$ to all the $a_i$’s are retrieved, it is not difficult to perform the identification of the activation functions $g_i$, see [23, Algorithm 4.1, Theorem 4.1]. The recovery of the network resulting from this algorithmic pipeline is summarized by the following statement.

**Theorem 2** ([23], Theorem 1.2) Let $f$ be a real-valued function defined on the neighborhood of $\Omega = B_1^d$, which takes the form

$$f(x) = \sum_{i=1}^{m} g_i(a_i \cdot x),$$

for $m \leq d$. Let $g_i$ be three times continuously differentiable on a neighborhood of $[-1, 1]$ for all $i = 1, \ldots, m$, and let $\{a_1, \ldots, a_m\}$ be linearly independent. We additionally assume both $J[f]$ and $H[f]$ of maximal rank $m$. Then, for all $\epsilon > 0$ (stepsize employed in the computation of finite differences), using at most $m^X(d+1)+(m+1)(m+2)/2$ random exact point evaluations of $f$, the nonconvex program (6) constructs approximations $\{\hat{a}_1, \ldots, \hat{a}_m\}$ of the weights $\{a_1, \ldots, a_m\}$ up to a sign change for which

$$\left(\sum_{i=1}^{m} \|\hat{a}_i - a_i\|_2^2\right)^{1/2} \lesssim \epsilon,$$

with probability at least $1 - m \exp\left(-\frac{m^{X}c}{2\max\{C_1, C_2\}m}\right)$, for a suitable constant $c > 0$ intervening (together with some fixed power of $m$) in the asymptotical constant of the approximation (7). Moreover, once the weights are retrieved one constructs an approximating function $\hat{f} : B_1^d \to \mathbb{R}$ of the form

$$\hat{f}(x) = \sum_{i=1}^{m} \hat{g}_i(\hat{a}_i \cdot x),$$

such that

$$\|f - \hat{f}\|_{C(\Omega)} \lesssim \epsilon.$$  

While this result have been generalized to the case of passive sampling in [23] and through whitening allows for the identification of nonorthogonal weights, it is restricted to the case of $m \leq d$ and linearly independent weights $\{a_i : i = 1, \ldots, m\}$.

The main goal of this paper is generalizing this approach to account for both the identification of two fully nonlinear hidden layer neural networks and the case where $m > d$ and the weights are not necessarily nearly orthogonal or even linearly independent (see Remark 10 below).

**1.2.3 Deeper Networks: The Two-Layer Case**

What follows further extends the theory discussed in the previous sections to a wider class of functions, namely neural networks with two hidden layers. By doing so, we...
will also address a relevant open problem that was stated in [23], which deals with
the identification of shallow neural networks where the number of neurons is larger
than the input dimension. First, we need a precise definition of the architecture of the
neural networks we intend to consider.

**Definition 3** Let \( 0 < m_1 \leq m_0 \leq d \), \( \{a_1, \ldots, a_{m_0}\} \subset \mathbb{S}^{d-1} \), \( \{b_1, \ldots, b_{m_1}\} \subset \mathbb{S}^{m_0-1} \), and let \( g_1, \ldots, g_{m_0} \) and \( h_1, \ldots, h_{m_1} \) be univariate functions. Denote \( A := (a_1|\ldots|a_{m_0}) \in \mathbb{R}^{d \times m_0} \), \( B := (b_1|\ldots|b_{m_1}) \in \mathbb{R}^{m_0 \times m_1} \), \( G_0 := \text{diag}(g_1'(0), \ldots, g_{m_0}'(0)) \), and assume the following:

(A1) \( g_i'(0) \neq 0 \) for all \( i \in [m_0] \),
(A2) the system \( \{a_1, \ldots, a_{m_0}, v_1, \ldots, v_{m_1}\} \subset \mathbb{S}^d \) with \( v_\ell := \frac{\text{AG}_0 b_\ell}{\|\text{AG}_0 b_\ell\|} \) satisfies a frame condition

\[
\|x\|^2 \leq \sum_{i=1}^{m_0} \langle x, a_i \rangle^2 + \sum_{\ell=1}^{m_1} \langle x, v_\ell \rangle^2 \leq C_F \|x\|^2
\]

for \( 0 < c_f \leq c_F \) and all \( x \in \text{span}\{a_1, \ldots, a_{m_0}\} \).

(A3) the derivatives of \( g_i \) and \( h_\ell \) are uniformly bounded according to

\[
\max_{i=1,\ldots,m_0} \sup_{t \in \mathbb{R}} |g_i^{(k)}(t)| \leq \kappa_k, \quad \text{and} \quad \max_{i=1,\ldots,m_1} \sup_{t \in \mathbb{R}} |h_\ell^{(k)}(t)| \leq \eta_k, \quad k = 0, 1, 2, 3.
\]

Then we define a set of two-layer networks by

\[
\mathcal{F}(d, m_0, m_1) := \left\{ f : \mathbb{R}^d \to \mathbb{R} : f(x) = \sum_{\ell=1}^{m_1} h_\ell \left( \sum_{i=1}^{m_0} b_{i\ell} g_i(\langle a_i^T x \rangle) \right), \text{ (A1)–(A3) are satisfied} \right\},
\]

where \( b_{i\ell} \) is the \((i, \ell)\)-th entry of \( B \), respectively, the \( i \)-th entry of vector \( b_\ell \).

Sometimes it may be convenient to use the more compact writing \( f(x) = 1^T h(B^T g(A^T x)) \) where \( g = (g_1, \ldots, g_{m_0}) \), \( h = (h_1, \ldots, h_{m_1}) \). In the previous section, we presented a dimension reduction that can be applied to one layer neural networks and which can be useful to reduce the dimensionality from the input dimension to the number of neurons of the first layer. The same approach can be applied to networks defined by the class \( \mathcal{F}(d, m_0, m_1) \). For the approximation error of the active subspace, we end up with the following corollary of Theorem 1.

**Corollary 4** (cf. Theorem 1) Assume that \( f \in \mathcal{F}(d, m_0, m_1) \) and let \( \tilde{P}_A \) be constructed as described in Algorithm 1 by sampling \( m_X(d+1) \) values of \( f \). Let \( 0 < s < 1 \) and assume that the \( m_0 \)-th singular value of \( J[f] \) fulfills \( \sigma_{m_0}(J[f]) \geq \alpha > 0 \). Then we have

\[
\|P_A - \tilde{P}_A\|_F \leq \frac{2C_3 \epsilon m_0 m_1}{\sqrt{(1-s)\alpha} - C_3 \epsilon m_0 m_1},
\]
with probability at least $1 - m_0 \exp(-\frac{s^2 m_0 \alpha}{2 C_4 m_1})$ and constants $C_3, C_4 > 0$ that depend only on $\kappa_j, \eta_j$ for $j = 0, \ldots, 3$.

In view of Corollary 4, we can again apply [23, Theorem 1.1] and assume, without loss of generality, that $d = m_0$, for which frame condition (9) automatically implies invertibility of $A$, as the vectors $v_\ell$’s are linear combinations of the $a_i$’s.

2 Approximating the Span of Tensors of Weights

In the one layer case, which was described earlier, the unique identification of the weights is made possible by constructing a matrix space whose rank-1 basis elements are outer products of the weight profiles of the network. This section illustrates the extension of this approach beyond shallow neural networks. Once again, we will make use of differentiation, and overall there will be many parallels to the approach in [23]. However, the intuition behind the matrix space will be less straightforward, because we cannot anymore directly express the second derivative of a two-layer network as a linear combination of symmetric rank-1 matrices. This is due to the fact that the Hessian matrix of a network $f \in \mathcal{F}(m_0, m_0, m_1)$ has the form

$$
\nabla^2 f(x) = \sum_{\ell=1}^{m_1} h'_\ell(b^T_\ell g(A^T x)) \sum_{i=1}^{m_0} b_{i\ell} g''_i(a^T_i x) a_i \otimes a_i + \frac{1}{2} \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} h''_\ell \left(b^T_\ell g(A^T x)\right) b_{i\ell} b_{j\ell} g'_i(a^T_i x) g'_j(a^T_j x)(a_i \otimes a_j + a_j \otimes a_i).
$$

Therefore, $\nabla^2 f(x) \in \text{span} \{a_i \otimes a_j + a_j \otimes a_i \mid i, j = 1, \ldots, m_0\}$, which has dimension $\frac{m_0(m_0+1)}{2}$ and is in general not spanned by symmetric rank-1 matrices. This expression is indeed quite complicated, due to the chain rule and the mixed tensor contributions, which are consequently appearing. At a first look, it would seem impossible to use a similar approach as the one for shallow neural networks recalled in the previous section. Nevertheless a relatively simple algebraic manipulation allows to recognize some useful structure: For a fixed $x \in \mathbb{R}^{m_0}$, we rearrange the expression as

$$
\nabla^2 f(x) = \sum_{\ell=1}^{m_1} h'_\ell(b^T_\ell g(A^T x)) \sum_{i=1}^{m_0} b_{i\ell} g''_i(a^T_i x) a_i \otimes a_i + \frac{1}{2} \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} h''_\ell \left(b^T_\ell g(A^T x)\right) \left[ \sum_{i=1}^{m_0} b_{i\ell} g'_i(a^T_i x) a_i \right] \otimes \left[ \sum_{j=1}^{m_0} b_{j\ell} g'_j(a^T_j x) a_j \right].
$$
which is a combination of symmetric rank-1 matrices since \( \sum_{j=1}^{m_0} b_{j\ell} g_j'(a_{j}^T x) a_j \in \mathbb{R}^{m_0} \). We write the latter expression more compactly by introducing the notation

\[
\nabla^2 f(x) = \sum_{i=1}^{m_0} \gamma_i(x) a_i \otimes a_i + \sum_{\ell=1}^{m_1} \tau_\ell(x) v_\ell(x) \otimes v_\ell(x),
\]

where \( G_x = \text{diag}\left( g_1'(a_1^T x), \ldots, g_{m_0}'(a_{m_0}^T x) \right) \in \mathbb{R}^{m_0 \times m_0} \) and

\[
\begin{align*}
v_\ell(x) &= A G_x b_\ell \in \mathbb{R}^{m_0} & \text{for } \ell \in [m_1], \\
\gamma_i(x) &= g_i''(a_i^T x) \sum_{\ell=1}^{m_1} h_\ell''(b_\ell^T g(A^T x)) b_i \ell \in \mathbb{R} & \text{for } i \in [m_0], \\
\tau_\ell(x) &= \frac{1}{2} h_\ell''(b_\ell^T g(A^T x)) \in \mathbb{R} & \text{for } \ell \in [m_1].
\end{align*}
\]

Let us now introduce the fundamental matrix space

\[
\mathcal{W} = \mathcal{W}(f) := \text{span} \left\{ a_1 \otimes a_1, \ldots, a_{m_0} \otimes a_{m_0}, v_1 \otimes v_1, \ldots, v_{m_1} \otimes v_{m_1} \right\},
\]

where the span is taken over \( \mathbb{R} \), \( a_1, \ldots, a_{m_0} \) are the weight profiles of the first layer, and

\[
\begin{align*}
v_\ell &:= \frac{v_\ell(0)}{\|v_\ell(0)\|_2} = \frac{A G_0 b_\ell}{\|A G_0 b_\ell\|_2} & \text{for } \ell \in [m_1],
\end{align*}
\]

encode “entangled” information between \( A \) and \( B \). For this reason, we call the \( v_\ell \)'s entangled weights. Let us stress at this point that the definition and the constructive
approximation of the space $\mathcal{W}$ is perhaps the most crucial and relevant contribution of this paper. In fact, by inspecting carefully the expression (10), we immediately notice that $\nabla^2 f(0) \in \mathcal{W}$, and also that the first sum in (10), namely $\sum_{i=1}^{m_0} \beta_i(x) a_i \otimes a_i$, lies in $\mathcal{W}$ for all $x \in \mathbb{R}^{m_0}$. Moreover, for arbitrary sampling points $x$, deviations of $\nabla^2 f(x)$ from $\mathcal{W}$ are only due to the second term in (10). The intuition is that for suitably centered distributions of sampling points $x_1, \ldots, x_{m_X}$, with $a_i^T x_i \approx 0$ so that $G_{x_i} \approx G_0$, the Hessians $\{\nabla^2 f(x_i) : i \in [m_X]\}$ are distributed around the space $\mathcal{W}$, see Fig. 1 for a two dimensional sketch of the geometrical situation. Hence, we would attempt an approximation of $\mathcal{W}$ by PCA of a collection of such approximate Hessians. Practically, by active sampling (targeted evaluations of the network $f$) we first construct estimates $\{\hat{\Delta}_1^2 f(x_i) : i \in [m_X]\}$ by finite differences of the Hessian matrices $\{\nabla^2 f(x_i) : i \in [m_X]\}$ (see Sect. 2.1), at sampling points $x_1, \ldots, x_{m_X} \in \mathbb{R}^m$ drawn independently from a suitable distribution $\mu_X$. Next, we define the matrix

$$
\hat{W} = \left( \text{vec}(\Delta_1^2 f(x_1)) | \ldots | \text{vec}(\Delta_1^2 f(x_{m_X})) \right) \in \mathbb{R}^{m_0 \times m_X},
$$

whose columns are the vectorization of the approximate Hessians. Finally, we produce the approximation $\hat{\mathcal{W}}$ to $\mathcal{W}$ as the span of the first $m_0 + m_1$ left singular vectors of the matrix $\hat{W}$, where the choice $m_0 + m_1$ enforces dim($\hat{\mathcal{W}}$) = dim($\mathcal{W}$) = $m_0 + m_1$. The whole procedure of calculating $\hat{\mathcal{W}}$ is given in Algorithm 2. It should be clear that the choice of $\mu_X$ plays a crucial role for the quality of this method. In the analysis that follows, we focus on distributions that are centered and concentrated. Figure 1 helps to form a better geometrical intuition of the result of the procedure. It shows the region covered by the Hessians, indicated by the light blue area, which envelopes the space $\mathcal{W}$ in a sort of nonlinear/nonconvex cone originating from $\nabla^2 f(0)$. In general, the Hessians do not concentrate around $\mathcal{W}$ in a symmetric way, which means that the “center of mass” of the Hessians can never be perfectly aligned with the space $\mathcal{W}$, regardless of the number of samples. In this analogy, the center of mass is equivalent to the space estimated by Algorithm 2, which essentially is a noncentered principal component analysis of observed Hessian matrices. The primary result of this section is Theorem 5, which provides an estimate of the approximation error of Algorithm 2 depending on the sub-Gaussian norm of the sample distribution $\mu_X$ and the number of neurons in the respective layers. More precisely, this result gives a precise worst case estimate of the error caused by the imbalance of mass. For reasons mentioned above, the error does not necessarily vanish with an increasing number of samples, but the probability under which the statement holds will tend to 1. In Fig. 1, the estimated region is illustrated by the gray cones that show the maximal, worst-case deviation of $\mathcal{W}$.

One crucial condition for Theorem 5 to hold is that there exists an $\alpha > 0$ such that

$$
\sigma_{m_0 + m_1} \left( \mathbb{E}_{X \sim \mu_X} \text{vec}(\nabla^2 f(X)) \otimes \text{vec}(\nabla^2 f(X)) \right) \geq \alpha.
$$

This assumption ensures that the space spanned by the observed Hessians has, in expectation, at least dimension $m_0 + m_1$. Aside from this technical aspect this condition implicitly helps to avoid network configurations, which are reducible, for certain weights can not be recovered. For example, we can define a network in $\mathcal{F}(2, 2, 1)$ with...
Algorithm 2: Approximating \( \mathcal{W} \)

\begin{algorithmic}[1]
\Input Neural network \( f \), number of estimated Hessians \( m_X \), step-size of the finite difference approximation \( \epsilon > 0 \), probability distribution \( \mu_X \)
\Begin
\State Draw \( x_1, \ldots, x_m \) independently from \( \mu_X \)
\State Calculate the matrix \( \hat{W} = \left( \text{vec}(\Delta^2 f(x_1)) \right| \ldots | \text{vec}(\Delta^2 f(x_m)) \right) \)
\State Set \( \hat{U} \hat{\Sigma} \hat{V}^T = \text{SVD}(\hat{W}) \)
\State Denote by \( \hat{U}_1 \) the first \( m_0 + m_1 \) columns of \( \hat{U} \).
\State Set \( P_{\hat{W}} := \hat{U}_1 \hat{U}_1^T \)
\End
\Output \( P_{\hat{W}} \)
\end{algorithmic}

weights given by

\[
\begin{align*}
  a_1 &= \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right), \\
  a_2 &= \left( -\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right), \\
  b_1 &= \begin{pmatrix} 1 \\ 0 \end{pmatrix}.
\end{align*}
\]

It is easy to see that \( a_2 \) will never be used during a forward pass through the network, which makes it impossible to recover \( a_2 \) from the output of the network.

In the theorem below and in the proofs that follow, we will make use of the sub-Gaussian norm \( \| \cdot \|_{\psi_2} \) of a random variable. This quantity measures how fast the tails of a distribution decay and such a decay plays an important role in several concentration inequalities. More in general, for \( p \geq 1 \), the \( \psi_p \)-norm of a scalar random variable \( Z \) is defined as

\[
\|Z\|_{\psi_p} = \inf \left\{ t > 0 : E \exp(|Z|/t^p) \leq 2 \right\}.
\]

For a random vector \( X \) on \( \mathbb{R}^d \) the \( \psi_p \)-norm is given by

\[
\|X\|_{\psi_p} = \sup_{x \in S^{d-1}} \|\langle X, x \rangle\|_{\psi_p}.
\]

The random variables for which \( \|X\|_{\psi_1} < \infty \) are called subexponential and those for which \( \|X\|_{\psi_2} < \infty \) are called sub-Gaussian. More in general, the Orlicz space \( L_{\psi_p}(\Omega, \Sigma, P) \) consists of all real random variables \( X \) on the probability space \( (\Omega, \Sigma, P) \) with finite \( \|X\|_{\psi_p} \) norm and its elements are called \( p \)-subexponential random variables. Below, we mainly focus on sub-Gaussian random variables. In particular, every bounded random variable is sub-Gaussian, which covers all the cases we discuss in this work. We refer to [65] for more details. One example of a sub-Gaussian distribution is the uniform distribution on the unit sphere, \( X \sim \text{Unif}(S^{d-1}) \), which has sub-Gaussian norm \( \|X\|_{\psi_2} = \frac{1}{\sqrt{d}} \).

**Theorem 5** Let \( f \in \mathcal{F}(m_0, m_0, m_1) \) be a neural network within the class described in Definition 3 and consider the space \( \mathcal{W} \) as defined in (14). Assume that \( \mu_X \) is a...
probability measure with \( \text{supp}(\mu_X) \subset B_{1}^{m_0} \), \( \mathbb{E}X = 0 \), and that there exists an \( \alpha > 0 \) such that
\[
\sigma_{m_0 + m_1} \left( \mathbb{E}_{X \sim \mu_X} \text{vec}(\nabla^2 f(X)) \otimes \text{vec}(\nabla^2 f(X)) \right) \geq \alpha. \tag{15}
\]

Then, for any \( \epsilon > 0 \), Algorithm 2 returns a projection \( P_{\hat{W}} \) that fulfills
\[
P_{\mathcal{W}^*} - P_{\hat{W}} \|_F \leq \frac{C \epsilon m_1 m_0^3 + C \|A\|^2 \|B\|^2 \|Y\|_2 \sqrt{m_1 \log(m_0 + 1)}}{\sqrt{\alpha^2 - C \alpha m_1 m_0^3}}, \tag{16}
\]
for a suitable subspace \( \mathcal{W}^* \subset \mathcal{W} \) (we can actually assume that \( \mathcal{W}^* = \mathcal{W} \) according to Remark 6 below) with probability at least
\[
1 - 2e^{-cm_0 m_X} - (m_0 + m_1) e^{-\frac{\alpha}{8C_1 \|A\|^2 \|B\|^2 m_1 m_X}}
\]
where \( c > 0 \) is an absolute constant and \( C, C_1, C_\Delta > 0 \) are constants depending on the constants \( \kappa_j, \eta_j \) for \( j = 0, \ldots, 3 \).

**Remark 6** If \( \epsilon > 0 \) is sufficiently small, due to (15) the space \( \hat{W} \) returned by Algorithm 2 has dimension \( m_0 + m_1 \). If the error bound (16) in Theorem 5 is such that \( \|P_{\mathcal{W}^*} - P_{\hat{W}}\|_F < 1 \), then \( \hat{W} \) and \( \mathcal{W}^* \) must have the same dimension. Moreover, \( \mathcal{W}^* \subset \mathcal{W} \) and \( \dim(\mathcal{W}) = m_0 + m_1 \) would necessarily imply that \( \mathcal{W} = \mathcal{W}^* \). Hence, for \( \|P_{\mathcal{W}^*} - P_{\hat{W}}\|_F < 1 \) and \( \epsilon > 0 \) sufficiently small, we have \( \mathcal{W}^* = \mathcal{W} \).

As already mentioned above, for \( \mu_X = \text{Unif}(S^{m_0-1}) \) we have \( \|X\|_{\psi_2} = \frac{1}{\sqrt{m_0}} \). In this case, the error bound 16 behaves like
\[
P_{\mathcal{W}} - P_{\hat{W}} \|_F \leq O \left( \epsilon m_1 m_0^3 + \sqrt{\frac{m_1}{m_0^2}} \log(m_0 + 1) \right),
\]
which is small for \( \epsilon > 0 \) small and \( m_0 \gg m_1 \). The latter condition seems favoring networks, for which the inner layer has a significantly larger number of neurons than the outer layer. This expectation is actually observed numerically, see Sect. 4. We have to add, though, that the parameter \( \alpha > 0 \) that intervenes in the error bound (16) might also depend on \( m_0, m_1 \) (as it is in fact an estimate of an \( (m_0 + m_1)^{th} \) singular value as in (15)). Hence, the dependency on the network dimensions is likely more complex and depends on the interplay between the input distribution \( \mu_X \) and the network architecture. In fact, at least judging from our numerical experiments, the error bound (16) is rather pessimistic, and it certainly describes a worst case analysis. One more reason might be that some crucial estimates in its proof could be significantly improved. Another reason could be the rather great generality of the activation functions of the networks, which we analyze in this paper, as described in Definition 3. Perhaps the specific instances used in the numerical experiments are enjoying better identification properties.
2.1 Estimating Hessians of the Network by Finite Differences

Before addressing the proof of Theorem 5, we give a precise definition of the finite differences we are using to approximate the Hessian matrices. Denote the $i$-th Euclidean canonical basis vector in $\mathbb{R}^d$ by $e_i$ and the second-order finite difference approximation of $\nabla^2 f(x)$ by

$$
\Delta_\epsilon^2 f(x)_{ij} := \frac{f(x + \epsilon e_i + \epsilon e_j) - f(x + \epsilon e_i) - f(x + \epsilon e_j) + f(x)}{\epsilon^2}
$$

for $i, j = 1, \ldots, d = m_0$ and a step-size $\epsilon > 0$.

Lemma 7 Let $f \in \mathcal{F}(m_0, m_0, m_1)$ be a neural network. Further assume that $\Delta_\epsilon^2 f(x)$ is constructed as in (17) for some $\epsilon > 0$. Then we have

$$
\sup_{x \in B_{m_0}^1} \|\nabla^2 f(x) - \Delta_\epsilon^2 f(x)\|_F \leq C_{\Delta} \epsilon m_1 m_0^3,
$$

where $C_{\Delta} > 0$ is a constant depending on the constants $\kappa_j, \eta_j$ for $j = 0, \ldots, 3$.

For the proof of Lemma 7, we simply use the Lipschitz continuity of the functions $g, h$ and of their derivatives, and make use of $\|a\|_2, \|b\|_2 \leq 1$. The details can be found in the Appendix (Sect. 1).

2.2 Span of Tensors of (Entangled) Network Weights: Proof of Theorem 5

The proof can essentially be divided into two separate bounds. Both will be addressed separately with the two lemmas below. For both lemmas, we will assume that $X_1, \ldots, X_{m_X} \sim \mu_X$ independently and that supp $(\mu_X) \subseteq B_{m_0}^1$. Additionally, we define the random matrices

$$
W := (\text{vec}(\nabla^2 f(X_1))| \ldots | \text{vec}(\nabla^2 f(X_{m_X}))),
$$

$$
\hat{W} := (\text{vec}(\Delta^2_\epsilon f(X_1))| \ldots | \text{vec}(\Delta^2_\epsilon f(X_{m_X}))),
$$

$$
W^* := (\text{vec}(P_W \nabla^2 f(X_1))| \ldots | \text{vec}(P_W \nabla^2 f(X_{m_X}))),
$$

where $P_W$ denotes the orthogonal projection onto $W$ (cf. (14)). For reader’s convenience, we recall here from (10) that the Hessian matrix of $f \in \mathcal{F}(m_0, m_0, m_1)$ can be expressed as

$$
\nabla^2 f(x) = \sum_{i=1}^{m_0} \gamma_i(x) a_i \otimes a_i + \sum_{\ell=1}^{m_1} \tau_\ell(x) u_\ell(x) \otimes v_\ell(x),
$$
where $\gamma_i(x)$, $\tau_{\ell}(x)$, and $v_{\ell}(x)$ are introduced in (11)–(13). We further simplify this expression by introducing the notations

$$V_x = \left(v_1(x) | \ldots | v_{m_1}(x)\right) = AG_x B,$$

$$\Gamma_x = \text{diag} \left(\gamma_1(x), \ldots, \gamma_{m_0}(x)\right),$$

$$T_x = \text{diag} \left(\tau_1(x), \ldots, \tau_{m_1}(x)\right),$$

which allow us to rewrite (10) in terms of matrix multiplications

$$\nabla^2 f(x) = A \Gamma_x A^T + V_x T_x V_x^T.$$

Lemma 8 Let $f \in \mathcal{F}(m_0, m_0, m_1)$ and let $\hat{W}$, $W^*$ be defined as in (19)–(20), where $\mu_X$ satisfies $\text{supp} (\mu_X) \subseteq B_1^{m_0}$ and has sub-Gaussian norm $\|X\|_{\psi_2}$. Then the bound

$$\|\hat{W} - W^*\|_F \leq \sqrt{m_X} \left( C_{\Delta} e m_1 m_0^{3/2} + C \|A\|^2 \|B\|^2 \|X\|_{\psi_2} \sqrt{m_1 \log (m_0 + 1)} \right)$$

holds with probability at least $1 - 2 \exp(-c m_1 m_X)$, where $c > 0$ is an absolute constant and $C, C_{\Delta} > 0$ depend only on the constants $\kappa_j, \eta_j$ for $j = 0, \ldots, 3$.

Proof By triangle inequality, we get

$$\|\hat{W} - W^*\|_F \leq \|\hat{W} - W\|_F + \|W^* - W\|_F. \hspace{1cm} (21)$$

For the first term on the right-hand side, we can use the worst-case estimate from Lemma 7 to get

$$\|\hat{W} - W\|_F \leq \sqrt{m_X} \sup_{x \in B_1^{m_0}} \|\Delta^2 f(x) - \nabla^2 f(x)\|_F \leq \sqrt{m_X} C_{\Delta} e m_1 m_0^{3/2} \hspace{1cm} (22)$$

for some constant $C_{\Delta} > 0$. The second term in (21) can be bounded by (the explanation of the individual identities and estimates follows immediately below)

$$\|W - W^*\|_F^2 = \sum_{i=1}^{m_X} \|\text{vec}(\nabla^2 f(X_i)) - \text{vec}(P_{V_i^0} \nabla^2 f(X_i))\|_2^2$$

$$= \sum_{i=1}^{m_X} \left\|V_x T_x V_x^T - V_0 T_x V_0^T\right\|_F^2 \leq \sum_{i=1}^{m_X} 4 \left\|(V_x - V_0) T_x V_x^T\right\|_F^2$$

$$\leq 4 \sum_{i=1}^{m_X} \left\|V_x - V_0\right\|_2 \left\|T_x\right\|_F \left\|V_x\right\|_2$$

$$\leq 4 \|A\|^4 \|B\|^4 \sum_{i=1}^{m_X} \left\|G_{X_i} - G_0\right\|_2 \left\|T_x\right\|_F \left\|G_{X_i}\right\|_2.$$
\[
\leq 4 \|A\|^4 \|B\|^4 m_X \kappa_1^2 \eta_2^2 \sum_{i=1}^{m_X} \|G_{X_i} - G_0\|^2
\leq 4 \kappa_1^2 \kappa_2^2 \eta_2^2 \|A\|^4 \|B\|^4 m_X \sum_{i=1}^{m_X} \|A^T X_i\|_\infty^2.
\]

In the first two equalities, we made use of the fact that \(A \Gamma X A^T \in \mathcal{W}\) and that by definition of an orthogonal projection \(\|V_{X_i} T_{X_i} V_{X_i}^T - P_{\mathcal{W}} V_{X_i} T_{X_i} V_{X_i}^T\|_F \leq \|V_{X_i} T_{X_i} V_{X_i}^T - V_0 T_{X_i} V_0^T\|_F\). The remaining inequalities follow directly from the submultiplicativity of \(\|\cdot\|_F\) and \(\|\cdot\|_\infty\) combined with the Lipschitz continuity of the activation functions and their derivatives (cf. 3 in Definition 3). Since \(\|a_j\| \leq 1\), we can estimate the sub-exponential norm of \(\|A^T X_i\|_\infty^2 = \max_{1 \leq j \leq m_0} \langle X_i, a_j \rangle^2\) by

\[
\|\max_{1 \leq j \leq m_0} \langle X_i, a_j \rangle^2\|_{\psi_1} \leq c_1 \log(m_0 + 1) \max_{1 \leq j \leq m_0} \|\langle X_i, a_j \rangle^2\|_{\psi_1}
= c_1 \log(m_0 + 1) \max_{1 \leq j \leq m_0} \|\langle X_i, a_j \rangle^2\|_{\psi_2} \leq c_1 \log(m_0 + 1) \|X\|_{\psi_2}^2,
\]

for an absolute constant \(c_1 > 0\), where we applied [64, Lemma 2.2.2] in the first inequality and used that \(\|Y\|_{\psi_2}^2 = \|Y\|_{\psi_1}^2\) for any scalar random variable \(Y\) together with the fact that the sub-Gaussian norm of a vector is defined by \(\|X\|_{\psi_2}^2 = \sup_{x \in S^{m-1}} |\langle x, X \rangle|\) (cf. [65]). The random vectors \(X_i \sim \mu_X\) are i.i.d., which allows us to drop the dependency on \(i\) in the last step. The previous bound also guarantees a bound on the expectation, which is due to \(E[|Y|^p] \leq p! \|Y\|_{\psi_1}\) (cf. [64]), namely, for \(p = 1\) and \(Y = \max_{1 \leq j \leq m_0} \langle X, a_j \rangle^2\)

\[
E\left[\max_{1 \leq j \leq m_0} \langle X, a_j \rangle^2\right] \leq \max_{1 \leq j \leq m_0} \langle X, a_j \rangle^2 \leq c_1 \log(m_0 + 1) \|X\|_{\psi_2}^2. \tag{23}
\]

Denote \(Z_i := \|A^T X_i\|_\infty^2\) for all \(i = 1, \ldots, m_X\), then

\[
\left\| W - W^* \right\|_F^2 \leq 4 \kappa_1^2 \kappa_2^2 \eta_2^2 \|A\|^4 \|B\|^4 m_X \sum_{i=1}^{m_X} Z_i. \tag{24}
\]

Therefore, applying the Bernstein inequality for sub-exponential random variables [65, Theorem 2.8.1] to the right sum in (24) yields

\[
\left\| W - W^* \right\|_F^2 \leq 4 \kappa_1^2 \kappa_2^2 \eta_2^2 \|A\|^4 \|B\|^4 (c_1 m_X m_1 \log(m_0 + 1) \|X\|_{\psi_2}^2 + t),
\]
with probability at least

\[
1 - 2 \exp \left( -c \min \left( \frac{t^2}{\sum_{i=1}^{m_X} \|Z_i\|_2^2}, \frac{t}{\max_{i \leq m_X} \|Z_i\|_2} \right) \right) \\
\geq 1 - 2 \exp \left( -c \min \left( \frac{t^2}{m_X (c_1 \log(m_0 + 1) \|X\|_{\psi_2}^2)}, \frac{t}{c_1 \log(m_0 + 1) \|X\|_{\psi_2}^2} \right) \right),
\]

for all \( t \geq 0 \) and an absolute constant \( c > 0 \). Then, by choosing \( t = c_1 m_X m_1 \log(m_0 + 1) \|X\|_{\psi_2} \), we get

\[
\|W - W^*\|_F^2 \leq 8\kappa_1^2 \kappa_2^2 \eta_2^2 \|A\|^4 \|B\|^4 c_1 m_X m_1 \log(m_0 + 1) \|X\|_{\psi_2}^2 \tag{25}
\]

with probability at least

\[
1 - 2 \exp \left( -c \min \left( m_1^2 m_X, m_1 m_X \right) \right) = 1 - 2 \exp \left( -cm_1 m_X \right). \tag{26}
\]

From (21), combining (22) and (25) yields

\[
\|\hat{W} - W^*\|_F \leq \sqrt{m_X} \sup_{x \in B_{m_0}^m} \|\Delta^2 f(x) - \nabla^2 f(x)\|_F \\
+ \sqrt{8c_1 \kappa_1 \kappa_2 \eta_2} \|A\|^2 \|B\|^2 \|X\|_{\psi_2} \sqrt{m_X m_1 \log(m_0 + 1)} \\
\leq \sqrt{m_X C_{\Delta} \epsilon m_1 m_0^{\frac{3}{2}}} \\
+ \sqrt{8c_1 \kappa_1 \kappa_2 \eta_2} \|A\|^2 \|B\|^2 \|X\|_{\psi_2} \sqrt{m_X m_1 \log(m_0 + 1)},
\]

where we used Lemma 7 in the second inequality, and the result holds at least with the probability given as in (26). Setting \( C := \sqrt{8c_1 \kappa_1 \kappa_2 \eta_2} > 0 \) finishes the proof.

**Lemma 9** Let \( \mu_X \) be centered with \( \text{supp} (\mu_X) \subseteq B_{m_0}^m \). Furthermore, assume that \( f \in \mathcal{F}(m_0, m_0, m_1) \) and that \( \hat{W} \) is given by (19) with step-size \( \epsilon > 0 \). If

\[
\sigma_{m_0 + m_1} \left( \mathbb{E} X \sim \mu_X \, \text{vec}(\nabla^2 f(X)) \otimes \text{vec}(\nabla^2 f(X)) \right) \geq \alpha > 0,
\]

then we have

\[
\sigma_{m_0 + m_1} (\hat{W}) \geq \sqrt{m_X} \left( \frac{\alpha}{2} - C_{\Delta} \epsilon m_1 m_0^{\frac{3}{2}} \right),
\]

with probability at least \( 1 - (m_0 + m_1) \exp \left( -\frac{m_X \alpha}{8C_1 \|A\|^2 \|B\|^2 m_1} \right) \), where \( C_{\Delta}, C_1 > 0 \) depend only on the constants \( \kappa_j, \eta_j \) for \( j = 0, \ldots, 3 \).
Proof By Weyl’s inequality and re-using (22), we obtain
\[ \sigma_{m_0 + m_1}(\hat{W}) \geq \sigma_{m_0 + m_1}(W) \geq \| W - \hat{W} \| \geq \sigma_{m_0 + m_1}(W) - C \Delta \epsilon m_1 m_0^\frac{3}{2}. \] (27)

For the first term of the right-hand side, we have \( \sigma_{m_0 + m_1}(W)^2 = \sigma_{m_0 + m_1}(WW^T) \), which can be written as a sum of the outer products of the columns
\[ \sigma_{m_0 + m_1}(WW^T) \geq \sum_{i=1}^{m_X} \text{vec}(\nabla^2 f(X_i)) \otimes \text{vec}(\nabla^2 f(X_i)) \].

Additionally, the matrices \( \text{vec}(\nabla^2 f(X_i)) \otimes \text{vec}(\nabla^2 f(X_i)) \) are independent and positive definite random matrices. The Chernoff bound for the eigenvalues for sums of random matrices, due to Gittens and Tropp [25], applied to the right-hand side of the last equation yields the following lower bound:
\[ \sigma_{m_0 + m_1} \left( \sum_{i=1}^{m_X} \text{vec}(\nabla^2 f(X_i)) \otimes \text{vec}(\nabla^2 f(X_i)) \right) \geq t m_X \alpha \text{ for } t \in [0, 1], \]
with probability at least
\[ 1 - (m_0 + m_1) \exp \left( -(1 - t)^2 \frac{m_X \alpha}{2} \right), \]
where we set \( K = \max_{x \in B_{m_0}^1} \| \text{vec}(\nabla^2 f(x)) \otimes \text{vec}(\nabla^2 f(x)) \| \). To estimate \( K \) more explicitly, we first have to bound the norm of the Hessian matrices. Let \( X \sim \mu_{x} \), then
\[ \| \nabla^2 f(X) \|_F \leq \sup_{x \in B_{m_0}^1} \| \nabla^2 f(x) \|_F = \sup_{x \in B_{m_0}^1} \| A \Gamma_x A^T + V_x T_x V_x^T \|_F \]
\[ \leq \sup_{x \in B_{m_0}^1} \| A \|_F^2 \left( \kappa_2 \| B \|_F \sum_{\ell=1}^{m_1} h'_\ell(b_{\ell}^T g(A^T x + \theta)) + \| B \|_F^2 \kappa_1^2 \| \Gamma_x \|_F \right) \]
\[ \leq \| A \|_F^2 \left( \kappa_2 \| B \|_F \eta_2 \sqrt{m_1} + \| B \|_F^2 \kappa_1^2 \eta_2 \sqrt{m_1} \right) \leq \sqrt{C_1} \| A \|_F^2 \| B \|_F^2 \sqrt{m_1}, \]
for some constant \( C_1 > 0 \). Now we can further estimate \( K \) by
\[ K = \max_{x \in B_{m_0}^1} \| \text{vec}(\nabla^2 f(x)) \otimes \text{vec}(\nabla^2 f(x)) \| = \max_{x \in B_{m_0}^1} \| \text{vec}(\nabla^2 f(x)) \otimes \text{vec}(\nabla^2 f(x)) \|_F \]
\[ = \max_{x \in B_{m_0}^1} \| \nabla^2 f(x) \|_F^2 \leq C_1 \| A \|_F^4 \| B \|_F^4 m_1. \]

Finally, we can finish the proof by plugging the above into (27) and by setting \( t = \frac{1}{2} \).
Proof of Theorem 5} The proof is a combination of the previous lemmas together with an application of Wedin’s bound [59,66]. Given \( \hat{W}, W^* \), let \( \hat{U} \hat{\Sigma} \hat{V}^T, U^* \Sigma^* V^* \) be their respective singular value decompositions. Furthermore, denote by \( U_1, U_1^* \) the matrices formed by only the first \( m_0 + m_1 \) columns of \( \hat{U}, U^* \), respectively. According to this notation, Algorithm 2 returns the orthogonal projection \( P_{\hat{W}} = \hat{U}_1 \hat{U}_1^T \). We also denote by \( P_{W^*} \) the projection given by \( P_{W^*} = U_1^* U_1^{*T} \). Then we can bound the difference of the projections by applying Wedin’s bound

\[
\| P_{\hat{W}} - P_{W^*} \|_F = \| \hat{U}_1 \hat{U}_1^T - U_1^* U_1^{*T} \|_F \leq \frac{2 \| \hat{W} - W^* \|_F}{\tilde{\alpha}}.
\]

as soon as \( \tilde{\alpha} > 0 \) satisfies

\[
\tilde{\alpha} \leq \min_{1 < j \leq m_0 + m_1} \left| \sigma_j(\hat{W}) - \sigma_k(W^*) \right| \quad \text{and} \quad \tilde{\alpha} \leq \min_{1 \leq j \leq m_0 + m_1} \sigma_j(\hat{W}).
\]

Since \( \mathcal{W} \) has dimension \( m_0 + m_1 \), we have \( \max_{k \geq m_0 + m_1 + 1} \sigma_k(W^*) = 0 \). Therefore the second inequality is equivalent to the first, and we can choose \( \tilde{\alpha} = \sigma_{m_0+m_1}(\hat{W}) \leq \min_{1 \leq j \leq m_0 + m_1} \sigma_j(\hat{W}) \). Thus, we end up with the inequality

\[
\| P_{\hat{W}} - P_{W^*} \|_F \leq \frac{2 \| \hat{W} - W^* \|_F}{\sigma_{m_0+m_1}(\hat{W})}.
\]

Applying the union bound for the two events in Lemma 8 and Lemma 9 in combination with the respective inequalities yields

\[
\| P_{\hat{W}} - P_{W^*} \|_F \leq \frac{C \Delta \epsilon m_1 m_0^2 + C \| A \|^2 \| B \|^2 \| X \|_2 \sqrt{m_1 \log(m_0 + 1)}}{\sqrt{\alpha} - C \Delta \epsilon m_1 m_0^2}
\]

with probability at least \( 1 - 2e^{-cm_1m_1X} - (m_0 + m_1)e^{-\frac{c}{8C1\|A\|^2\|B\|^3m_1}m_1X} \), where \( C, C_1, C_\Delta, c > 0 \) are the constants from the lemmas above.

3 Recovery of Individual (Entangled) Neural Network Weights

The symmetric rank-1 matrices \( \{a_i \otimes a_i : i \in [m_0]\} \cup \{v_\ell \otimes v_\ell : \ell \in [m_1]\} \) made of tensors of (entangled) neural network weights are the spanning elements of \( \mathcal{W} \), which in turn can be approximated by \( \hat{\mathcal{W}} \) as has been proved above. In this section, we explain under which conditions it is possible to stably identify approximations to the network profiles \( \{a_i : i \in [m_0]\} \cup \{v_\ell : \ell \in [m_1]\} \) by a suitable selection process, Algorithm 3.

To simplify notation, we drop the differentiation between weights \( a_i \) and \( v_\ell \) and simply denote \( \mathcal{W} = \text{span} \{w_1 \otimes w_1, \ldots, w_m \otimes w_m\} \), where \( m = m_0 + m_1 \), and

\( \hat{\mathcal{W}} \) Springer
every $w_\ell$ equals either one of the $a_i$'s or one of the $v_\ell$'s. Thus, $m$ may be larger than $d$. We also use the notations $W_j := w_j \otimes w_j$, and $\hat{W}_j := P_{\hat{W}}(W_j)$. Provided that the approximation error $\delta := \| P_W - P_{\hat{W}} \|_F$ satisfies $\delta < 1$ (cf. Theorem 5), $\{ \hat{W}_j : j \in [m] \}$ is the image of a basis under a bijective map and thus can be used as a basis for $\hat{W}$ (see Lemma 32 in the Appendix). We quantify the deviation from orthonormality by $\nu := C_F - 1$, see (9). As an example of suitable frames, normalized tight frames achieve the bounds $c_f = C_F = m/d$ [4, Theorem 3.1], see also [11].

Finally, for any real, symmetric matrix $X$, we let $X = \sum_{j=1}^{d} \lambda_j(X) u_j(X) \otimes u_j(X)$ be the spectral decomposition ordered according to $\|X\| = \lambda_1(X) \geq \ldots \geq \lambda_d(X)$ (in case $\lambda_1(X) = -\|X\|$, we actually consider $-X$ instead of $X$). In the following, we are able to provide in Theorem 11 general recovery guarantees of network weights provided by the eigenvector associated to the largest eigenvalue in absolute value of any suitable matrix $M \in \hat{W} \cap S$.

**Remark 10** The problem considered in this section is how to approximate the individual $w_\ell \otimes w_\ell$ within the space $W$ or more precisely by using its approximation $\hat{W}$. As the analysis is completely unaware of how the space $\hat{W}$ has been constructed, in particular it does not rely on the fact that it comes from second-order differentiation of a two hidden layer network, here we are actually implicitly able of addressing also the problem of the identification of weights for one hidden layer networks (2) with a number $m$ of neurons larger than the input dimension $d$, which was left as an open problem from [23].

### 3.1 Recovery Guarantees

The network profiles $\{w_j, j \in [m]\}$ are (up to sign) uniquely defined by matrices $\{W_j : j \in [m]\}$ as they are precisely the eigenvectors corresponding to the unique nonzero eigenvalue. Therefore, it suffices to recover $\{W_j : j \in [m]\}$, and we have to study when such matrices can be uniquely characterized within the matrix space $W$ by their rank-1 property. Let us stress that this problem is strongly related to similar and very relevant ones appearing recently in the literature addressing nonconvex programs to identify sparse vectors and low-rank matrices in linear subspaces, see, e.g., in [45,49]. In Appendix 2 (Lemma 30 and Corollary 31), we prove that unique identification is possible if any subset of $\lceil m/2 \rceil + 1$ vectors of $\{w_j : j \in [m]\}$ is linearly independent and that such subset linear independence is actually implied by the frame bounds (9) if $\nu = C_F - 1 < \lceil m/2 \rceil ^{-1}$. Unfortunately, this assumption seems a bit too restrictive in our scenario; hence, we instead resort to a weaker and robust version given by the
following result. In particular, we prove that any near rank-1 matrix in $\hat{W}$ of unit Frobenius norm is not too far from one of the $W_j$'s, provided that $\delta$ and $v$ are small.

**Theorem 11** Let $M \in \hat{W} \cap S$ and assume $\max\{\delta, v\} \leq 1/4$. If $\lambda_1(M) > \max\{2\delta, \lambda_2(M)\}$ then

$$\min_{j=1,\ldots,m, s \in \{-1, 1\}} \|s w_j - u_1(M)\| \leq \sqrt{2} \frac{\sqrt{\nu + 2\delta}}{\lambda_1(M) - \lambda_2(M)}.$$  \hspace{1cm} (28)

Before proving Theorem 11, we need the following technical result.

**Lemma 12** For any $M = \sum_{j=1}^m \sigma_j \hat{W}_j \in \hat{W} \cap S$ with $\lambda_1(M) \geq \delta/(1 - \delta)$ we have $\max_i \sigma_i \geq 0$.

**Proof** Assume, to the contrary, that $\max_j \sigma_j < 0$, and denote $Z = \sum_{j=1}^m \sigma_j W_j$ with $M = P_{\hat{W}}(Z)$. $Z$ is negative definite, since $\nu^T Z v = \sum_{j=1}^m \sigma_j \langle w_j, v \rangle^2$, and $\sigma_j < 0$ for all $i = 1, \ldots, m$. Moreover, we have $\|Z\|_F \leq (1 - \delta)^{-1}$ by Lemma 32, and thus, we get a contradiction by

$$\frac{\delta}{1 - \delta} \leq \lambda_1(M) \leq \lambda_1(Z) + \|M - Z\|_F < \|M - Z\|_F \leq \frac{\delta}{1 - \delta}.$$  \hspace{1cm} (29)

**Proof of Theorem 11** Let $\lambda_1 := \lambda_1(M)$, $u_1 := u_1(M)$ for short in this proof. We can represent $M$ in terms of the basis elements of $\hat{W}$ as $M = \sum_{j=1}^m \sigma_j \hat{W}_j$, and let $Z \in W$ satisfy $M = P_{\hat{W}}(Z)$. Furthermore, let $\sigma_{j^*} = \max_j \sigma_j \geq 0$ where the nonnegativity follows from Lemma 12. Using $Z = \sum_{j=1}^m \sigma_j w_j \otimes w_j$ and $\|Z\|_F \leq (1 - \delta)^{-1}$, we first notice that

$$\begin{align*}
\lambda_1 &= \langle M, u_1 \otimes u_1 \rangle = \langle Z, u_1 \otimes u_1 \rangle + \langle M - Z, u_1 \otimes u_1 \rangle \\
&\leq \sum_{j=1}^m \sigma_j \langle w_j, u_1 \rangle^2 + \|M - Z\|_F \leq \sigma_{j^*} C_F + 2\delta \leq \sigma_{j^*} + v + 2\delta,
\end{align*}$$  \hspace{1cm} (29)

and

$$\begin{align*}
\lambda_1 &= \langle M, u_1 \otimes u_1 \rangle \geq \max_j \{\langle Z, w_j \otimes w_j \rangle - 2\delta \geq \sigma_{j^*} + \sum_{i \neq j^*} \sigma_i \langle w_i, u_1 \rangle^2 - 2\delta \\
&\geq \sigma_{j^*} - \|\sigma\|_{\infty} v - 2\delta \geq \sigma_{j^*} - 2v - 2\delta,
\end{align*}$$

where we used $\|\sigma\|_{\infty} \leq (1 - \delta)^{-1} (1 - v)^{-1} \leq 2$ according to Lemma 33. Hence $|\lambda_1 - \sigma_{j^*}| \leq 2\delta + 2v$. Define now $Q := \text{Id} - u_1 \otimes u_1$. Choosing $s \in (-1, 1)$ so that $s\{w_{j^*}, u_1\} \geq 0$, we can bound the left hand side in (28) by

$$\|s w_{j^*} - u_1\|^2 = 2(1 - \langle s w_{j^*}, u_1 \rangle) \leq 2(1 - \langle w_{j^*}, u_1 \rangle^2) = 2 \|Q w_{j^*}\|^2 = 2 \|Q W_{j^*}\|_F^2.$$  \hspace{1cm} (29)
Viewing $W_{j*} = w_{j*} \otimes w_{j*}$ as the orthogonal projection onto the eigenspace of the matrix $\lambda_1 W_{j*}$, corresponding to eigenvalues in $[\infty, \lambda_1]$, we can use Davis-Kahans Theorem in the version of [6, Theorem 7.3.1] to further obtain

$$\| s w_{j*} - u_1(M) \| \leq \sqrt{2} \frac{\| Q W_{j*} \|_F}{\lambda_1 - \lambda_2} \leq \sqrt{2} \frac{\| Q (\lambda_1 W_{j*} - M) W_{j*} \|_F}{\lambda_1 - \lambda_2}.$$

(30)

To bound the numerator, we first use $\| Z - M \|_F \leq \delta/(1 - \delta)$ in the decomposition

$$\| (\lambda_1 W_{j*} - M) W_{j*} \|_F \leq \| (\lambda_1 W_{j*} - Z) W_{j*} \|_F + \| Z - M \|_F \leq \| (\lambda_1 W_{j*} - Z) W_{j*} \|_F + \frac{\delta}{1 - \delta},$$

and then bound the first term using $|\lambda_1 - \sigma_{j*}| \leq 2\delta + 2\nu$ and the frame property (9) by

$$\| (\lambda_1 W_{j*} - Z) W_{j*} \|_F = \| (\lambda_1 - \sigma_{j*}) W_{j*} + \sum_{j \neq j*} \sigma_j (w_j \otimes w_j) W_{j*} \|_F \leq \| \lambda_1 - \sigma_{j*} \| + \sum_{j \neq j*} |\sigma_j| |\{w_{j*}, w_j\}| \leq 2\delta + 2\nu + \| \sigma \|_2 \sqrt{\sum_{j \neq j*} (w_{j*}, w_j)^2} \leq 2\delta + 2\nu + \| \sigma \|_2 \sqrt{\nu}.$$

Combining these estimates with (30) and $\delta/(1 - \delta) \leq 2\delta$, we obtain

$$\| s w_{j*} - u_1(M) \| \leq \sqrt{2} \frac{2\delta + 2\nu + \| \sigma \|_2 \sqrt{\nu} + 2\delta}{\lambda_1 - \lambda_2} = \sqrt{2} \frac{\| \sigma \|_2 \sqrt{\nu} + 2\nu + 4\delta}{\lambda_1 - \lambda_2}.$$ 

The result follows since $\{w_j \otimes w_j : j \in [m]\}$ is a Riesz basis and thus $\| \sigma \|_2 \leq c_r^{-1/2} \| Z \|_F \leq 2c_r^{-1/2}$.

The preceding result provides recovery guarantees for network weights provided by the eigenvector associated to the largest eigenvalue in absolute value of any suitable matrix $M \in \hat{W} \cap S$. The estimate is inversely proportional to the spectral gap $\lambda_1(M) - \lambda_2(M)$. The problem then becomes the constructive identification of matrices $M$ belonging to $\hat{W} \cap S$, which simultaneously maximize the spectral gap. Inspired by the results in [23], we propose to consider the following nonconvex program as
selector of such matrices

\[ M = \arg \max \|M\| \quad \text{s.t.} \quad M \in \hat{W}, \quad \|M\|_F \leq 1. \tag{31} \]

By maximizing the spectral norm under a Frobenius norm constraint, a local maximizer of the program should be as nearly rank one as possible within a given neighborhood. Moreover, if rank one matrices exist in \( \hat{W} \), these are precisely the global optimizers.

### 3.2 A Nonlinear Program: Properties of Local Maximizers of (31)

In this section, we prove that, except for spurious cases, local maximizers of (31) are generically almost rank-1 matrices in \( \hat{W} \). In particular, we show that local maximizers either satisfy

\[ \|M\|_2 \geq 1 - c\delta - c'\nu, \]

for some small constants \( c, c' \), implying near-minimal rankness, or

\[ \|M\|_2 \leq c\delta + c'\nu, \]

i.e., all eigenvalues of \( M \) are small, the mentioned spurious cases. Before addressing these estimates, we provide a characterization of the first- and second-order optimality conditions for (31), see [23] and also [61,62].

**Theorem 13** (Theorem 3.4 in [23]) Let \( M \in \hat{W} \cap \mathbb{S} \) and assume there exists a unique \( i^* \in [d] \) satisfying \( |\lambda_{i^*}(M)| = \|M\| \). If \( M \) is a local maximizer (31); then, it fulfills the stationary or first order optimality condition

\[ u_{i^*}(M)^T X u_{i^*}(M) = \lambda_{i^*}(M) \langle X, M \rangle \tag{32} \]

for all \( X \in \hat{W} \). A stationary point \( M \) (in the sense that \( M \) fulfills (32)) is a local maximizer of (31) if and only if for all \( X \in \hat{W} \)

\[ 2 \sum_{k \neq i^*} \frac{(u_{i^*}(M)^T X u_k(M))^2}{|\lambda_{i^*}(M) - \lambda_k(M)|} \leq |\lambda_{i^*}(M)| \|X - \langle X, M \rangle M\|_F^2. \tag{33} \]

**Proof** The statement requires minor modification of [23, Theorem 3.4] and the proof follows along analogous lines. For reader’s convenience, we give self-contained proof of the statement below, with some key computations borrowed from [23].

For simplicity, we drop the argument \( M \) in \( \lambda_i, u_i \), and without loss of generality we assume \( \lambda_{i^*} = \|M\| \), otherwise we consider \(-M\). Following the analysis in [23], for \( X \in \hat{W} \cap \mathbb{S} \) we can consider the function

\[ f_X(\alpha) = \frac{\|M + \alpha X\|}{\|M + \alpha X\|_F}, \]

because \( M \) is a local maximizer if and only if \( \alpha = 0 \) is a local maximizer of \( f_X \) for all \( X \in \hat{W} \cap \mathbb{S} \).

Let us consider \( X \in \hat{W} \cap \mathbb{S} \) with \( X \perp M \) first. We note that the simplicity of \( \lambda_{i^*} \) implies that there exist analytic functions \( \lambda_{i^*}(\alpha) \) and \( u_{i^*}(\alpha) \) with \( (M + \alpha X)u_{i^*}(\alpha) = \lambda_{i^*}(\alpha)u_{i^*}(\alpha) \) for all \( \alpha \) in a neighborhood around 0 [40,50]. Therefore we can use a
Taylor expansion $\|M + \alpha X\| = \lambda_{i^*} + \lambda'_{i^*}(0)\alpha + \lambda''_{i^*}(0)\alpha^2/2 + O(\alpha^3)$ and combine it with $\|M + \alpha X\|_F = \sqrt{1 + \alpha^2} = 1 - \alpha^2/2 + O(\alpha^4)$ to get

$$f_X(\alpha) = \left(1 - \alpha^2/2\right)\left(\lambda_{i^*} + \lambda'_{i^*}(0)\alpha + \lambda''_{i^*}(0)\alpha^2/2\right) + O(\alpha^3) \quad \text{as} \quad \alpha \to 0.$$ 

Differentiating once, we get $f'_X(0) = \lambda'_{i^*}(0)$; hence, $\alpha = 0$ is a stationary point if and only if $\lambda'_{i^*}(0)$ vanishes. Following the computations in [23], we find that $\lambda'_{i^*}(0) = u_{i^*}(0)^T X u_{i^*}(0) = 0$, and thus, (32) follows for any $X \perp M$. For general $X$, we split $X = \langle X, M \rangle M + X_\perp$, and get $u_{i^*}(0)^T X u_{i^*}(0) = \langle X, M \rangle u_{i^*}(0)^T M u_{i^*}(0) = \lambda_{i^*}(0) \langle X, M \rangle$.

For (33), we have to check additionally $f''_X(\alpha) \leq 0$. The second derivative of $f_X(\alpha)$ at zero is given by $f''_X(0) = \lambda''_{i^*}(0) - \lambda_{i^*}(0)$; hence, the condition for attaining a local maximum is $\lambda''_{i^*}(0) \leq \lambda_{i^*}(0)$. Again, we can follow the computations in [23] to obtain

$$\lambda''_{i^*}(0) = 2 \sum_{k \neq i^*} \frac{(u_{i^*}^T(0) X u_k(0))^2}{|\lambda_{i^*}(0) - \lambda_k(0)|},$$

and (33) follows immediately for any $X \perp M$, $\|X\|_F = 1$. For general $X$, we decompose it into $X = \langle X, M \rangle M + X_\perp$. Since $u_{i^*}^T(0) M u_k(0) = 0$ for all $k \neq i^*$, we get

$$2 \sum_{k \neq i^*} \frac{(u_{i^*}^T(0) \langle X, M \rangle M + X_\perp u_k(0))^2}{|\lambda_{i^*}(0) - \lambda_k(0)|} \leq \lambda_{i^*}(0) \|X_\perp\|^2_F,$$

and the result follows from $\|X_\perp\|_F = \|X - \langle X, M \rangle M\|_F$.

For simplicity, we denote $u_i := u_i(M)$ and $\lambda_i = \lambda_i(M)$ throughout the rest of this section. Moreover, we assume $M$ satisfies

(A1) $\lambda_1 = \|M\|$ (this is without loss of generality because $-M$ and $M$ may be both local maximizers),

(A2) $\lambda_1 > \lambda_2$. (This is a useful technical condition in order to use the second-order optimality condition (33).)

To derive the bounds for $\lambda_1$, we establish an inequality $0 \leq \lambda_1^2(\lambda_1^2 - 1) + c\delta + c'\nu$, which implies that $\lambda_1^2(M)$ is either close to 0 or close to 1. A first ingredient for obtaining the inequality is

$$\|\hat{W}_j u_1\|^2_2 \geq u_1^T \hat{W}_j u_1 - 2\delta = \lambda_1 \langle \hat{W}_j, M \rangle - 2\delta,$$  \quad (34)
where we used $|\|\hat{W}_j u_1\| - u_1^T \hat{W}_j u_1|$ $\leq 2\delta$ in the inequality, see Lemma 33 in Appendix 2, and (32) in the equality. The other useful technical estimate is provided in the following Lemma, which is proven by leveraging the second order optimality condition (33).

**Lemma 14** Assume that $M$ is a local maximizer satisfying (A1) and (A2) and let $\max\{\delta, \nu\} < 1/4$. For any $X \in \hat{W}$ with $\|X\|_F \leq 1$ we have

$$\|Xu_1\|_2^2 \leq \frac{1}{2} \left( \lambda_1^2 + \langle X, M \rangle^2 \right) + 5\delta + 2\nu. \quad (35)$$

For the proof of Lemma 14, we need a lower bound for the smallest eigenvalue (see Appendix 2 for the proof of Lemma 15).

**Lemma 15** Assume that $M$ is a stationary point of (31) satisfying (A1) and (A2). If $\max\{\delta, \nu\} < 1/4$, then $\lambda_D \geq -2\delta\lambda_1^{-1} - 8\delta - 4\nu$.

**Proof of Lemma 14** We first use (32) and (33) to get

$$\frac{2}{\lambda_1 - \lambda_D} \left( \|Xu_1\|_2^2 - \lambda_1^2 \langle X, M \rangle^2 \right) = \frac{2}{\lambda_1 - \lambda_D} \left( \|Xu_1\|_2^2 - \langle Xu_1, u_1 \rangle^2 \right) = \frac{2}{\lambda_1 - \lambda_D} \sum_{i=2}^{D} \langle Xu_1, u_k \rangle^2 \leq 2 \sum_{i=2}^{D} \frac{(u_1^T Xu_k)^2}{\lambda_1 - \lambda_k} \leq \lambda_1 \|X - \langle X, M \rangle M\|_F^2,$$

and then rearrange the inequality to obtain

$$\|Xu_1\|_2^2 \leq \frac{\lambda_1 (\lambda_1 - \lambda_D)}{2} \left( \|X\|_F^2 - \langle X, M \rangle^2 \right) + \lambda_1^2 \langle X, M \rangle^2 \leq \frac{\lambda_1 (\lambda_1 - \lambda_D)}{2} + \frac{\lambda_1 (\lambda_1 + \lambda_D)}{2} \langle X, M \rangle^2 = \lambda_1^2 \frac{1}{2} + \frac{\lambda_1 (\lambda_1 + \lambda_D)}{2} \langle X, M \rangle^2 - \lambda_1 \lambda_D \frac{1}{2} - \langle X, M \rangle^2.$$

Using the lower bound for $\lambda_D$ from Lemma 15, and $\lambda_1 \leq 1$, we get

$$\|Xu_1\|_2^2 \leq \frac{\lambda_1^2}{2} \left( 1 + \frac{\langle X, M \rangle^2}{2} \right) + \lambda_1 \left( 2\delta\lambda_1^{-1} + 8\delta + 4\nu \right) \frac{1 - \langle X, M \rangle^2}{2} \leq \frac{\lambda_1^2}{2} + (10\delta + 4\nu) \frac{1 - \langle X, M \rangle^2}{2} = \lambda_1^2 \frac{1}{2} + \frac{\langle X, M \rangle^2}{2} + 5\delta + 2\nu.$$

By combining (34) and (35), the bounds for $\lambda_1$ follow.
Theorem 16 Assume that $M$ is a local maximizer of (31) satisfying 3.2 and 3.2, and assume $38\delta + 13\nu < 1/4$. Then we have $\lambda_1^2 \geq 1 - 38\delta - 13\nu$ or $\lambda_1^2 \leq 38\delta + 13\nu$.

Proof Let $j^* = \arg\max_j \sigma_j$. We first note that we can assume $\sigma_{j^*} \geq 0$ without loss of generality by Lemma 12, since there is nothing to show if $\lambda_1 \leq 2\delta$. Now we consider (34) and (35) for $\hat{X} = \hat{W}_{j^*}$ to get the inequality

$$\lambda_1^2 \geq \frac{1}{2} \left[ 1 + \langle \hat{W}_{j^*}, M \rangle \right] + 5\delta + 2\nu \geq \lambda_1 \left( \langle \hat{W}_{j^*}, M \rangle - 2\delta \right),$$

or, equivalently, $0 \leq \lambda_1^2 - 1 + \left( 1 - \lambda_1 \langle \hat{W}_{j^*}, M \rangle \right)^2 + 14\delta + 4\nu$

or, equivalently, $0 \leq \lambda_1^2 - 1 + \left( 1 - \lambda_1 \sigma_{j^*} \| \hat{W}_{j^*} \|^2_F + \lambda_1 \left( \sigma_{j^*} \| \hat{W}_{j^*} \|^2_F - \langle \hat{W}_{j^*}, M \rangle \right) \right)^2$

$$+ 14\delta + 4\nu.$$  

(36)

We separate two cases. In the first case, we have $\sigma_{j^*} > 1$, which implies $\langle \hat{W}_{j^*}, M \rangle > 1 - 5\delta - 2\nu$ and thus $\langle \hat{W}_{j}, M \rangle > 1 - 6\delta - 2\nu$ by Lemma 33 and $\max \{ \delta, \nu \} < 1/4$. Since $\langle \hat{W}_{j}, M \rangle = w_j^T M w_j$, this implies $\lambda_1 > 1 - 6\delta - 2\nu$, i.e., the result is proven.

We continue with the case $\sigma_{j^*} \leq 1$, which implies $\lambda_1 \sigma_{j^*} \| \hat{W}_{j^*} \|^2_F \leq 1$. Using Lemma 33 to bound $\sigma_{j^*} \| \hat{W}_{j} \|^2_F - (\langle \hat{W}_{j^*}, M \rangle, \lambda_1 < 1$ and $\| \hat{W}_{j} \|^2_F \geq 1 - 2\delta$, the last inequality in (36) implies

$$0 \leq \lambda_1^2 - 1 + \left( 1 - \lambda_1 \sigma_{j^*} + 6\delta + 2\nu \right)^2 + 14\delta + 4\nu.$$  

(37)

Furthermore, by following the computation we performed for (29), we get $\sigma_{j^*} \geq \lambda_1 - \nu - 2\delta$, and inserting it in (37) we obtain

$$0 \leq \lambda_1^2 - 1 + \left( 1 - \lambda_1^2 + 8\delta + 3\nu \right)^2 + 14\delta + 4\nu,$$

implying $0 \leq \lambda_1^2 \left( \lambda_1^2 - 1 \right) + 38\delta + 13\nu$.

Provided that $38\delta + 13\nu < 1/4$, this quadratic inequality (in the unknown $\lambda_1^2$) has solutions $\lambda_1^2 \geq 1 - 38\delta - 13\nu$, or $\lambda_1^2 \leq 38\delta + 13\nu$.

In Sect. 3.2, we analyze local maximizers of (31) and show that there exist small constants $c, c'$ such that either $\| M \|^2 \geq 1 - c\delta + c'\nu$, or $\| M \|^2 \leq c\delta + c'\nu$. Therefore, a local maximizer of (31) is either almost rank-1 or it has its energy distributed across many eigenvalues. This criterion can be easily checked in practice, and therefore maximizing (31) is a suitable approach for finding near rank-1 matrices in $\hat{V}$. In this section, we show how those individual symmetric rank-1 tensors can be approximated by a simple iterative algorithm, Algorithm 3, making exclusive use of the projection $P_{\hat{V}}$. Algorithm 3 strives to solve the nonconvex program (31), by iteratively increasing the spectral norm of its iterations. Our approach is closely related to the projected gradient ascent iteration [23, Algorithm 4.1], but we introduce some modifications, in particular we exchange the order of the normalization and the projection onto $\hat{V}$. The proof of convergence of [23, Algorithm 4.1] takes advantage of that different ordering.
of these operations to address the case where $\mathcal{W}$ is spanned by at most $m \leq d$ rank-1 matrices formed as tensors of nearly orthonormal vectors (after whitening). In fact, its analysis is heavily based on approximated singular value or spectral decompositions. Unfortunately in our case, the decomposition $M = \sum_{j=1}^{m} \sigma_j w_j \otimes w_j$ does not approximate the singular value or spectral decomposition since the $w_j$’s are redundant (they form a frame) and therefore are not properly nearly orthonormal in the sense required in [23].

Algorithm 3 is based on the iterative application of the operator $F_\gamma$ defined by

$$F_\gamma(X) := P_\mathbb{S} \circ P_{\hat{\mathcal{W}}}(X + \gamma u_1(X) \otimes u_1(X)),$$

with $\gamma > 0$ and $P_\mathbb{S}$ as the projection onto the sphere $\mathbb{S} = \{X : \|X\|_F = 1\}$. The following Lemma shows that, if $\lambda_1(X) > 0$, the operator $F_\gamma$ is well-defined, in the sense that it is a single-valued operator.

**Lemma 17** Let $X \in \hat{\mathcal{W}} \cap \mathbb{S}$ with $\lambda_1(X) > 0$ and $\gamma > 0$. Then $\|P_{\hat{\mathcal{W}}}(X + \gamma u_1(X) \otimes u_1(X))\|_F^2 = 1 + 2\gamma \lambda_1(X) + \gamma^2 \|P_{\hat{\mathcal{W}}}(u_1(X) \otimes u_1(X))\|_F^2$. In particular, $F_\gamma(X)$ is well-defined and can be explicitly expressed as

$$F_\gamma(X) = \frac{P_{\hat{\mathcal{W}}}(X + \gamma u_1(X) \otimes u_1(X))}{\|P_{\hat{\mathcal{W}}}(X + \gamma u_1(X) \otimes u_1(X))\|_F}.$$

**Proof** The result follows from $\langle X, P_{\hat{\mathcal{W}}}(u_1(X) \otimes u_1(X)) \rangle = \lambda_1(X)$ and computing explicitly the squared norm $\|P_{\hat{\mathcal{W}}}(X + \gamma u_1(X) \otimes u_1(X))\|_F^2$.

We analyze next the sequence $(M_j)_{j \in \mathbb{N}}$ generated by Algorithm 3. We show that $(\lambda_1(M_j))_{j \in \mathbb{N}}$ is a strictly monotone increasing sequence, converging to a well-defined limit $\lambda_\infty = \lim_{j \to \infty} \lambda_1(M_j)$, and, if $\lambda_1(M_j) > 1/\sqrt{2}$ for some $j$, all convergent subsequences of $(M_j)_{j \in \mathbb{N}}$ converge to fixed points of $F_\gamma$. Moreover, we prove that such fixed points satisfy (32) and are thus stationary points of (31). We begin by providing two equivalent characterizations of (32).

**Lemma 18** For $M \in \hat{\mathcal{W}}$ and $c \neq 0$, we have

$$v^T X v = c \langle X, M \rangle \quad \text{for all } X \in \hat{\mathcal{W}} \quad \text{if and only if} \quad M = c^{-1} P_{\hat{\mathcal{W}}}(v \otimes v).$$
Lemma 19 Let $X \in \hat{\mathcal{V}} \cap \mathbb{S}$. We have $\| P_{\hat{\mathcal{V}}} (u_j \otimes u_j) \|_F \geq | \lambda_j (X) |$ with equality if and only if $X = \lambda_j (X)^{-1} P_{\hat{\mathcal{V}}} (u_j \otimes u_j)$. 

**Proof** We drop the argument $X$ for $\lambda_j (X)$ and $u_j (X)$ for simplicity. We first calculate

$$
\| P_{\hat{\mathcal{V}}} (u_j \otimes u_j) \|_F = \| P_{\hat{\mathcal{V}}} (u_j \otimes u_j) \|_F \| X \|_F \geq | \langle P_{\hat{\mathcal{V}}} (u_j \otimes u_j), X \rangle |
$$

Moreover, we have equality if and only if $\| P_{\hat{\mathcal{V}}} (u_j \otimes u_j) \|_F = | \lambda_j |$, hence (38) is actually a chain of equalities. Specifically,

$$
\| P_{\hat{\mathcal{V}}} (u_j \otimes u_j) \|_F \| X \|_F = | \langle P_{\hat{\mathcal{V}}} (u_j \otimes u_j), X \rangle | ,
$$

which implies $X = c P_{\hat{\mathcal{V}}} (u_j \otimes u_j)$ for some scalar $c$. Since $\| X \|_F = 1$, $c = \lambda_j^{-1}$ follows from

$$
1 = \langle c P_{\hat{\mathcal{V}}} (u_j \otimes u_j), X \rangle = c \langle u_j \otimes u_j, X \rangle = c \lambda_j .
$$

Lemmas 18 and 19 show that the stationary point condition (32) for $M$ with $\| M \| = | \lambda_{i^*} (M) |$ and isolated $\lambda_{i^*}$ is equivalent to both

$$
M = \lambda_{i^*}^{-1} P_{\hat{\mathcal{V}}} (u_{i^*} (M) \otimes u_{i^*} (M)) , \quad \text{and} \quad \| P_{\hat{\mathcal{V}}} (u_{i^*} (M) \otimes u_{i^*} (M)) \|_F = | \lambda_{i^*} (X) | .
$$

A similar condition appears naturally if we characterize the fixed points of $F_{\gamma}$. 

Lemma 20 Let $\gamma > 0$ and $X \in \hat{\mathcal{V}} \cap \mathbb{S}$ with $\lambda_1 (X) > 0$. Then we have

$$
0 < \lambda_1 (X) < \| P_{\hat{\mathcal{V}}} (u_1 (X) \otimes u_1 (X)) \|_F \quad \text{if and only if} \quad \lambda_1 (F (X)) > \lambda_1 (X) , \quad (39)
$$

$$
\lambda_1 (X) = \| P_{\hat{\mathcal{V}}} (u_1 (X) \otimes u_1 (X)) \|_F \quad \text{if and only if} \quad F_{\gamma} (X) = X . \quad (40)
$$

**Proof** For simplicity, we denote $u := u_1 (X)$ and $\lambda = \lambda_1 (X)$ in this proof. We first prove that $0 < \lambda < \| P_{\hat{\mathcal{V}}} (u \otimes u) \|_F$ implies $\lambda_1 (F (X)) > \lambda$. It suffices to show that there exists any unit vector $v$ such that $v^T F_{\gamma} (X) v > \lambda$. In particular, we can test
$F_{\gamma}(X)$ with $v = u$, which yields the identity

$$u^T F_{\gamma}(X)u - \lambda = \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F^{-1}\{P_{\tilde{W}}(X + \gamma u \otimes u), u \otimes u\} - \lambda$$

$$= \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F^{-1}\{(X, u \otimes u) + \gamma\{P_{\tilde{W}}(u \otimes u), u \otimes u\}\} - \lambda$$

$$= \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F^{-1}\{\lambda + \gamma\|P_{\tilde{W}}(u \otimes u)\|_F^2\} - \lambda$$

$$= \frac{\lambda \left(1 - \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F\right) + \gamma\|P_{\tilde{W}}(u \otimes u)\|_F^2}{\|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F}.$$

By using now $\lambda < \|P_{\tilde{W}}(u \otimes u)\|_F$, we can bound

$$1 - \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F = 1 - \sqrt{\|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F^2}$$

$$= 1 - \sqrt{1 + \gamma^2\|P_{\tilde{W}}(u \otimes u)\|_F^2 + 2\gamma\langle X, P_{\tilde{W}}(u \otimes u)\rangle}$$

$$= 1 - \sqrt{1 + \gamma^2\|P_{\tilde{W}}(u \otimes u)\|_F^2 + 2\gamma\lambda}$$

$$> 1 - \sqrt{\left(1 + \gamma\|P_{\tilde{W}}(u \otimes u)\|_F\right)^2}$$

$$= -\gamma\|P_{\tilde{W}}(u \otimes u)\|_F.$$

Inserting this inequality in the previous identity, we obtain the wished result by

$$uF_{\gamma}(X)u - \lambda = \frac{\lambda \left(1 - \|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F\right) + \gamma\|P_{\tilde{W}}(u \otimes u)\|_F^2}{\|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F}$$

$$> \frac{-\lambda\gamma\|P_{\tilde{W}}(u \otimes u)\|_F + \gamma\|P_{\tilde{W}}(u \otimes u)\|_F^2}{\|P_{\tilde{W}}(X + \gamma u \otimes u)\|_F} > 0. \quad \text{(41)}$$

We show now that $F_{\gamma}(X) = X$ implies $\lambda = \|P_{\tilde{W}}(u_1(X) \otimes u_1(X))\|_F$. We notice that $F_{\gamma}(X) = X$ implies $\lambda(F_{\gamma}(X)) = \lambda$, and thus $\lambda \geq \|P_{\tilde{W}}(u \otimes u)\|_F$ according to (39). Since generally $\lambda \leq \|P_{\tilde{W}}(u \otimes u)\|_F$ by Lemma 19, equality follows.

We address now the converse, i.e., $\lambda = \|P_{\tilde{W}}(u \otimes u)\|_F$ implies $F_{\gamma}(X) = X$, and we note that $\lambda = \|P_{\tilde{W}}(u \otimes u)\|_F$ implies $X = \lambda^{-1}P_{\tilde{W}}(u \otimes u)$ by Lemma 19. Using
this and the definition of $F_\gamma(X)$, we get

$$
F_\gamma(X) = \frac{P_{\hat{W}}(X + \gamma u \otimes u)}{\|P_{\hat{W}}(X + \gamma u \otimes u)\|_F} = \frac{(\lambda^{-1} + \gamma) P_{\hat{W}}(u \otimes u)}{(\lambda^{-1} + \gamma) \|P_{\hat{W}}(u \otimes u)\|_F} = \frac{P_{\hat{W}}(u \otimes u)}{\|P_{\hat{W}}(u \otimes u)\|_F} = X
$$

To conclude the proof it remains to show $\lambda_1(F(X)) > \lambda$ implies $0 < \lambda < \|P_{\hat{W}}(u \otimes u)\|_F$. As $\lambda \leq \|P_{\hat{W}}(u \otimes u)\|_F$ and $\lambda_1(F(X)) > \lambda$ implies $F_\gamma(X) \neq X$ and therefore $\lambda \neq \|P_{\hat{W}}(u \otimes u)\|_F$, then necessarily $\lambda < \|P_{\hat{W}}(u \otimes u)\|_F$.

The preceding Lemma implies the convergence of $(\lambda_1(M_j))_{j \in \mathbb{N}}$ by monotonicity. Moreover, we can also use such convergence to establish $\|M_{j+1} - M_j\|_F \to 0$.

**Lemma 21** Let $\gamma > 0$, $M_0 \in \hat{W} \cap \mathbb{S}$ with $\lambda_1(M_0) > 0$, and let $M_j := F_\gamma(M_{j-1})$. The sequence $(\lambda_1(M_j))_{j \in \mathbb{N}}$ converges to a well-defined limit $\lambda_\infty$, and $\lim_{j \to \infty} \|M_{j+1} - M_j\|_F = 0$.

**Proof** Denote $U_j := P_{\hat{W}}(u(M_j) \otimes u(M_j))$, $\lambda_j = \lambda(M_j)$ for simplicity. The sequence $(\lambda_j)_{j \in \mathbb{N}}$ is monotone in the bounded domain $[0, 1]$ by Lemma 20 and therefore converges to a limit $\lambda_\infty$. To prove $\|M_{j+1} - M_j\|_F \to 0$, we will exploit $(\lambda_{j+1} - \lambda_j) \to 0$. We first have ($\|U_j\|_F - \lambda_j \to 0$ since (41) yields

$$
\lambda_{j+1} - \lambda_j \geq \frac{\gamma}{\|M_j + \gamma U_j\|_F} (\|U_j\|_F - \lambda_j) \geq \frac{\gamma}{1 + \gamma} \|U_j\|_F (\|U_j\|_F - \lambda_j),
$$

and $\|U_j\|_F \geq \lambda_j \geq \lambda_0$ for all $j$. Define the shorthand $\Delta_j := \|U_j\|_F - \lambda_j$. We will now show that $\|M_{j+1} - M_j\|_F \leq C \Delta_j$ for some constant $C$. First notice that

$$
\|M_j - \lambda_j^{-1} U_j\|_F = \sqrt{1 + \frac{\|U_j\|_F^2}{(\lambda_j)^2} - 2\lambda_j^{-1} \langle M_j, U_j \rangle} = \sqrt{\frac{\|U_j\|_F^2}{(\lambda_j)^2} - 1}
$$

$$
= \frac{\|U_j\|_F^2 - (\lambda_j)^2}{(\lambda_j)^2} \leq \lambda_0^{-1} \sqrt{2\Delta_j}.
$$

Therefore there exists a matrix $E_j$ with $M_j = \lambda_j^{-1} U_j + E_j$ and $\|E_j\| \leq \lambda_0^{-1} \sqrt{2\Delta_j}$. Furthermore, by the triangle inequality we have

$$
\|M_{j+1} - M_j\|_F \leq \|M_{j+1} - \lambda_j^{-1} U_j\|_F + \lambda_0^{-1} \sqrt{2\Delta_j},
$$

hence it remains to bound the first term. Using $M_j = \lambda_j^{-1} U_j + E_j$ and $M_{j+1} = M_j + \gamma U_j$ $\|M_j + \gamma U_j\|_F^2 (M_j + \gamma U_j)$, we have $\|M_j + \gamma U_j\|_F M_{j+1} = (\lambda_j^{-1} + \gamma) U_j + E_j$.
and thus
\[ \left\| M_j + \gamma U_j \right\|_F (M_{j+1} - \lambda_j^{-1} U_j) \right\|_F \\
= \left\| (\lambda_j^{-1} + \gamma) U_j + E_j - (\lambda_j^{-1} + \gamma) U_j + E_j \right\|_F \lambda_j^{-1} U_j \right\|_F \\
\leq \left\| \lambda_j^{-1} + \gamma - (\lambda_j^{-1} + \gamma) U_j + E_j \right\|_F \left\| U_j \right\|_F + \left\| E_j \right\|_F \\
\leq \left( (\lambda_j^{-1} + \gamma) \left\| U_j \right\|_F \lambda_j^{-1} - (\lambda_j^{-1} + \gamma) \right) \left\| U_j \right\|_F + (1 + 2\lambda_0^{-1}) \left\| E_j \right\|_F \\
\leq (\lambda_0^{-1} + \gamma)(\lambda_0^{-1} + 1) \left\| U_j \right\|_F + (1 + 2\lambda_0^{-1}) \left\| E_j \right\|_F \leq (\lambda_0^{-1} + \gamma)\lambda_0^{-1} \Delta_j + (1 + 2\lambda_0^{-1}) \sqrt{\Delta_j}.
\]

Since \( \left\| M_j + \gamma U_j \right\|_F \geq 1 \) according to Lemma 17, \( \left\| M_{j+1} - M_j \right\| \to 0 \) follows.

It remains to show that convergent subsequences of \( (M_j)_{j \in \mathbb{N}} \) converge to fixed points of \( F_\gamma \). Then by (40), Lemmas 18 and 19, fixed points satisfy the first-order optimality condition (32) and are stationary points of (31). To prove convergence of subsequences to fixed points, we require continuity of \( F_\gamma \). The following Lemma shows that \( F_\gamma \) is continuous for matrices \( X \) satisfying \( \lambda_1(X) > \sqrt{1/2}, \) i.e., if the largest eigenvector is isolated and \( u_1(X) \) is a continuous function of \( X \).

**Lemma 22** Let \( \gamma > 0, \epsilon > 0 \) arbitrary, and define \( \mathcal{M}_\epsilon := \{ M \in \hat{W} \cap S : \lambda(M) \geq (1/2 + \epsilon)^{-1/2} \} \). Then \( F_\gamma(X) \in \mathcal{M}_\epsilon \) for all \( X \in \mathcal{M}_\epsilon \), and \( F_\gamma \) is \( \| \cdot \|_F \)-Lipschitz continuous, with Lipschitz constant \( (1 + \gamma/\epsilon) \).

**Proof** \( F_\gamma(X) \in \mathcal{M}_\epsilon \) follows directly from Lemma 20, i.e., from the fact that the largest eigenvalue is only increased by applying \( F_\gamma \). For the continuity, consider \( X, Y, Y \in \mathcal{M}_\epsilon \). We first note that by using [6, Theorem 7.3.1] and \( \lambda_i(Y) \leq \sqrt{1/2 - \epsilon} \) for \( i = 2, \ldots, m_0 \) we get
\[
\left\| X + \gamma P_{\hat{W}}(u_1(X) \otimes u_1(X)) - Y - \gamma P_{\hat{W}}(u_1(Y) \otimes u_1(Y)) \right\|_F \\
\leq \left\| X - Y \right\|_F + \gamma \left\| u_1(X) \otimes u_1(X) - u_1(Y) \otimes u_1(Y) \right\|_F \\
\leq \left\| X - Y \right\|_F + \gamma \frac{\left\| X - Y \right\|_F}{\sqrt{1/2 - \epsilon} \geq 1} \leq \left( 1 + \frac{\gamma}{\epsilon} \right) \left\| X - Y \right\|_F.
\]

Furthermore, we have \( \left\| X + \gamma P_{\hat{W}}(u_1(X) \otimes u_1(X)) \right\|_F^2 \geq 1 \) according to Lemma 17, and therefore, \( P_{\hat{W}} \) acts on \( X + \gamma P_{\hat{W}}(u_1(X) \otimes u_1(X)) \) and \( Y + \gamma P_{\hat{W}}(u_1(Y) \otimes u_1(Y)) \) as a projection onto the convex set \( \{ X : \left\| X \right\|_F \leq 1 \} \). Therefore it acts as a contraction and the result follows from
\[
\left\| F_\gamma(X) - F_\gamma(Y) \right\|_F \leq \left\| X + \gamma P_{\hat{W}}(u_1(X) \otimes u_1(X)) - Y - \gamma P_{\hat{W}}(u_1(Y) \otimes u_1(Y)) \right\|_F.
\]

The convergence to fixed points of any subsequence of \( (M_j)_{j \in \mathbb{N}} \) now follows as a corollary of Lemma 34 in the Appendix.
Theorem 23 Let $\epsilon > 0$, $\gamma > 0$, $M_0 \in \hat{\mathcal{W}} \cap \mathbb{S}$ with $\lambda(M_0) \geq 1/\sqrt{2} + \epsilon$ and let $M_{j+1} := F_{\gamma}(M_j)$ as generated by Algorithm 3. Then $(M_{j+1})_{j \in \mathbb{N}}$ has a convergent subsequence, and any such subsequence converges to a fixed point of $F_{\gamma}$, respectively, a stationary point of (31).

Proof By Lemma 22, the operator $F_{\gamma}$ is continuous on $\mathcal{M}_\epsilon := \{M \in \hat{\mathcal{W}} \cap \mathbb{S} : \lambda_1(M) \geq (1/2 + \epsilon)^{-1/2}\}$ for any $\epsilon > 0$. Moreover, by Lemma 20 we have $(M_{j+1})_{j \in \mathbb{N}} \subset \mathcal{M}_\epsilon$, and by Lemma 21 we have $\|M_{j+1} - M_j\|_{\mathcal{F}} \to 0$. Therefore we can apply Lemma 34 to see that any convergent subsequence converges to a fixed point of $F_{\gamma}$. Moreover, since $(M_{j+1})_{j \in \mathbb{N}}$ is bounded, there exists at least one convergent subsequence by Bolzano–Weierstrass. Finally, any fixed point $M$ of $F_{\gamma}$ can be written as $M = \lambda_1(M) P_{\mathcal{W}}(u_1(M) \otimes u_1(M))$ by Lemma 19 and Lemma 20. Since $\lambda_1(M) > 1/\sqrt{2}$, it is an isolated eigenvalue satisfying $\lambda_1(M) = \|M\|$, and thus, $M$ satisfies the first order optimality condition (32) of (31) by Theorem 13.

Remark 24 The analysis of the convergence of Algorithm 3 we provide above does not use the structure of the space $\mathcal{W}$ and focuses exclusively on the behavior of the first eigenvalue $\lambda_1$. As a consequence, it does guarantee that its iterations have monotonically increasing spectral norm and that they generically converges to stationary points of (31). However, it does not ensure convergence to nonspurious, minimal rank local minimizers of (31). In the numerical experiments of Sect. 4, where $\{w_j : j \in [m]\}$ are sampled randomly from certain distributions, an overwhelming majority of sequences $(M_j)_{j \in \mathbb{N}}$ converges to a near rank-1 matrix with an eigenvalue close to one, whose corresponding eigenvector approximates a network profile with good accuracy. To explain this success, we would need a finer and quantitative analysis of the increase of the spectral norm during the iterations, for instance by quantifying the gap

$$\left[ \Theta \left\| P_{\hat{\mathcal{W}}}(u_1(X) \otimes u_1(X)) \right\|_{\mathcal{F}} - \lambda_1(X) \right] \geq 0,$$

by means of a suitable constant $0 < \Theta < 1$. As clarified in the proof of Lemma 19, the smaller the constant $\Theta > 0$ is, the larger is the increase of the spectral norm $\|M_{j+1}\| > \|M_j\|$ between iterations of Algorithm 3. The following result is an attempt to gain a quantitative estimate for $\Theta$ by injecting more information about the structure of the space $\mathcal{W}$.

In order to simplify the analysis, let us assume $\delta = 0$ or $\hat{\mathcal{W}} = \mathcal{W}$.

Proposition 25 Assume that $\{W_\ell := w_\ell \otimes w_\ell : \ell \in [m]\}$ forms a frame for $\mathcal{W}$, i.e., there exist constants $c_{\mathcal{W}}, C_{\mathcal{W}} > 0$ such that for all $X \in \mathcal{W}$

$$c_{\mathcal{W}}\|X\|_{\mathcal{F}}^2 \leq \sum_{\ell=1}^{m} \langle X, w_\ell \otimes w_\ell \rangle_{\mathcal{F}}^2 \leq C_{\mathcal{W}}\|X\|_{\mathcal{F}}^2.$$

Denote $\{\tilde{W}_\ell : \ell \in [m]\}$ the canonical dual frame so that

$$P_{\hat{\mathcal{W}}}(X) = \sum_{\ell=1}^{m} \langle X, \tilde{W}_\ell \rangle_{\mathcal{F}} W_\ell.$$  

\(\odot\) Springer
for any symmetric matrix $X$. Then, for $X \in \mathcal{W}$ and the notation $\lambda_j := \lambda_j(X)$, $\lambda_1 = \|X\|$ and $u_j := u_j(X)$, we have

$$
\lambda_1 = \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \sum_{j=1}^{m_0} \sum_{\ell=1}^{m} \frac{\lambda_j \langle u_j \otimes u_j, \tilde{W}_\ell \rangle_{F}}{\|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F} \right) \\
\leq \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \sum_{\ell=1}^{m} \left( \sum_{j=1}^{m_0} \lambda_j \|P_{\mathcal{W}}(u_j \otimes u_j)\|_F \right)^{1/2} \right) \\
\leq \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \sum_{\ell=1}^{m} \lambda_j \|P_{\mathcal{W}}(u_j \otimes u_j)\|_F \right)^{1/2} \\
=: \Theta.
$$

**Proof** Let us fix $X \in \mathcal{W}$. Then we have two ways of representing $X$, its frame decomposition and its spectral decomposition:

$$
X = \sum_{\ell=1}^{m} (X, \tilde{W}_\ell)_{F} W_\ell = \sum_{j=1}^{m_0} \lambda_j u_j \otimes u_j.
$$

By using both the decompositions and again the notation $W_\ell = w_\ell \otimes w_\ell$, we obtain

$$
\lambda_1 = u_1^T X u_1 = \sum_{\ell=1}^{m} \left( \sum_{j=1}^{m_0} \lambda_j \langle u_j \otimes u_j, \tilde{W}_\ell \rangle_{F} \right) u_1^T W_\ell u_1 \\
= \sum_{\ell=1}^{m} \left( \sum_{j=1}^{m_0} \lambda_j \langle u_j \otimes u_j, \tilde{W}_\ell \rangle \right) \langle w_\ell, u_1 \rangle^2 \\
= \sum_{\ell=1}^{m} \sum_{j=1}^{m_0} \lambda_j \langle u_j \otimes u_j, \tilde{W}_\ell \rangle \langle w_\ell, u_1 \rangle^2 \\
= \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \sum_{j=1}^{m_0} \sum_{\ell=1}^{m} \lambda_j \langle u_j \otimes u_j, \tilde{W}_\ell \rangle_{F} \langle w_\ell, u_1 \rangle^2 \right) \\
= \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \sum_{j=1}^{m_0} \sum_{\ell=1}^{m} \lambda_j \frac{\langle u_j \otimes u_j, \tilde{W}_\ell \rangle_{F}}{\|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F} \langle w_\ell, u_1 \rangle^2 \right)^{1/2}.
$$

By observing that $\sum_{\ell=1}^{m} \langle u_j \otimes u_j, \tilde{W}_\ell \rangle_{F} \leq A^{-1} \|P_{\mathcal{W}}(u_j \otimes u_j)\|^2_{F}$ (canonical dual frame upper bound), and using Cauchy–Schwarz inequality, we can further estimate

$$
\lambda_1 \leq \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \frac{c_{\mathcal{W}}}{c_{\mathcal{W}}} \sum_{j=1}^{m_0} \lambda_j \|P_{\mathcal{W}}(u_j \otimes u_j)\|_F \right)^{1/2} \left( \sum_{\ell=1}^{m} \langle w_\ell, u_1 \rangle^4 \right)^{1/2} \\
\leq \|P_{\mathcal{W}}(u_1 \otimes u_1)\|_F \left( \frac{c_{\mathcal{W}}}{c_{\mathcal{W}}} \right)^{1/2} \left( \sum_{j=1}^{m_0} \lambda_j \|P_{\mathcal{W}}(u_j \otimes u_j)\|_F \right)^{1/2}.
$$
where in the last inequality we applied the estimates
\[ \sum_{\ell=1}^{m} |\langle w_{\ell}, u_1 \rangle|^4 = \sum_{\ell=1}^{m} |\langle w_{\ell} \otimes w_{\ell}, u_1 \otimes u_1 \rangle|^2_F \leq C_{\mathcal{W}} \| P_{\mathcal{W}}(u_1 \otimes u_1) \|_F^2. \]

The meaning of estimate (42) is explained by the following mechanism: Whenever the deviation of an iteration \( M_j \) of Algorithm 3 from being a rank-1 matrix in \( \mathcal{W} \) is large, in the sense that \( \| P_{\mathcal{W}}(u_1 \otimes u_1) \|_F \) is small, the constant \( \Theta = \left( \frac{C_{\mathcal{W}}}{c_{\mathcal{W}}} \right)^{1/2} \left( \sum_{\lambda_j > 0} \lambda_j \| P_{\mathcal{W}}(u_j \otimes u_j) \|_F \right) \) is also small and the iteration \( M_{j+1} = F_{\gamma}(M_j) \) will efficiently increase the spectral norm. The gain will reduce as soon as the iteration \( M_j \) gets closer and closer to a rank-1 matrix. It would be perhaps possible to get an even more precise analysis of the behavior of Algorithm 3, by considering simultaneously the dynamics of (the gaps between) different eigenvalues (not only focusing on \( \lambda_1 \)). Unfortunately, we could not find yet a proper and conclusive argument.

4 Numerical Experiments About the Recovery of Network Profiles

In this section, we present numerical experiments about the recovery of network weights \( \{a_i : i \in [m_0]\} \) and \( \{v_\ell : \ell \in [m_1]\} \) from few point queries of the network. The recovery procedure leverages the theoretical insights that have been provided in previous sections. Without much loss of generality, we neglect the active subspace reduction argument.

We perform experiments for different scenarios, where either the activation function or the construction of the network weights varies. Guided by our theoretical results, we pay particular attention to how the network architecture, e.g., \( m_0 \) and \( m_1 \), influences the simulation results. The entire procedure is rather flexible and can be adjusted in different ways, e.g., changing the distribution \( \mu_X \). To provide a fair account of the success, we fix hyperparameters of the approach throughout all experiments. Test scenarios, hyperparameters, and error measures are reported below in more detail. Afterwards, we present and discuss the results.

Scenarios and construction of the networks

The network is constructed by choosing activation functions and network weights \( \{a_i : i \in [m_0]\}, \{b_\ell : \ell \in [m_1]\} \), for which \( v_\ell \) is then defined via \( v_\ell = \frac{AG_{ab_\ell}}{\|AG_{ab_\ell}\|_2} \), see Definition 3. To construct activation functions, we set \( g_\ell(t) = \phi(t + \theta_i) \) for \( i \in [m_0] \), and \( h_\ell(t) = \phi(t + \tau_\ell) \) for \( \ell \in [m_1] \). We choose
either \( \phi(t) = \tanh(t) \) or \( \phi(t) = \frac{1}{1 + e^{-t}} - \frac{1}{2} \) (shifted sigmoid function), and sample offsets (called also biases) \( \theta_i, \tau_{\ell} \) independently at random from \( \mathcal{N}(0, 0.01) \).

As made clear by our theory, see Theorem 16, a sufficient condition for successful recovery of the entangled weights is \( \nu = C_F - 1 \) to be small, where \( C_F \) is the upper frame constant of the entangled weights as in Definition 3. In the following numerical experiments, we wish to verify how crucial is this requirement. Thus, we test two different scenarios for the weights. The first scenario, which is designed to best fulfill the sufficient condition \( \nu \approx 0 \), models both \( \{a_i : i \in [m_0]\} \) and \( \{b_{\ell} : \ell \in [m_1]\} \) as perturbed orthogonal systems. For their construction, we first sample orthogonal bases uniformly at random and then apply a random perturbation. The perturbation is such that \( (\sum_{i=1}^{m_0} (\sigma_i(A) - 1)^2)^{-1/2} \approx (\sum_{i=1}^{m_1} (\sigma_i(B) - 1)^2)^{-1/2} \approx 0.3 \), where \( \sigma_i(A) \) and \( \sigma_i(B) \) denote singular values of \( A \) and \( B \). In the second case, we sample the (entangled) weights independently from \( \mathcal{U}(\mathbb{S}^{m_0-1}) \). In this situation, as the dimensionality \( d = m_0 \) is relatively small, the system will likely not fulfill well the condition \( \nu \approx 0 \); however, as the dimension \( d = m_0 \) is chosen larger, the weights tend to be more incoherent and gradually approaching the previous scenario.

Hyperparameters. Unless stated differently, we sample \( m_X = 1000 \) Hessian locations from \( \mu_X = \sqrt{m_0} \mathcal{U}(\mathbb{S}^{m_0-1}) \), and use \( \epsilon = 10^{-5} \) in the finite difference approximation (17). We generate 1000 random matrices \( \{M_k^j : k \in [1000]\} \) by sampling \( m^k \sim \mathcal{N}(0, \text{Id}_{m_0+m_1}) \), and by defining \( M_0^j := \mathbb{P}_S(\sum_{i=1}^{m_0+m_1} m_i^k u_i) \), where the \( u_i \)'s are as in Algorithm 2. The constant \( \gamma = 2 \) is used in the definition of \( F_\gamma \), and the iteration is stopped if \( \lambda_1(M_{j+1}^k) - \lambda_1(M_j^k) < 10^{-5} \), or after 200 iterations. kMeans++ is run with default settings using sklearn. All reported results are averages over 30 repetitions.

Error measures. Three error measures are reported:

- the normalized projection error \( \frac{\|\hat{P}_{\mathcal{M}} - P_{\mathcal{M}}\|_F^2}{m_0+m_1} \),
- a false positive rate \( \text{FP}(T) = \frac{\#\{i : E(u_i) > T\}}{m_0+m_1} \), where \( T > 0 \) is a threshold, and \( E(u) \) is defined by,
  \[
  E(u) := \min_{w \in \{\pm a_i, \pm v_{\ell} : i \in [m_0], \ell \in [m_1]\}} \|u - w\|_2^2,
  \]

- recovery rate \( R_\alpha(T) = \frac{\#\{i : E(a_i) < T\}}{m_0} \), and \( R_v(T) = \frac{\#\{\ell : E(v_{\ell}) < T\}}{m_1} \), where
  \[
  E(u) := \min_{w \in \{\pm \hat{\alpha}_j, \pm \hat{v}_j : j \in [m_0+m_1]\}} \|u - w\|_2^2.
  \]

The results of the study are presented in Figs. 2 and 3 and show that our procedure typically recovers many of the network weights, while suffering only few false positives. Considering for example a sigmoidal network, we have almost perfect recovery of the weights in both layers at a threshold of \( T = 0.05 \) for any network architecture, see Fig. 3a, c. For a tanh-network, the performance is slightly worse, but we still...
Fig. 2  Error in approximating $W$ for perturbed orthogonal weights and different activation functions

Fig. 3  False positive and recovery rates for perturbed orthogonal weights and for different activation functions

recover most weights in the second layer, and a large portion in the first layer at a reasonable threshold, see Fig. 3b, d.

Inspecting the plots more closely, we can notice some shared trends and differences between sigmoid and tanh networks. In both cases, the performance improves when increasing the input dimensionality or, equivalently, the number of neurons in the first layer, even though the number of weights that need to be recovered increases
accordingly. This is particularly the case for tanh-networks as visualized in Fig. 3b, d and is most likely caused by reduced correlation of the weights in higher dimensions. As previously mentioned, the correlation is encoded within the constant \( v = C_F - 1 \) used in the analysis in Sect. 3.

For fixed \( m_0 \) on the other hand, different activation functions react differently to changes of \( m_1 \). For \( m_1 \) larger, considering a sigmoid network, the projection error increases, and the recovery of weights in the second layer worsens as shown in Fig. 2a, c. This is expected by Theorem 5. Inspecting the results for tanh networks, the projection error actually decreases when increasing \( m_1 \), see Fig. 2b, and the recovery performance gets better. Figure 3d shows that especially weights in the first layer are more easily recovered if \( m_1 \) is large, such that the case \( m_0 = 45, m_1 = 23 \) allows for perfect recovery at a threshold \( T = 0.05 \). This behavior cannot be fully explained by our general theory, e.g., Theorem 5.

**Results for random weights from the unit sphere** When sampling the weights independently from the unit sphere, the recovery problem seems more challenging for moderate dimension \( d = m_0 \) and for both activation functions. This confirms the expectation that the smallness of \( v = C_F - 1 \) is somehow crucial. Figure 4c, d suggest that especially recovering the weights of the second layer is more difficult than in the perturbed orthogonal case. Still, we achieve good performance in many cases. For sigmoid networks, Fig. 4c shows that we always recover most weights in the first layer, and a large portion of weights in the second layer if \( m_1/m_0 \) is small. Moreover, keeping \( m_1/m_0 \) constant while increasing \( m_0 \) improves the performance significantly, as we expect from an improved constant \( v = C_F - 1 \). Figure 4a, c show almost perfect recovery for \( m_0 = 45, m_1 = 5 \), while suffering only few false positives.

For tanh-networks, Fig. 4d shows that increasing \( m_0 \) benefits recovery of weights in both layers, while increasing \( m_1 \) benefits recovery of first layer weights and harms recovery of second layer weights. We still achieve small false positive rates in Fig. 4b, and good recovery for \( m_0 = 45 \), and the trend continues when further increasing \( m_0 \).

Finally, a notable difference between the perturbed orthogonal case and the unit-sphere case is the behavior of the projection error \( \| P_{\hat{Y}_J} - P_Y \|_F/(m_0 + m_1) \) for networks with sigmoid activation function. Comparing Figs. 2a and 5a, the dependency of the projection error on \( m_1 \) is stronger when sampling independently from the unit-sphere. This is explained by Theorem 5 since \( \| B \|_2^2 \) is independent of \( m_1 \) in the perturbed orthogonal case and grows like \( O(\sqrt{m_1}) \) when sampling from the unit-sphere.

**5 Open Problems**

With the previous theoretical results of Sect. 3 and the numerical experiments of Sect. 4, we show how to reliably recover the entangled weights \( \{ \hat{w}_j : j \in [m_0 + m_1] \} \approx \{ w_j : j \in [m_0 + m_1] \} = \{ a_i : i \in [m_0] \} \cup \{ v_\ell : \ell \in [m_1] \} \). However, some issues remain open.
(i) In Theorem 5, the dependency of $\alpha > 0$ on the network architecture and on the input distribution $\mu_X$ is left implicit. However, it plays a crucial role for fully estimating the overall sample complexity.

(ii) While we could prove that Algorithm 3 is increasing the spectral norm of its iterates in $\mathcal{V} \cap \mathbb{S}$, we could not show yet that it converges always to nearly rank-1 matrices in $\mathcal{V}$, despite it is so numerically observed, see also Remark 24. We also could not exclude the existence of spurious local minimizers of the nonlinear program (31), as stated in Theorem 16. However, we conjecture that there are none or that they are somehow hard to observe numerically.

(iii) Obtaining the approximating vectors $\{\hat{w}_j : j \in [m_0 + m_1]\} \approx \{w_j : j \in [m_0 + m_1]\} = \{a_i : i \in [m_0]\} \cup \{v_\ell : \ell \in [m_1]\}$ does not suffice to reconstruct the entire network. In fact, it is impossible a priori to know whether $\hat{w}_j$ approximates one $a_i$ or some other $v_\ell$, up to sign and permutations, and the attribution to the corresponding layer needs to be derived from querying the network.

(iv) Once we obtained, up to sign and permutations, $\{\hat{a}_i : i \in [m_0]\} \approx \{a_i : i \in [m_0]\}$ and $\{\hat{v}_\ell : \ell \in [m_1]\} \approx \{v_\ell : \ell \in [m_1]\}$ from properly grouping $\{\hat{w}_j : j \in [m_0 + m_1]\}$, it would remain to approximate/identify the activations functions $g_i$ and $h_\ell$. In the case where $g_i(\cdot) = \phi(\cdot - \theta_i)$ and $h_\ell(\cdot) = \phi(\cdot - \tau_\ell)$, this would simply mean to be able to identify the shifts $\theta_i, i \in [m_0]$, and $\tau_\ell, \ell \in [m_1]$. Such identification is also crucial for computing the matrix $G_0 = \text{diag}(g_1'(0), \ldots, g_{m_0}'(0))$ which allows the disentanglement of the weights $b_\ell$ from the weights $A$ and $v_\ell = AG_0 b_\ell / \|AG_0 b_\ell\|_2$. At this point, the network is fully reconstructed.

(v) The generalization of our approach to networks with more than two hidden layers is clearly the next relevant issue to be considered as a natural development of this work.

While problems (i) and (ii) seem to be difficult to solve by the methods we used in this paper, we think that problems (iii) and (iv) are solvable both theoretically and numerically with just a bit more effort. For a self-contained conclusion of this paper, in the following sections we sketch some possible approaches to these issues, as a glimpse towards future developments, which will be more exhaustively included in [21]. The generalization of our approach to networks with more than two hidden layers as mentioned in (v) is surprisingly simpler than one may expect, and it is in the course of finalization [21]. For a network $f(x) := f(x; W_1, \ldots, W_L) = 1^T g_L(W_L^T g_{L-1}(W_{L-1}^T \ldots (g_1(W_1^T x)) \ldots), with $L > 2$, again by second order differentiation is possible to obtain an approximation space

\[ \hat{\mathcal{V}} \approx \text{span}\{w_{1,i} \otimes w_{1,j}, (W_2 G_1 w_{1,j}) \otimes (W_2 G_1 w_{1,j}), \ldots, (W_L G_L \ldots W_2 G_1 w_{1,j}) \otimes (W_L G_L \ldots W_2 G_1 w_{1,j})\}, \]

of the matrix space spanned by the tensors of entangled weights, where $G_i$ are suitable diagonal matrices depending on the activation functions. The tensors $(W_k G_k \ldots W_2 G_1 w_{1,j}) \otimes (W_k^T G_k \ldots W_2^T G_1 w_{1,j})$ can be again identified by a minimal rank principle. The disentanglement goes again by a layer by layer procedure as in this paper, see also [5].
6 Reconstruction of the Entire Network

In this section, we address problems (iii) and (iv) as described in Sect. 5. Our final goal is of course to construct a two-layer network $\hat{f}$ with number of nodes equaling $m_0$ and $m_1$ such that $\hat{f} \approx f$. Additionally we also study whether the individual building
blocks (e.g., matrices $\hat{A}$, $\hat{B}$, and biases in both layers) of $\hat{f}$ match their corresponding counterparts of $f$.

To construct $\hat{f}$, we first discuss how recovered entangled weights $\{\hat{w}_i : i \in [m_0 + m_1]\}$ (see Sect. 4) can be assigned to either the first, or the second layer, depending on whether $\hat{w}_j$ approximates one of the $a_i$'s, or one of the $v_\ell$'s. Afterwards we discuss a modified gradient descent approach that optimizes the deparametrized network (its entangled weights are known at this point!) over the remaining, unknown parameters of the network function, e.g., biases $\theta_i$ and $\tau_\ell$.

6.1 Distinguishing First and Second Layer Weights

Attributing approximate entangled weights to first or second layer is generally a challenging task. In fact, even the true weights $\{a_i : i \in [m_0]\}, \{v_\ell : \ell \in [m_1]\}$ cannot be assigned to the correct layer based exclusively on their entries when no additional a priori information (e.g., some distributional assumptions) is available. Therefore, assigning $\hat{w}_j, j \in [m_0 + m_1]$ to the correct layer requires using again the network $f$ itself and thus to query additional information.

The strategy we sketch here is designed for sigmoidal activation functions and networks with (perturbed) orthogonal weights in each layer. Sigmoidal functions are monotonic, have bell-shaped first derivative, and are bounded by two horizontal asymptotes as the input tends to $\pm\infty$. If activation functions $\{g_i : i \in [m_0]\}$ and $\{h_\ell : \ell \in [m_1]\}$ are translated sigmoidal, their properties imply

$$\|\nabla f(tw)\|_2 = \left( \sum_{i=1}^{m_0} g_i'(ta_i^T w)^2 \left( \sum_{\ell=1}^{m_1} h_\ell'(b_\ell^T g(tA^T w))b_\ell \right)^2 \right)^{1/2} \to 0, \quad \text{as } t \to \infty,$$

whenever any direction $w$ has nonzero correlation $a_i^T w \neq 0$ with each first layer neuron in $\{a_i : i \in [m_0]\}$.

Assume now that $\{a_i : i \in [m_0]\}$ is a perturbed orthonormal system, and that second layer weights $\{b_\ell : \ell \in [m_1]\}$ are generic and dense (nonsparse). Recalling the definition $v_\ell = AG_0b_\ell/\|AG_0b_\ell\|$, the vector $v_\ell$ has, in this case, generally nonzero angle with each vector in $\{a_i : i \in [m_0]\}$, while $a_i^T a_j \approx 0$ for any $i \neq j$. Utilizing this with observation (43), it follows that $\|\nabla f(ta_i)\|$ is expected to tend to 0 much slower than $\|\nabla f(tv_\ell)\|$ as $t \to \infty$. In fact, if $\{a_i : i \in [m_0]\}$ was an exactly orthonormal system, $\|\nabla f(ta_i)\|$ eventually would equal a positive constant when $t \to \infty$. We illustrate in Fig. 6 the different behavior of the trajectories $t \to \|\nabla f(tw)\|_2$ for $w \in \{\hat{w}_j \approx a_i \text{ for some } i\}$ and for $w \in \{\hat{w}_j \approx v_\ell \text{ for some } \ell\}$.

Practically, for $T \in \mathbb{N}$ and for each candidate vector in $\{\hat{w}_j : j \in [m_0 + m_1]\}$ we query $f$ to compute $\Delta_\epsilon f(t_k\hat{w}_j)$ for few steps $\{t_k : k \in [T]\}$ in order to approximate

$$\|\|\nabla f(t\hat{w}_j)\|_2^2\|_{L_2([-\infty, \infty])} \approx \sum_{k=1}^{T} \|\nabla f(t_k\hat{w}_j)\|^2 \approx \sum_{k=1}^{T} \|\Delta_\epsilon f(t_k\hat{w}_j)\|^2 := \hat{I}(w_j).$$
Then we compute a permutation \( \pi : [m] \to [m] \) to order the weights so that \( \hat{L}(w_{\pi(i)}) \geq \hat{L}(w_{\pi(j)}) \) whenever \( \pi(i) > \pi(j) \). The candidates \( \{w_{\pi(j)} : j = 1, \ldots, m_0\} \) have the slowest decay, respectively, largest norms and are thus assigned to the first layer. The remaining candidates \( \{w_{\pi(\ell)} : \ell = m_0 + 1, \ldots, m_1\} \) are assigned to the second layer.

**Numerical experiments**

We have applied the proposed strategy to assign vectors \( \{\hat{w}_j : j \in [m_0 + m_1]\} \), which are outputs of experiments conducted in Sect. 4, to either the first or the second layer. Since each \( \hat{w}_j \) does not exactly correspond to a vector in \( \{a_i : i \in [m_0]\} \) or \( \{v_\ell : \ell \in [m_1]\} \), we assign a ground truth label \( L_j = 1 \) to \( \hat{w}_j \) if the closest vector to \( \hat{w}_j \) belongs to \( \{a_i : i \in [m_0]\} \), and \( L_j = 2 \) if it belongs to the set \( \{v_\ell : \ell \in [m_1]\} \). Denoting similarly the predicted label \( \hat{L}_j = 1 \) if \( \pi(j) \in \{1, \ldots, m_0\} \) and \( \hat{L}_j = 2 \) otherwise, we compute the success rates

\[
L_1 := \#\{j : L_j = 1 \text{ and } \hat{L}_j = 1\} / m_0, \quad L_2 := \#\{j : L_j = 2 \text{ and } \hat{L}_j = 2\} / m_1
\]  

(44)
to assess the proposed strategy. Hyperparameters are $\epsilon = 10^{-5}$ for the step length in the finite difference approximation $\Delta_{\epsilon} f(\cdot)$, and $t_k = -20 + k$ for $k \in [40]$.

The results for all four scenarios considered in Sect. 4 are reported in Table 1. We see that our simple strategy achieves remarkable success rates, in particular if the network weights in each layer represent perturbed orthogonal systems. If the weights are sampled uniformly from the unit sphere with moderate dimension $d = m_0$, then, as one may expect, the success rate drops. In fact, for small $d = m_0$, the vectors $\{a_i : i \in [m_0]\}$ tend to be less orthogonal, and thus, the assumption $a_i^T a_j \approx 0$ for $i \neq j$ is not satisfied anymore.

Finally, we stress that the proposed strategy is simple, efficient and relies only on few additional point queries of $f$ that are negligible compared to the recovery step itself (for reasonable query size $T$). In fact, the method relies on a single (nonlinear) feature of the map $t \mapsto \|\nabla f(t\hat{w}_j)\|_2$ in order to decide upon the label of $\hat{w}_j$. We identify it as an interesting future investigation to develop more robust approaches, potentially using higher-dimensional features of trajectories $t \mapsto \|\nabla f(t\hat{w}_j)\|_2$, to achieve high success rates even if $a_i^T a_j \approx 0$ for $i \neq j$ may not hold anymore.

### 6.2 Reconstructing the Network Function Using Gradient Descent

The previous section allows assigning unlabeled candidates $\{\hat{w}_j : j \in [m_0 + m_1]\}$ to either the first or second layer, resulting in matrices $\hat{A} = [\hat{a}_1 | \ldots | \hat{a}_{m_0}]$ and $\hat{V} = [\hat{v}_1 | \ldots | \hat{v}_{m_1}]$ that ideally approximate $A$ and $V$ up to column signs and permutations. Assuming that the network $f \in F(m_0, m_0, m_1)$ is generated by shifts of one activation function, i.e., $g_i(t) = \phi(t + \theta_i)$ and $h_\ell(t) = \phi(t + \tau_\ell)$ for some $\phi$, this means only signs, permutations, and bias vectors $\theta \in \mathbb{R}^{m_0}, \tau \in \mathbb{R}^{m_1}$ are missing to fully reconstruct $f$. In this section, we show how to identify these remaining parameters by applying a gradient descent method to minimize the least squares of the output misfit of the deparametrized network. In fact, as we clarify below, the original network $f$ can be explicitly described as a function of the known entangled weights $a_i$ and $v_\ell$ and of the unknown remaining parameters (signs, permutations, and biases), see Proposition 26 and Corollary 27.

Let now $\mathcal{D}_m$ denote the set of $m \times m$ diagonal matrices, and define a parameter space $\Omega := \mathcal{D}_{m_1} \times \mathcal{D}_{m_0} \times \mathcal{D}_{m_0} \times \mathbb{R}_+^{m_0} \times \mathbb{R}_+^{m_1}$. To reconstruct the original network $f$, we propose to fit parameters $(D_1, D_2, D_3, w, z) \in \Omega$ of a function $\hat{f} : \mathbb{R}_+^{m_0} \times \Omega \to \mathbb{R}$ defined by

$$\hat{f}(x; D_1, D_2, D_3, w, z) = 1^T \phi(D_1 \hat{V}^T \hat{A}^{-T} D_2 \phi(D_3 \hat{A}^T x + w) + z)$$

to a number of additionally sampled points $\{(X_i, Y_i) : i \in [m_f]\}$ where $Y_i = f(X_i)$ and $X_i \sim \mathcal{N}(0, \mathbf{I}_{m_0})$. The parameter fitting can be formulated as solving the least
squares

\[
\min_{(D_1, D_2, D_3, w, z) \in \Omega} J(D_1, D_2, D_3, w, z) := \sum_{i=1}^{m_f} \left( Y_i - \hat{f}(X_i; D_1, D_2, D_3, w, z) \right)^2.
\]

(45)

We note that, due to the identification of the entangled weights and deparametrization of the problem, \( \dim(\Omega) = 3m_0 + 2m_1 \), which implies that the least squares has significantly fewer free parameters compared to the number \( m_0^2 + (m_0 \times m_1) + (m_0 + m_1) \) of original parameters of the entire network. Hence, our previous theoretical results of Sect. 3 and numerical experiments of Sect. 4 greatly scale down the usual effort of fitting all parameters at once. We may also mention at this point that the optimization (45) might have multiple global solutions due to possible symmetries, see also [20] and Remark 28, and we shall try to keep into account the most obvious ones in our numerical experiments.

We will now show that there exists parameters \((D_1, D_2, D_3, w, z) \in \Omega\) that allow for exact recovery of the original network, whenever \(A\) and \(V\) are correct up to signs and permutation. We first need the following proposition that provides a different reparametrization of the network using \(\hat{A}\) and \(\hat{V}\). The proof of the proposition requires only elementary linear algebra and properties of sign and permutation matrices. Details are deferred to Appendix 3.

**Proposition 26** Let \(f \in \mathcal{F}(m_0, m_0, m_1)\) with \(g_i(t) = \phi(t + \theta_i)\) and \(h_\ell(t) = \phi(t + \tau_\ell)\), and define the function \(\tilde{f} : \mathbb{R}^{m_0} \times D_{m_0} \times D_{m_1} \times \mathbb{R}^{m_0} \times \mathbb{R}^{m_1} \to \mathbb{R}\) via

\[
\tilde{f}(x; D, D', w, z) = 1^T \phi(D' \hat{B}^T \phi(D \hat{A}^T x + w) + z), \quad \text{with} \quad \hat{b}_i := \text{diag} \left( (\phi'(w))^{-1} \right) D \hat{A}^{-1} \hat{v}_\ell.
\]

If there are sign matrices \(S_A, S_V\), and permutations \(\pi_A, \pi_V\) such that \(A\pi_A = \hat{A}S_A\), \(V\pi_V = \hat{V}S_V\), then we have \(f(x) = \tilde{f}(x; S_A, S_V, \pi_A^T \theta, \pi_V^T \tau)\).

We note here that replacing \(\hat{f}\) by \(\tilde{f}\) in (45) is tempting because it further reduces the number of parameters (\(\dim(D_{m_0} \times D_{m_1} \times \mathbb{R}^{m_0} \times \mathbb{R}^{m_1}) = 2(m_0 + m_1)\)), but, by an explicit computation, one can show that evaluating the gradient of \(\tilde{f}\) with respect to \(D\) requires also the evaluation of \(D^{-1}\). Having in mind that \(D\) ideally converges to \(S_A\) during the optimization, diagonal entries of \(D\) are likely to cross zero while optimizing. Thus such minimization may result unstable, and we instead work with \(\hat{f}\). The following Corollary shows that also this form allows finding optimal parameters leading to the original network.

**Corollary 27** Let \(f \in \mathcal{F}(m_0, m_0, m_1)\) with \(g_i(t) = \phi(t + \theta_i)\) and \(h_\ell(t) = \phi(t + \tau_\ell)\). If there exist sign matrices \(S_A, S_V\), and permutations \(\pi_A, \pi_V\) such that \(A\pi_A = \hat{A}S_A\), \(V\pi_V = \hat{V}S_V\), there exist diagonal matrices \(D_1, D_2\) such that \(f(x) = \hat{f}(x; D_1, D_2, S_A, \pi_A^T \theta, \pi_V^T \tau)\).
Based on Proposition 26, we can rewrite $f(x) = 1^T \phi(S_V \hat{B}^T \phi(S_A \hat{A}^T x + \pi_A^T w) + \pi_V^T z)$, so it remains to show that $S_V \hat{B}^T = D_1 \hat{V}^T \hat{A}^{-T} D_2$ for diagonal matrices $D_1, D_2$. First we note

$$\text{diag}(\phi'(\pi_A^T \theta)^{-1}) = \pi_A^T \text{diag}(\phi'(\theta)^{-1}) \pi_A = \pi_A^T G^{-1} \pi_A.$$  

Using this, and $D = S_A$ in the definition of $\hat{B}$ in Proposition 26, it follows that

$$\hat{B}^T = \text{diag}(\|\pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} v_1\|, \ldots, \|\pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} v_m\|) \hat{V}^T \hat{A}^{-T} S_A \pi_A^T G^{-1} \pi_A.$$  

Multiplying by $S_V$ from the left, we obtain

$$S_V \hat{B}^T = S_V \text{diag}(\|\pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} v_1\|, \ldots, \|\pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} v_m\|) = D_1 \hat{V}^T \hat{A}^{-T} S_A \pi_A^T G^{-1} \pi_A = D_2.$$  

**Remark 28** (Simplification for odd functions) If $\phi$ in Proposition 26 satisfies $\phi(-t) = -\phi(t)$, then $\hat{f}(x; D_1, S D_2, S D_3, S w, z) = \hat{f}(x; D_1, D_2, D_3, w, z)$ for arbitrary sign matrix $S \in \mathcal{D}_{m_0}$. Thus, choosing $S = S_A$, there are also diagonal $D_1$ and $D_2$ with $f(x) = \hat{f}(x; D_1, D_2, \text{Id}_{m_0}, S_A \pi_A^T w, \pi_V^T \tau)$.

Assuming $\hat{A}$ and $\hat{V}$ are correct up to sign and permutation, Corollary 27 implies that $J = 0$ is the global optimum, and it is attained by parameters leading to the original network $f$. Furthermore, Remark 28 implies that there is ambiguity with respect to $D_3$, if $\phi$ is an odd function. Thus we can also prescribe $D_3 = \text{Id}_{m_0}$ and neglect optimizing this variable if $\phi$ is odd.

We now study numerically the feasibility of (45). First, we consider the case $\hat{A} = A$ and $\hat{V} = V$ to assess (45), isolated so not to suffer possible errors from other parts of our learning procedure (see Sects. 4 and 6.1). Afterwards we take into consideration also these additional approximations, and present results for $\hat{A} \approx A$ and $\hat{V} \approx V$.

We minimize (45) by standard gradient descent and learning rate 0.5 if $\phi(t) = \frac{1}{1 + e^{-t}} - \frac{1}{2}$ (shifted sigmoid), respectively, learning rate 0.025 if $\phi(t) = \tanh(t)$. We sample $m_f = 10(m_0 + m_1)$ additional points, which is only slightly more than the number of free parameters. Gradient descent is run for 500K iterations (due to small number of variables, this is not time consuming), and only prematurely stopped it, if the iteration stalls. Initially we set $D_2 = D_3 = \text{Id}_{m_0}$, and all other variables are set to random draws from $N(0, 0.1)$.

Denoting $\omega^* = (D_1^*, D_2^*, D_3^*, w^*, z^*) \in \Omega$ as the gradient descent output, we measure the relative mean squared error (MSE) and the relative $L_\infty$-error.

$$\text{MSE} = \frac{\sum_{i=1}^{m_{\text{test}}} (\hat{f}(Z_i; \omega^*) - f(Z_i))^2}{\sum_{i=1}^{m_{\text{test}}} f(Z_i)^2}, \quad E_\infty = \max_{i \in [m_{\text{test}}]} \left| \frac{\hat{f}(Z_i; \omega^*) - f(Z_i)}{f(Z_i)} \right|.$$  

$\odot$ Springer
using $m_{test} = 50,000$ samples $Z_i \sim \mathcal{N}(0, |d_{m_0}|)$. Moreover, we also report the relative bias errors

$$ E_{\theta} = \frac{\| w^* - \theta \|_2^2}{\| \theta \|_2^2}, \quad E_{\eta} = \frac{\| z^* - \eta \|_2^2}{\| \eta \|_2^2}, $$

which indicate if the original bias vectors are recovered. We repeat each experiments 30 times and report averaged values.

Table 2 presents the results of the experiments and shows that we reconstruct a network function that is very close to the original network $f$ in both $L_2$ and $L_\infty$ norm, and in every scenario. The maximal error is $\approx 10^{-3}$, which is likely further reducible by increasing the number of gradient descent iterations, or using finer-tuned learning rates or acceleration methods. Therefore, the experiments strongly suggest that we are indeed reconstructing a function that approximates $f$ uniformly well. Inspecting the errors $E_{\theta}$ and $E_{\eta}$ also supports this claim, at least in all scenarios where the tanh activation is used. In many cases, the relative errors are below $10^{-7}$, implying that we recover the original bias vectors of the network. Surprisingly, the accuracy of recovered biases slightly drops of few orders of magnitude in the sigmoid case, despite convincing results when measuring predictive performance in $L_2$ and $E_{\infty}$. We believe that this is due to faster flattening of the gradients around the stationary point compared to the case of a tanh activation function and that it can be improved by using
Table 3  Errors of the reconstructed network using (45) when using approximated $\hat{A} \approx A$ and $\hat{V} \approx V$ (up to sign and permutation)

| Scenario | $m_0 = 30$ | | | | $m_0 = 45$ | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | $m_1 = 3$ | $m_1 = 9$ | $m_1 = 15$ | $m_1 = 5$ | $m_1 = 14$ | $m_1 = 23$ | |
| POD/sig | MSE | $6.4e^{-5}$ | $3.1e^{-2}$ | $2.9e^{-2}$ | $6.8e^{-5}$ | $1.1e^{-2}$ | $5.4e^{-4}$ | |
| | $E_\infty$ | $1.1e^{-2}$ | $7.9e^{-2}$ | $9.0e^{-2}$ | $1.4e^{-2}$ | $4.5e^{-2}$ | $4.7e^{-2}$ | |
| | Trials [%] | 63 | 80 | 90 | 37 | 67 | 93 | |
| POD/tanh | MSE | — | $2.7e^{-2}$ | $8.9e^{-3}$ | — | $4.2e^{-3}$ | $7.7e^{-3}$ | |
| | $E_\infty$ | — | $1.9e^{-1}$ | $1.2e^{-1}$ | — | $8.3e^{-2}$ | $9.7e^{-2}$ | |
| | Trials [%] | 0 | 23 | 76 | 0 | 43 | 96 | |

Trials indicates the percentage of repetitions where $\hat{A}$ and $\hat{V}$ satisfy (46). The scenarios correspond to those in Sects. 4 and 6.1

more sophisticated strategies of choosing a gradient descent step size. We also tested (45) when fixing $D = \text{Id}_{m_0}$ since tanh and the shifted sigmoid are odd functions, and thus Remark 28 applies. The results are consistently slightly better than Table 2, but are qualitatively similar.

We ran similar experiments for perturbed orthogonal weights and when using $\hat{A}$ and $\hat{V}$ precomputed with the methods we described in Sects. 4 and 6.1. The quality of the results varies dependent on whether $\hat{A} \approx A$ and $\hat{V} \approx V$ (up to sign and permutation) holds, or a fraction of the weights has not been recovered. To isolate cases where $\hat{A} \approx A$ and $\hat{V} \approx V$ holds, we compute averaged MSE and $L_\infty$ over all trials satisfying

$$
\sum_{i=1}^{m_0} \mathcal{E}(a_i) + \sum_{\ell=1}^{m_1} \mathcal{E}(v_\ell) < 0.5, \quad \text{(see Section 4 for the Definition of $\mathcal{E}$).} \quad (46)
$$

We report the averaged errors and the number of trials satisfying this condition in Table 3. It shows that the reconstructed function is close to the original function, even if the weights are only approximately correct. Therefore we conclude that that minimizing (45) provides a very efficient way of learning the remaining network parameters from just few additional samples, once entangled network weights $A$ and $V$ are (approximately) known.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.
Appendix

The following Lemma implies that \{a_1 \otimes a_1, \ldots, a_{m_0} \otimes a_{m_0}, v_1 \otimes v_1, \ldots, v_{m_1} \otimes v_{m_1}\} satisfying the properties of Definition 3 is a system of linearly independent matrices.

Lemma 29 Let \{z_1, \ldots, z_m\} \subset \mathbb{R}^m have unit norm and satisfy \(\sum_{j=1}^{m} |z_j|^2 \leq C_F\) for all \(j = 1, \ldots, m\). If \(1 < C_F < 2\), the system \(\{\otimes z_1, \ldots, z_m \otimes z_m\}\) is linearly independent.

Proof Assume to the contrary the \(\{\otimes z_1, \ldots, z_m \otimes z_m\}\) are not linearly independent, then there exists \(\sigma \neq 0 \in \mathbb{R}^m\) with \(0 = \sum_{j=1}^{m} \sigma_j z_j \otimes z_j\), or equivalently \(0 = \sum_{j=1}^{m} \sigma_j (x, z_j)^2\) for all \(x \in \mathbb{R}^d\). Without loss of generality assume \(\|\sigma\|_\infty = \max_i \sigma_i\) (otherwise we multiply the representation by \(-1\)), and denote by \(i^*\) the index achieving the maximum. Then we have

\[
0 = \sum_{i=1}^{N} \sigma_i \langle z_i, z_{i^*} \rangle^2 = \sigma_{i^*} + \sum_{i \neq i^*} \sigma_i \langle z_i, z_{i^*} \rangle^2 \geq \sigma_{i^*} + \min_i \sigma_i \sum_{i \neq i^*} \langle z_i, z_{i^*} \rangle^2,
\]

where we used \(\|z_{i^*}\| = 1\). Since \(\min_i \sigma_i \geq 0\) immediately yields a contradiction, we continue with the case \(\min_i \sigma_i < 0\). We can further bound

\[
0 \geq \sigma_{i^*} + \min_i \sigma_i \sum_{i \neq i^*} \langle z_i, z_{i^*} \rangle^2 \geq \sigma_{i^*} + \min_i \sigma_i (C_F - 1) \|z_{i^*}\|^2
\]

and by division through \((C_F - 1)\) and subtracting \(\min_i \sigma_i\) we obtain \(|\min_i \sigma_i| \geq \sigma_{i^*} (C_F - 1)^{-1}\). Since \((C_F - 1)^{-1} > 1\), this yields the contradiction \(\|\sigma\|_\infty \geq |\min_i \sigma_i| > \sigma_{i^*} = \|\sigma\|_\infty\).

The linear independence of the system \(\{a_1 \otimes a_1, \ldots, a_{m_0} \otimes a_{m_0}, v_1 \otimes v_1, \ldots, v_{m_1} \otimes v_{m_1}\}\) implies that it is a Riesz basis for \(\mathcal{W} := \text{span}\{a_1 \otimes a_1, \ldots, a_{m_0} \otimes a_{m_0}, v_1 \otimes v_1, \ldots, v_{m_1} \otimes v_{m_1}\}\). As such there exists constants \(c_r, C_R\) such that for every \(\sigma \in \mathbb{R}^{m_0 + m_1}\)

\[
c_r \|\sigma\|_2^2 \leq \left\| \sum_{i=1}^{m_0} \sigma_i a_i \otimes a_i + \sum_{i=1}^{m_1} \sigma_{m_0+i} v_i \otimes v_i \right\|^2 \leq C_R \|\sigma\|_2^2. \tag{47}
\]

Additional Proofs for Section 2

Proof of Lemma 7 Fix any pair \(k, n \in [d]\) and define \(\phi(t) = f(x + te_k + e_n) - f(x + te_k)\), where \(e_k\) denotes the \(k\)-th standard vector. By the mean value theorem and for
Assume $k, n$ to be fixed and denote $\bar{x} = x + \xi_1 e_k + \xi_2 e_n$. By recalling our definition of $\nabla^2 f(x)$ in (10), it follows $\frac{\partial^2 f}{\partial x_k \partial x_n}(x) = \varphi_1(x) + \varphi_2(x)$, where

\[
\varphi_1(x) := \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} h_{ij}^\ell (b_{ij}^\ell g(A^T x)) g_i^\ell (a_i^T x) a_{ki} a_{nj} b_{i\ell} b_{j\ell},
\]

\[
\varphi_2(x) := \sum_{\ell=1}^{m_1} \sum_{i=1}^{m_0} h_{ij}^\ell (b_{ij}^\ell g(A^T x)) g_i''(a_i^T x) a_{ki} a_{mi} b_{i\ell}.
\]

Thus

\[
|\nabla^2 f(x)|_{kn} - |\Delta^2 [f](x)|_{kn} \leq |\varphi_1(x) - \varphi_1(\bar{x})| + |\varphi_2(x) - \varphi_2(\bar{x})|.
\]

As before, we start by applying the Lipschitz continuity to the summands of $|\varphi_1(x) - \varphi_1(\bar{x})|:

\[
\begin{align*}
&|h_{ij}^\ell (b_{ij}^\ell g(A^T x)) g_i^\ell (a_i^T x) a_{ki} a_{nj} b_{i\ell} b_{j\ell} - h_{ij}^\ell (b_{ij}^\ell g(A^T \bar{x})) g_i^\ell (a_i^T \bar{x}) a_{ki} a_{nj} b_{i\ell} b_{j\ell}| \\
&\leq \eta_2 |g_i(a_i^T x) a_{ki} a_{nj} b_{i\ell} - g_i(a_i^T \bar{x}) a_{ki} a_{nj} b_{i\ell}| \\
&+ \kappa_1^2 |h_{ij}^\ell (b_{ij}^\ell g(A^T x)) - h_{ij}^\ell (b_{ij}^\ell g(A^T \bar{x}))| \\
&\leq \eta_2 \left[ |g_i(a_i^T x) - g_i(a_i^T \bar{x})| a_{ki} a_{nj} b_{i\ell} b_{j\ell} + \kappa_1^2 \right] \\
&\leq \eta_2 \left[ \sum_{I=1}^{m_0} b_{I\ell} \left( g_i(a_i^T x) - g_i(a_i^T \bar{x}) \right) - \kappa_1 \eta_3 \right] \\
&+ \kappa_1 \eta_3 \sum_{I=1}^{m_0} b_{I\ell} \left( g_i(a_i^T x) - g_i(a_i^T \bar{x}) \right) \\
&\leq \eta_2 \left[ \sum_{I=1}^{m_0} b_{I\ell} \left( g_i(a_i^T x) - g_i(a_i^T \bar{x}) \right) + \kappa_1 \eta_3 \right] \\
&\leq \eta_2 \left[ \sum_{I=1}^{m_0} b_{I\ell} \left( g_i(a_i^T x) - g_i(a_i^T \bar{x}) \right) + \kappa_1 \eta_3 \right]
\end{align*}
\]
\[
\leq \eta_2 \left[ \kappa_1 \left| g'_i(a_i^T x) - g'_i(a_i^T \tilde{x}) \right| + \kappa_1 \left| g'_j(a_j^T x) - g'_j(a_j^T \tilde{x}) \right| \right] + \kappa_1^2 \eta_3 \sum_{l=1}^{m_0} b_{l\ell} a_i^T (x - \tilde{x})
\]
\[
\leq \eta_2 \kappa_1 \kappa_2 \left[ |a_i^T (x - \tilde{x})| + |a_j^T (x - \tilde{x})| \right] + \kappa_1^3 \eta_3 \sum_{l=1}^{m_0} b_{l\ell} a_i^T (x - \tilde{x})
\]
\[
\leq \eta_2 \kappa_1 \kappa_2 \left[ |\xi_1 a_{ki} + \xi_2 a_{ni}| + |\xi_1 a_{kj} + \xi_2 a_{nj}| \right] + \kappa_1^3 \eta_3 \sum_{l=1}^{m_0} b_{l\ell} (|a_{ki}| + |a_{nj}|)
\]
\[
\leq \tilde{C} \epsilon \left[ |a_{ki}| + |a_{ni}| + |a_{kj}| + |a_{nj}| + \sum_{l=1}^{m_0} |b_{l\ell}| (|a_{ki}| + |a_{nj}|) \right].
\]

where \( \tilde{C} = \max \{ \eta_2 \kappa_1 \kappa_2, \kappa_1^3 \eta_3 \} \). Hence,
\[
|\varphi_1(x) - \varphi_1(\tilde{x})| \leq \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \tilde{C} \epsilon \left[ |a_{ki}| + |a_{ni}| + |a_{kj}| + |a_{nj}| + \sum_{l=1}^{m_0} |b_{l\ell}| (|a_{ki}| + |a_{nj}|) \right] |b_{i\ell} b_{j\ell} a_{ki} a_{nj}|.
\]

Now
\[
\sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \tilde{C} \epsilon \left[ |a_{ki}| + |a_{ni}| + |a_{kj}| + |a_{nj}| + \sum_{l=1}^{m_0} |b_{l\ell}| (|a_{ki}| + |a_{nj}|) \right] |b_{i\ell} b_{j\ell} a_{ki} a_{nj}|
\]
\[
= \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \tilde{C} \epsilon \left[ |a_{ki}^2 a_{nj}| + |a_{ni} a_{ki} a_{nj}| + |a_{kj} a_{ki} a_{nj}| + |a_{nj}^2 a_{ki}| \right] |b_{i\ell} b_{j\ell}|.
\]

Applying the triangle inequality of the Frobenius norm results in
\[
\tilde{C} \epsilon \left( \sum_{k,n=1}^{d} \left[ \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \left[ |a_{ki}^2 a_{nj}| + |a_{ni} a_{ki} a_{nj}| + |a_{kj} a_{ki} a_{nj}| + |a_{nj}^2 a_{ki}| \right] |b_{i\ell} b_{j\ell}| \right] \right)^{\frac{1}{2}}
\]
\[
\leq 2\tilde{C} \epsilon \left( \sum_{k,n=1}^{d} \left[ \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \left[ a_{ki}^2 a_{nj} \right] |b_{i\ell} b_{j\ell}| \right] \right)^{\frac{1}{2}}
\]
\[
+ 2\tilde{C} \epsilon \left( \sum_{k,n=1}^{d} \left[ \sum_{\ell=1}^{m_1} \sum_{i,j=1}^{m_0} \left[ a_{ni} a_{ki} a_{nj} \right] |b_{i\ell} b_{j\ell}| \right] \right)^{\frac{1}{2}}
\]
\[ \leq 2\tilde{C}\varepsilon \sum_{\ell=1}^{m_{1}} \sum_{i,j=1}^{m_{0}} |b_{i\ell}b_{j\ell}| \left( \sum_{k,n=1}^{d} |a_{k\ell}^{2}a_{n\ell}|^{2} \right)^{\frac{1}{2}} \]

\[ + 2\tilde{C}\varepsilon \sum_{\ell=1}^{m_{1}} \sum_{i,j=1}^{m_{0}} |b_{i\ell}b_{j\ell}| \left( \sum_{k,n=1}^{d} |a_{n\ell}a_{k\ell}a_{n\ell}|^{2} \right)^{\frac{1}{2}} \]

\[ \leq 2\tilde{C}\varepsilon \sum_{\ell=1}^{m_{1}} \sum_{i=1}^{m_{0}} |b_{i\ell}| \sum_{j=1}^{m_{0}} |b_{j\ell}| \left\{ \left( \sum_{k,n=1}^{d} |a_{k\ell}^{2}a_{n\ell}|^{2} \right)^{\frac{1}{2}} + \left( \sum_{k,n=1}^{d} |a_{n\ell}a_{k\ell}a_{n\ell}|^{2} \right)^{\frac{1}{2}} \right\} \]

\[ \leq 2\tilde{C}\varepsilon \sum_{\ell=1}^{m_{1}} \sum_{i=1}^{m_{0}} |b_{i\ell}| \sum_{j=1}^{m_{0}} |b_{j\ell}| \|a_{i\ell}\|\|a_{j\ell}\| \]

\[ \leq 4\tilde{C}\varepsilon \sum_{\ell=1}^{m_{1}} \|b_{\ell\ell}\|^{2} \]

\[ \leq 4\tilde{C}\varepsilon m_{1}m. \]

The last inequalities are due to \( \|a_{i\ell}\| = 1 \) for all \( i \in [m_{0}] \) and \( \|b_{\ell\ell}\| \leq \sqrt{m_{0}}\|b_{\ell\ell}\| = \sqrt{m_{0}} \) for all \( \ell \in [m_{1}] \). A similar computation yields

\[ \tilde{C}\varepsilon \left( \sum_{k,n=1}^{d} \left[ \sum_{\ell=1}^{m_{1}} \sum_{i,j=1}^{m_{0}} \sum_{l=1}^{m_{0}} (|a_{k\ell}| + |a_{n\ell}|) |a_{k\ell}a_{n\ell}| |b_{I\ell}b_{I\ell}b_{j\ell}| \right] \right)^{\frac{1}{2}} \leq 4\tilde{C}\varepsilon m_{1}m_{0}^{3}. \]

Combining both results gives

\[ \left\{ \sum_{k,n=1}^{d} \left| \phi_{1}(x) - \phi_{1}(x + \xi_{1,kn}e_{k} + \xi_{2,kn}e_{n}) \right|^{2} \right\}^{\frac{1}{2}} \leq 8\tilde{C}\varepsilon m_{1}m_{0}^{3}. \]

Here we denote \( \xi_{1,kn}, \xi_{2,kn} \), to make clear that \( \xi_{1}, \xi_{2} \) are changing for every partial derivative of second order. However, all \( \xi_{1,kn}, \xi_{2,kn} \) are bounded by \( \varepsilon \), so our result still holds. Applying the same procedure to \( |\phi_{2}(x) - \phi_{2}(\tilde{x})| \) yields

\[ \left| h_{\ell}^{\prime}(b_{\ell}^{T}g(A^{T}x))g_{i}^{\prime\prime}(a_{i}^{T}x) - g_{\ell}^{\prime}(b_{\ell}g(A^{T}x))g_{i}^{\prime\prime}(a_{i}^{T}\tilde{x}) \right| \]

\[ \leq \eta_{1}\eta_{3}\varepsilon \left( |a_{k\ell}| + |a_{n\ell}| + \kappa_{1}\kappa_{2}\eta_{1}\varepsilon \sum_{l=1}^{m_{1}} |b_{I\ell}(|a_{I\ell}| + |a_{I\ell}|) |. \right. \]
By setting \( \hat{\mathcal{C}} = \max \{ \eta_1 \kappa_3, \kappa_1 \kappa_2 \eta_2 \} \), we can can develop the same bounds for both parts of the right sum as for \( \varphi_1 \), and get

\[
\left\{ \sum_{k,n=1}^{d} \left| \varphi_2(x) - \varphi_2(x + \xi_{1,kn} e_k + \xi_{2,kn} e_n) \right|^2 \right\}^{\frac{1}{2}} \leq \hat{\mathcal{C}} \epsilon m_1 m_0^\frac{3}{2}.
\]

Finally, we get

\[
\| \nabla^2 f(x) - \Delta^2 f(x) \|_F \leq \left\{ \sum_{k,n=1}^{d} \left| \frac{\partial^2 \hat{f}}{\partial x_k \partial x_n}(x) - \frac{\partial^2 x}{\partial \xi_{1,kn} e_k + \xi_{2,kn} e_n} \right|^2 \right\}^{\frac{1}{2}}
\]

\[
= \left\{ \sum_{k,n=1}^{d} \left| \varphi_1(x) + \varphi_2(x) - \varphi_1(x + \xi_{1,kn} e_k + \xi_{2,kn} e_n) - \varphi_2(x + \xi_{1,kn} e_k + \xi_{2,kn} e_n) \right|^2 \right\}^{\frac{1}{2}}
\]

\[
\leq \hat{\mathcal{C}} \epsilon m_1 m_0^\frac{3}{2} + 2 \hat{\mathcal{C}} \epsilon m_1 m_0^\frac{3}{2}.
\]

Setting \( C_\Delta = 16 \max \{ \hat{\mathcal{C}}, \hat{\mathcal{C}} \} \) finishes the proof.

**Additional Results and Proofs for Section 3**

**Lemma 30** Let \( \{w_\ell \otimes w_\ell : \ell \in [m]\} \) be a set of \( m < 2d - 1 \) rank one matrices in \( \mathbb{S} \) such that any subset of \( [m/2] + 1 \) vectors \( \{w_\ell : j \in \lfloor [m/2] + 1 \rfloor \} \) is linearly independent. Then for any \( X \in \text{span} \{w_\ell \otimes w_\ell : \ell \in [m]\} \cap \mathbb{S} \) with \( \text{rank}(X) = 1 \), there exists \( \ell^* \) such that \( X = w_{\ell^*} \otimes w_{\ell^*} \).

**Proof** Let \( X = \sum_{\ell=1}^{m} \alpha_\ell w_\ell \otimes w_\ell \in \mathbb{S} \), and denote \( I = \{ \ell \in [m] : \alpha_\ell \neq 0 \} \). If \( 1 < |I| \leq [m/2] + 1 \), the vectors \( \{w_i : i \in I\} \) are linearly independent, and thus, \( \text{rank}(X) = |I| > 1 \). Otherwise, we split \( I = I_1 \cup I_2 \) with \( |I_1| = [m/2] + 1 \) and \( |I_2| \leq m - [m/2] - 1 \leq m/2 - 1 \). If we accordingly split \( X = X_1 \cup X_2 \) with \( X_j := \sum_{\ell \in I_j} \alpha_\ell w_\ell \otimes w_\ell \), the assumption implies \( \text{rank}(A_1) = [m/2] + 1 \) and \( \text{rank}(A_2) \leq m/2 - 2 \). Since furthermore \( \text{rank}(X) \geq \text{rank}(X_1) - \text{rank}(X_2) \), it follows that \( \text{rank}(X) \geq [m/2] + 1 - (m/2 - 1) \geq 2 \).

**Corollary 31** Assume \( m < 2d - 1 \), and \( \{w_\ell : \ell \in [m]\} \) satisfies the upper frame bound (9) with \( \nu := C_F - 1 \leq \left[ \frac{m}{2} \right]^{-1} \). Then for \( X \in \mathcal{W} \cap \mathbb{S} \) of \( \text{rank}(X) = 1 \), there exists \( \ell^* \) such that \( X = w_{\ell^*} \otimes w_{\ell^*} \).

**Proof** To apply Lemma 30, we establish a lower bound for the size of the smallest linearly dependent subset of \( \{w_\ell : \ell \in [m]\} \), denoted commonly also by \( \text{spark}(\{w_\ell : \ell \in [m]\}) \), see [63]. Following [63], it is bounded from below by

\[
\text{spark}(\{w_\ell : \ell \in [m]\}) \geq \min \{ k : \mu_1(k - 1) \geq 1 \},
\]

where \( \mu_1(k - 1) := \max_{\mathcal{I} \subseteq [m]} \max_{|\mathcal{I}| = k - 1} \sum_{i \in \mathcal{I}} |(w_i, w_j)| \).
Using the frame property (9), we can bound

\[ \mu_1(k - 1) = \max_{I \subset [m]} \max_{\{i \notin I\}} \sum_{i \in I} |\langle w_i, w_j \rangle| \]

\[ \leq \sqrt{k - 1} \max_{I \subset [m]} \max_{\{i \notin I\}} \left( \sum_{i \in I} \langle w_i, w_j \rangle \right)^2 \leq \sqrt{(k - 1) \nu}. \]

Taking additionally into account \( \nu < \left\lceil \frac{m}{2} \right\rceil^{-1} \), it follows that

\[ \text{spark}(\{w_\ell : \ell \in [m]\}) \geq \min\{k : \mu_1(k - 1) \geq 1\} \geq \min\{k : \sqrt{(k - 1) \nu} \geq 1\} \]

\[ = \min \left\{ k : k \geq 1 + \frac{1}{\nu} \right\} > 1 + \left\lceil \frac{m}{2} \right\rceil. \]

The result follows by applying Lemma 30.

**Lemma 32** Let \( \mathcal{W}, \hat{\mathcal{W}} \) be matrix subspaces of equal dimensions (e.g., subspaces of matrices in \( \mathbb{R}^{d \times d} \)) with corresponding orthogonal projections \( P_\mathcal{W}, P_{\hat{\mathcal{W}}} \). Assume \( \delta := \| P_\mathcal{W} - P_{\hat{\mathcal{W}}} \|_F < 1 \). For any \( W \in \mathcal{W} \) we have

\[ \| P_{\hat{\mathcal{W}}}(W) \|_F \leq \| W \|_F \leq (1 - \delta)^{-1} \| P_{\hat{\mathcal{W}}}(W) \|_F. \]

In particular \( P_{\hat{\mathcal{W}}} : \mathcal{W} \to \hat{\mathcal{W}} \) is a bijection.

**Proof** The right inequality follows by

\[ \| P_{\hat{\mathcal{W}}}(W) \|_F = \| W + P_{\hat{\mathcal{W}}}(W) - P_\mathcal{W}(W) \|_F \geq \left( 1 - \| P_{\hat{\mathcal{W}}} - P_\mathcal{W} \|_F \right) \| W \|_F \geq (1 - \delta) \| W \|_F. \]

**Lemma 33** Let \( \{w_i : i \in [m]\} \subset \mathbb{R}^d \) be a set of unit-norm vectors satisfying the frame assumption

\[ (1 - \nu) \| x \|_2^2 \leq \sum_{i=1}^m \langle x, w_i \rangle^2 \leq (1 + \nu) \| x \|_2^2 \quad \text{for all } x \in \mathbb{R}^d \]

and some \( 0 < \nu < 1 \). Let \( \mathcal{W} = \text{span}\{w_i \otimes w_i : i \in [m]\} \) with corresponding orthogonal projection \( P_\mathcal{W} \) and let \( \hat{\mathcal{W}} \subset \mathbb{R}^{d \times d} \) be a subspace of symmetric matrices with orthogonal projection \( P_{\hat{\mathcal{W}}} \) such that \( \delta := \| P_\mathcal{W} - P_{\hat{\mathcal{W}}} \|_F < 1 \). Define \( \hat{W}_i := P_{\hat{\mathcal{W}}}(w_i) \), take \( M \in \hat{\mathcal{W}} \cap S \) with \( M = \sum_{i=1}^m \sigma_i \hat{W}_i \), and let \( Z \in \mathcal{W} \) satisfy \( M = P_{\hat{\mathcal{W}}}(Z) \).
Then

\[
\|\sigma\|_\infty \leq \frac{1}{(1-\delta)(1-\nu)},
\]

(48)

\[
\sigma_j \left\| \hat{W}_j \right\|_F^2 - \left\langle \hat{W}_j, M \right\rangle \leq \frac{\delta}{1-\delta} + (\delta + \nu) \|\sigma\|_\infty
\]

(49)

Moreover, for any unit norm vector \(v\) and any \(\hat{W}_j\), we have

\[
\left\| \hat{W}_j v \right\|_2^2 - v^T \hat{W}_j v \leq 2\delta.
\]

(50)

**Proof** We first note that \(1 = \|M\|_F = \|P_{\hat{W}}(Z)\|_F \geq (1-\delta) \|Z\|_F\) implies \(\|Z\|_F \leq (1-\delta)^{-1}\). For (48), we assume without loss of generality \(\sigma_k = \|\sigma\|_\infty\) (otherwise we perform the proof for \(-M\)), and denote \(j = \arg\max_i \sigma_i\). Then we have

\[
(1-\delta)^{-1} \geq \|Z\|_F \geq \|Z\| \geq w_j^T Z w_j = \sum_{i=1}^m \sigma_i \langle w_j, w_i \rangle^2 = \|\sigma\|_\infty + \sum_{i \neq j} \sigma_i \langle w_j, w_i \rangle^2
\]

\[
\geq \|\sigma\|_\infty \left(1 - \sum_{i \neq j} \langle w_j, w_i \rangle^2\right) \geq \|\sigma\|_\infty (1 - (C_F - 1)) \geq \|\sigma\|_\infty (1 - \nu).
\]

For (49), we first notice that

\[
\left\langle \hat{W}_j, M \right\rangle = \left\langle \hat{W}_j, \sum_{i=1}^m \sigma_i \hat{W}_i \right\rangle = \sigma_j \left\| \hat{W}_j \right\|_F^2 + \left\langle \hat{W}_j, \sum_{i \neq j} \sigma_i \hat{W}_i \right\rangle
\]

\[
= \sigma_j \left\| \hat{W}_j \right\|_F^2 + \left\langle \hat{W}_j, \sum_{i \neq j} \sigma_i W_i \right\rangle
\]

\[
= \sigma_j \left\| \hat{W}_j \right\|_F^2 + \left\langle \hat{W}_j - W_j, \sum_{i \neq j} \sigma_i W_i \right\rangle + \sum_{i \neq j} \sigma_i \langle W_j, W_i \rangle,
\]

and thus is suffices to bound the last two terms. For the first term, we get

\[
\left\| \hat{W}_j - W_j, \sum_{i \neq j} \sigma_i W_i \right\| \leq \delta \left\| \sum_{i \neq j} \sigma_i W_i \right\|_F \leq \frac{\delta}{1-\delta} + \|\sigma\|_\infty \delta,
\]

and for the second

\[
\left| \sum_{i \neq j} \sigma_i \langle W_j, W_i \rangle \right| \leq \|\sigma\|_\infty \left( \sum_{i \neq j} \langle w_j, w_i \rangle^2 \right) \leq \|\sigma\|_\infty \nu.
\]
For (50), we first rewrite
\[
\left\| \hat{W}_j u \right\|^2 - u^T \hat{W}_j u = \left\| \left( \hat{W}_j^2, u \otimes u \right) - \left( \hat{W}_j, u \otimes u \right) \right\| \leq \hat{W}_j^2 - \hat{W}_j.
\]

Now denote \( \Delta := \hat{W}_j - W_j \). Since \( W_j^2 = W_j \), we have
\[
\left\| \hat{W}_j^2 - \hat{W}_j \right\| = \left\| \left( \Delta + W_j \right)^2 - W_j - \Delta \right\| = \left\| \hat{W}_j \Delta - \Delta \left( \mathbb{I} - W_j \right) \right\| \leq \| \Delta \| \left( \| \hat{W}_j \| + \| \mathbb{I} - W_j \| \right) \leq 2\delta,
\]
since \( \mathbb{I} - W_j \) is a projection matrix onto \( \text{span}\{w_j\}^\perp \).

**Proof of Lemma 15** We first calculate a lower bound for \( \lambda_D \) in terms of the \( \min_i \sigma_i \) by
\[
\lambda_D = \sum_{i=1}^m \sigma_i \left( u_D \otimes u_D, \hat{W}_i \right) = \sum_{i=1}^m \sigma_i \left( u_D \otimes u_D, W_i \right) + \sum_{i=1}^m \sigma_i \left( u_D \otimes u_D, \hat{W}_i - W_i \right)
\]
\[
\geq \sum_{i=1}^m \sigma_i \left( u_D, w_i \right)^2 - \sum_{i=1}^m \sigma_i \left( \hat{W}_i - W_i \right) \right\|_F.
\]
\[
\geq C \min_i \sigma_i - \| P_{\hat{W}_i}(Z) - Z \|_F \geq C \min_i \sigma_i - \frac{\delta}{1 - \delta},
\]
where \( C = c_f \) if \( \min_i \sigma_i > 0 \) and \( C = C_F \) if \( \min_i \sigma_i \leq 0 \). We are left with bounding \( \sigma_j^* := \min \{ \sigma_i \} \). Clearly, if \( \sigma_j^* > 0 \), the result follows immediately. Therefore, we concentrate on the case \( \sigma_j^* \leq 0 \) in the following. We first use (32) to get
\[
\lambda_1 \left( \hat{W}_j^*, M \right) = \left( \hat{W}_j^*, u_1_1 \otimes u_1 \right) = \left( W_j^*, u_1_1 \otimes u_1 \right) + \left( \hat{W}_j^* - W_j^*, u_1_1 \otimes u_1 \right)
\]
\[
\geq \left( w_j^*, u_1 \right)^2 - \| \hat{W}_j^* - W_j^* \| \geq -\delta.
\]

Applying now Lemma 33, and \( \| \hat{W}_j^* \| \geq 1 - \delta \), we obtain from (51)
\[
-\frac{\delta}{\lambda_1} \leq \sigma_j^* \geq \frac{\delta}{\lambda_1(1 - \delta)^2} - \frac{\delta}{(1 - \delta)^2} \| \sigma \|_\infty.
\]
Using this in the previously derived bound for $\lambda_m$, and using $C_F < 1 + \nu$, we have

$$\lambda_D \geq C_F \sigma j^* - \frac{\delta}{1 - \delta} \geq -(1 + \nu) \left( \frac{\delta}{\lambda_1 (1 - \delta)^2} + \frac{\delta}{(1 - \delta)^3} + \frac{(\delta + \nu)}{(1 - \delta)^2} \|\sigma\|_\infty \right) - \frac{\delta}{1 - \delta}.$$

Since $\delta, \nu < 1/4$ we obtain from (48) that $\|\sigma\|_\infty \leq 2$, and

$$\lambda_D \geq -2\delta \lambda_1^{-1} - 3\delta - 2(\delta + \nu) \|\sigma\|_\infty \geq -2\delta \lambda_1^{-1} - 8\delta - 4\nu.$$

**Lemma 34** Let $(A, d)$ be a metric space and $F : A \to A$ be a continuous function. Let $(X_j)_{j \in \mathbb{N}}$ be a sequence generated by $X_j = F^j(X_0)$ for some $X_0 \in A$ and assume $d(X_{j+1}, X_j) \to 0$. Then any convergent subsequence of $(X_j)_{j \in \mathbb{N}}$ converges to a fixed point of $F$.

**Proof** Let $(X_{jk})_{k \in \mathbb{N}}$ be a convergent subsequence of $(X_j)_{j \in \mathbb{N}}$ with limit $\bar{X} = \lim_{k \to \infty} X_{jk}$. Then the subsequence $X_{jk+1}$ satisfies $d(X_{jk+1}, \bar{X}) \leq d(X_{jk+1}, X_{jk}) + d(X_{jk}, \bar{X}) \to 0$ as $k \to \infty$ and thus also $(X_{jk+1})_{k \in \mathbb{N}}$ converges to $\bar{X}$. By construction $X_{jk+1} = F(X_{jk})$. Taking the limit $k \to \infty$ on both sides and using the continuity of $F$, we get

$$\bar{X} = \lim_{k \to \infty} X_{jk+1} = \lim_{k \to \infty} F(X_{jk}) = F \left( \lim_{k \to \infty} X_{jk} \right) = F(\bar{X}).$$

**Proof of Proposition 26**

**Proof of Proposition 26** The first step is to replace first layer weights $A$ by $\hat{A} S_A$. This can be achieved by inserting the permutation $\pi_A$ in the first layer and replacing by $\hat{A} S_A$ according to

$$f(x) = 1^T \phi(B^T \phi(A^T x + \theta) + \tau) = 1^T \phi(B^T \pi_A \phi((A \pi_A)^T x + \pi_A^T \theta) + \tau) = 1^T \phi((\pi_A^T B)^T \phi(S_A \hat{A}^T x + \pi_A^T \theta) + \tau).$$

(52)

Next we need to replace the matrix $\pi_A^T B$ using $V$ respectively $\hat{V}$. Let $n \in \mathbb{R}^{m_1}$ be defined as $n_\ell = \|AGb\|_\ell^{-1}$, and $N = \text{diag}(n)$. By definition of the entangled weights, we have $V = AGBN$, implying the relation $B = G^{-1} A^{-1} V N^{-1}$. Using assumptions $A = \hat{A} S_A \pi_A^T$ and $V = \hat{V} S_V \pi_V^T$, and the properties $S_A^{-1} = S_A, \pi_A^{-1} = \pi_A^T$, it follows that

$$\pi_A^T B = \pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} \hat{V} S_V \pi_V^T N^{-1} = (\pi_A^T G \pi_A)^{-1} S_A \hat{A}^{-1} \hat{V} S_V \pi_V^T N^{-1}$$

Springer
Since $G = \text{diag}(\phi'(\theta))$, we have $\pi_A^T G \pi_A = \text{diag}(\pi_A^T (\phi'(\theta))) = \text{diag}((\phi'(\pi_A^T \theta))) =: \tilde{G}$. Inserting into (52), we get

$$f(x) = 1^T \phi(N^{-1} \pi S_V \hat{V}^T \hat{A}^{-T} S_A \tilde{G}^{-1} \phi(S_A \hat{A}^T x + \pi_A^T \theta) + \tau)$$

The dot product with a 1-vector is permutation invariant; hence, we can get an additional $\pi_V^T$ into the second layer. Then, using that the diagonal matrix $\tilde{N} := \pi_V^T N \pi_V$ commutes with $S_V$ we get

$$f(x) = 1^T \phi(\tilde{N} S_V \hat{V}^T \hat{A}^{-T} S_A \tilde{G}^{-1} \phi(S_A \hat{A}^T x + \pi_A^T \theta) + \pi_V^T \tau)$$

$$= 1^T \phi(\tilde{N}^{-1} S_V \hat{V}^T \hat{A}^{-T} S_A \tilde{G}^{-1} \phi(S_A \hat{A}^T x + \pi_A^T \theta) + \pi_V^T \tau)$$

$$= 1^T \phi(S_V \tilde{N}^{-1} \hat{V}^T \hat{A}^{-T} S_A \tilde{G}^{-1} \phi(S_A \hat{A}^T x + \pi_A^T \theta) + \pi_V^T \tau).$$

It remains to show that $\hat{B} = \tilde{G}^{-1} S_A \hat{A}^{-1} \hat{V} \tilde{N}^{-1}$, which is implied if $\tilde{N}_{\ell \ell} = \|\tilde{G}^{-1} S_A \hat{A}^{-1} \hat{v}_\ell\|$. By the normalization property $\|b_\ell\| = 1$ (see Definition 3) and $B = G^{-1} A^{-1} V N^{-1}$, we first have

$$1 = \|b_\ell\| = \frac{\|G^{-1} A^{-1} v_\ell\|}{n_\ell}$$

and thus $n_\ell = \|G^{-1} A^{-1} v_\ell\|$.

Using this, and the assumptions $A^{-1} = \pi_A S_A \hat{A}^{-1}$, $V \pi_V = \hat{V} S_V$, we obtain

$$\tilde{N} = \pi_V^T N \pi_V = \text{diag}(\pi_V^T n) = \text{diag} \left( \left\| G^{-1} A^{-1} v_1 \right\|, \ldots, \left\| G^{-1} A^{-1} v_{m_1} \right\| \right)$$

$$= \text{diag} \left( \left\| G^{-1} A^{-1} (V \pi_V)_1 \right\|, \ldots, \left\| G^{-1} A^{-1} (V \pi_V)_{m_1} \right\| \right)$$

$$= \text{diag} \left( \left\| G^{-1} \pi_A S_A \hat{A}^{-1} \hat{V} S_V)_1 \right\|, \ldots, \left\| G^{-1} \pi_A S_A \hat{A}^{-1} (\hat{V} S_V)_{m_1} \right\| \right)$$

$$= \text{diag} \left( \left\| G^{-1} \pi_A S_A \hat{A}^{-1} \hat{v}_1 \right\|, \ldots, \left\| G^{-1} \pi_A S_A \hat{A}^{-1} \hat{v}_{m_1} \right\| \right),$$

where we used that $S_V$ affects $v_\ell$ only by multiplication with $\pm 1$. The result follows since $\pi_A^T$ is orthogonal and thus $\|G^{-1} \pi_A S_A \hat{A}^{-1} \hat{v}_\ell\| = \|\pi_A^T G^{-1} \pi_A S_A \hat{A}^{-1} \hat{v}_\ell\| = \|\tilde{G}^{-1} S_A \hat{A}^{-1} \hat{v}_\ell\|$. 

References

1. Anandkumar, A., Ge, R., Janzamin, M.: Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates (2014), arXiv:1402.5180
2. Anthony, M., Bartlett, P.L.: Neural Network Learning: Theoretical Foundations. Cambridge University Press, Cambridge (2009)
3. Bach, F.: Breaking the curse of dimensionality with convex neural networks. J. Mach. Learn. Res. 18(1), 629–681 (2017)
4. Benedetto, J.J., Fickus, M.: Finite normalized tight frames. Adv. Comput. Math. 18(2–4), 357–385 (2003)
5. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: Advances in Neural Information Processing Systems, pp. 153–160 (2007)
6. Bhatia, R.: Matrix Analysis, vol. 169. Springer, Berlin (2013)
7. Blum, A., Rivest, R.L.: Training a 3-node neural network is np-complete. In: Advances in Neural Information Processing Systems, pp. 494–501 (1989)
8. Breuel, T.M., Ul-Hasan, A., Al-Azawi, M.A., Shafait, F.: High-performance OCR for printed English and Fraktur using LSTM networks. In: 12th international conference on document analysis and recognition, pp. 683–687 (2013)
9. Bruna, J., Mallat, S.: Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. 35(8), 1872–1886 (2013)
10. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 39–57 (2017)
11. Casazza, P.G., Leonard, N.: Classes of finite equal norm parseval frames. Contemp. Math. 451, 11–32 (2008)
12. Cireşan, D., Meier, U., Masci, J., Schmidhuber, J.: Multi-column deep neural network for traffic sign classification. Neural Netw. 32, 333–338 (2012)
13. Cohen, A., Daubechies, I., DeVore, R., Kerkyacharian, G., Picard, D.: Capturing ridge functions in high dimensions from point queries. Constr. Approx. 35(2), 225–243 (2012)
14. Constantine, P.G.: Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, vol. 2. SIAM (2015)
15. Constantine, P.G., Dow, E., Wang, Q.: Active subspace methods in theory and practice: applications to kriging surfaces. SIAM J. Sci. Comput. 36(4), A1500–A1524 (2014)
16. De Silva, V., Lim, L.-H.: Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM J. Matrix Anal. Appl. 30(3), 1084–1127 (2008)
17. DeVore, R.A., Oskolkov, K., Petrushev, P.: Approximation by feed-forward neural networks. Ann. Numer. Math. 4, 261–288 (1996)
18. Devroye, L., Györfi, L.: Nonparametric Density Estimation: The L1 View. Wiley Interscience Series in Discrete Mathematics. Wiley (1985)
19. Elbrächter, D., Perekrstenko, D., Grohs, P., Bölcskei, H.: Deep neural network approximation theory. IEEE Trans. Inf. Theory 67(5), 2581–2623 (2021)
20. Fefferman, C.: Reconstructing a neural net from its output. Rev. Mat. Iberoam. 10(3), 507–555 (1994)
21. Fiedler, C., Fornasier, M., Klock, T., Rauchensteiner, M.: Stable recovery of entangled weights: towards robust identification of deep neural networks from minimal samples (2021). arXiv:2101.07150
22. Fornasier, M., Schnass, K., Vybiral, J.: Learning functions of few arbitrary linear parameters in high dimensions. Found. Comput. Math. 12(2), 229–262 (2012)
23. Fornasier, M., Vybiral, J., Daubechies, I.: Robust and resource efficient identification of shallow neural networks by fewest samplesgraphic. Inf. Infer. J. IMA (2021). https://doi.org/10.1093/imaiai/iaaa036
24. Foucart, S., Rauhut, H.: A Mathematical Introduction to Compressive Sensing. Applied and Numerical Harmonic Analysis, Birkhäuser, Basel (2013)
25. Gittens, A., Tropp, J.A.: Tail bounds for all eigenvalues of a sum of random matrices (2011). arXiv:1104.4513
26. Golowich, N., Rakhlin, A., Shamir, O.: Size-independent sample complexity of neural networks (2017). arXiv:1712.06541
27. Graves, A., Mohamed, A.-R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645–6649 (2013)
28. Hästads, J.: Tensor rank is NP-complete. J. Algorithms 11(4), 644–654 (1990)
29. Hillar, C.J., Lim, L.-H.: Most tensor problems are NP-hard. J. ACM 60(6), 45 (2013)
30. Hristache, M., Juditsky, A., Spokoiny, V.: Direct estimation of the index coefficient in a single-index model. Ann. Stat. 29, 595–623 (2001)
31. Ichimura, H.: Semiparametric least squares (SLS) and weighted SLS estimation of single-index models. J. Econom. 58(1–2), 71–120 (1993)
32. Janzamin, M., Sedghi, H., Anandkumar, A.: Beating the perils of non-convexity: guaranteed training of neural networks using tensor methods (2015). arXiv:1506.08473
33. Judd, J.S.: Neural Network Design and the Complexity of Learning. MIT Press (1990)
34. Kawaguchi, K.: Deep learning without poor local minima. In: Advances in Neural Information Processing Systems, pp. 586–594 (2016)
35. Kolda, T.G.: Symmetric orthogonal tensor decomposition is trivial (2015). arXiv:1503.01375
36. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
37. Li, K.-C.: On principal hessian directions for data visualization and dimension reduction: another application of Stein’s lemma. J. Am. Stat. Assoc. 87(420), 1025–1039 (1992)
38. Li, X.: Interpolation by ridge polynomials and its application in neural networks. J. Comput. Appl. Math. 144(1–2), 197–209 (2002)
39. Light, W.: Ridge functions, sigmoidal functions and neural networks. In: Approximation Theory VII, pp. 163–206 (1992)
40. Magnus, J.R.: On differentiating eigenvalues and eigenvectors. Econom. Theory 1(2), 179–191 (1985)
41. Mayer, S., Ullrich, T., Vybiral, J.: Entropy and sampling numbers of classes of ridge functions. Constr. Approx. 42(2), 231–264 (2015)
42. Mei, S., Misiakiewicz, T., Montanari, A.: Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. In: Conference on Learning Theory (pp. 2388–2464). PMLR (2019)
43. Mondelli, M., Montanari, A.: On the connection between learning two-layers neural networks and tensor decomposition. In: The 22nd International Conference on Artificial Intelligence and Statistics (pp. 1051–1060). PMLR (2019)
44. Moravčík, M., Schmid, M., et al.: Deepstack: expert-level artificial intelligence in heads-up no-limit poker. Science 356(6337), 508–513 (2017)
45. Nakatsukasa, Y., Soma, T., Uschmajew, A.: Finding a low-rank basis in a matrix subspace. Math. Program. 162(1–2), 325–361 (2017)
46. Petrushev, P.P.: Approximation by ridge functions and neural networks. SIAM J. Math. Anal. 30(1), 155–189 (1998)
47. Pinkus, A.: Approximating by ridge functions. In: Surface Fitting and Multiresolution Methods, pp. 279–292 (1997)
48. Pinkus, A.: Approximation theory of the MLP model in neural networks. Acta Numer. 8, 143–195 (1999)
49. Qu, Q., Sun, J., Wright, J.: Finding a sparse vector in a subspace: linear sparsity using alternating directions. In: Advances in Neural Information Processing Systems, pp. 3401–3409 (2014)
50. Rellich, F., Berkowitz, J.: Perturbation Theory of Eigenvalue Problems. CRC Press (1969)
51. Robeva, E.: Orthogonal decomposition of symmetric tensors. SIAM J. Matrix Anal. Appl. 37(1), 86–102 (2016)
52. Rotksoff, G.M., Vanden-Eijnden, E.: Neural networks as interacting particle systems: asymptotic convexity of the loss landscape and universal scaling of the approximation error (2018). arXiv:1805.00915
53. Shaham, U., Cloninger, A., Coifman, R.R.: Provable approximation properties for deep neural networks. Appl. Comput. Harmon. Anal. 44(3), 537–557 (2018)
54. Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press (2014)
55. Silver, D., Huang, A., Maddison, C.J., et al.: Mastering the game of go with deep neural networks and tree search. Nature 529(7587), 484 (2016)
56. Soudry, D., Carmon, Y.: No bad local minima: data independent training error guarantees for multilayer neural networks (2016). arXiv:1605.08361
57. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: Benchmarking machine learning algorithms for traffic sign recognition. Man Comput. Neural Netw. 32, 323–332 (2012)
58. Stein, C.M.: Estimation of the mean of a multivariate normal distribution. Ann. Stat. 9, 1135–1151 (1981)
59. Stewart, G.W.: Perturbation theory for the singular value decomposition. Technical report (1998)
60. Sturm, I., Lapuschkin, S., Samek, W., Müller, K.-R.: Interpretable deep neural networks for single-trial EEG classification. J. Neurosci. Methods 274, 141–145 (2016)
61. Tao, T.: When are eigenvalues stable? (2008). https://terrytao.wordpress.com/2008/10/28/when-are-eigenvalues-stable/. Accessed 2019-09-29
62. Tao, T.: Topics in Random Matrix Theory, vol. 132. American Mathematical Soc. (2012)
63. Tropp, J.A.: Greed is good: algorithmic results for sparse approximation. IEEE Trans. Inf. Theory 50(10), 2231–2242 (2004)
64. van der Vaart, A.W., Wellner, J.A.: Weak Convergence and Empirical Processes. Springer Series in Statistics. Springer (1996)
65. Vershynin, R.: High-Dimensional Probability: An Introduction with Applications in Data Science, vol. 47. Cambridge University Press (2018)
66. Wedin, P.-Å.: Perturbation bounds in connection with singular value decomposition. BIT Numer. Math. 12(1), 99–111 (1972)
67. Wiatowski, T., Grohs, P., Bölcskei, H.: Energy propagation in deep convolutional neural networks. IEEE Trans. Inf. Theory 64(7), 4819–4842 (2017)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.