Journal of Statistical Mechanics: Theory and Experiment
An IOP and SISSA journal

PAPER: ML 2021

Matrix inference and estimation in multi-layer models*

Parthe Pandit[1,2,], Mojtaba Sahraee-Ardakan[1,2], Sundeep Rangan[3], Philip Schniter[4] and Alyson K Fletcher[1,2]**

[1] Dept. ECE, UC, Los Angeles, CA, United States of America
[2] Dept. Statistics, UC, Los Angeles, CA, United States of America
[3] Dept. ECE, NYU, NY, United States of America
[4] Dept. ECE, The Ohio State University, OH, United States of America
E-mail: parthepandit@ucla.edu, msahraee@ucla.edu, srangan@nyu.edu, schniter.1@osu.edu and akfletcher@ucla.edu

Received 30 October 2021
Accepted for publication 9 November 2021
Published 29 December 2021

Online at stacks.iop.org/JSTAT/2021/124004
https://doi.org/10.1088/1742-5468/ac3a75

Abstract. We consider the problem of estimating the input and hidden variables of a stochastic multi-layer neural network (NN) from an observation of the output. The hidden variables in each layer are represented as matrices with statistical interactions along both rows as well as columns. This problem applies to matrix imputation, signal recovery via deep generative prior models, multi-task and mixed regression, and learning certain classes of two-layer NNs. We extend a recently-developed algorithm—multi-layer vector approximate message passing, for this matrix-valued inference problem. It is shown that the performance of the proposed multi-layer matrix vector approximate message passing algorithm can be exactly predicted in a certain random large-system limit, where the dimensions $N \times d$ of the unknown quantities grow as $N \to \infty$ with d fixed. In the two-layer neural-network learning problem, this scaling corresponds to the case where the number of input features as well as training samples grow to infinity but the

*This article is an updated version of: Pandit P, Sahraee Ardakan M, Rangan S, Schniter P and Fletcher A K 2020 Matrix inference and estimation in multi-layer models *Advances in Neural Information Processing Systems* vol 33 ed H Larochelle, M Ranzato, R Hadsell, M F Balcan and H Lin (New York: Curran Associates) pp 22456–67. Code available at https://github.com/parthe/ML-Mat-VAMP.

**Author to whom any correspondence should be addressed.

number of hidden nodes stays fixed. The analysis enables a precise prediction of the parameter and test error of the learning.

Keywords: inference of graphical models, machine learning, message-passing algorithms, statistical inference

Contents

1. Introduction 3

2. Example applications 5
2.1. Multi-task and mixed regression problems 5
2.2. Sketched clustering 6
2.3. Learning the input layer of a two-layer neural network 6
2.4. Model-based matrix completion 7

3. Multi-layer matrix VAMP 8
3.1. MAP and MMSE inference 8
3.2. Algorithm details 8

4. Analysis in the large system limit 10
4.1. Main result 11

5. Numerical experiments 12

6. Conclusions 14

Acknowledgments 14

Appendix A. State evolution equations 15

Appendix B. Large system limit details 17

Appendix C. Proof of theorem 1 17

Appendix D. General multi-layer recursions 18

Appendix E. Proof of theorem 2 24
E.1. Overview of the induction sequence 24
E.2. Base case: proof of 25
E.3. Inductive step: proof of 25

References 30

J. Stat. Mech. (2021) 124004

1. Introduction

Consider an L-layer stochastic neural network (NN) given by

$$\mathbf{Z}_\ell^0 = \mathbf{W}_\ell \mathbf{Z}_{\ell-1}^0 + \mathbf{B}_\ell + \mathbf{\Xi}_\ell^0, \quad \ell = 1, 3, \ldots, L-1, \tag{1a}$$

$$\mathbf{Z}_\ell^0 = \boldsymbol{\phi}_\ell(\mathbf{Z}_{\ell-1}^0, \mathbf{\Xi}_\ell^0), \quad \ell = 2, 4, \ldots, L, \tag{1b}$$

where, for $\ell = 0, 1, \ldots, L$, we have *true* activations $\mathbf{Z}_\ell^0 \in \mathbb{R}^{n_\ell \times d}$, weights $\mathbf{W}_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$, bias matrices $\mathbf{B}_\ell \in \mathbb{R}^{n_\ell \times d}$, and *true* noise realizations $\mathbf{\Xi}_\ell^0$. The activation functions $\boldsymbol{\phi}_\ell : \mathbb{R}^{n_{\ell-1} \times d} \to \mathbb{R}^{n_\ell \times d}$ are known non-linear functions acting row-wise on their inputs. See figure 1 (TOP). We use the superscript ^{0} in $\mathbf{Z}_\ell^0$ to indicate the true values of the variables, in contrast to estimated values denoted by $\widehat{\mathbf{Z}}_\ell$ discussed later. We model the true values $\mathbf{Z}_0^0$ as a realization of random $\mathbf{Z}_0$, where the rows $\mathbf{z}_{0,i:}^\mathsf{T}$ of $\mathbf{Z}_0$ are i.i.d. with distribution p_0: $p(\mathbf{Z}_0) = \prod_{i=1}^{n_0} p_0(\mathbf{z}_{0,i:})$. Similarly, we also assume that $\mathbf{\Xi}_\ell^0$ are realizations of random $\mathbf{\Xi}_\ell$ with i.i.d. rows $\boldsymbol{\xi}_{\ell,i:}^\mathsf{T}$. For odd ℓ, the rows $\boldsymbol{\xi}_{\ell,i:}$ are zero-mean multivariate Gaussian with covariance matrix $\mathbf{N}_\ell^{-1} \in \mathbb{R}^{d \times d}$, whereas for even ℓ, the rows $\boldsymbol{\xi}_{\ell,i:}$ can be arbitrarily distributed but i.i.d.

Denoting by $\mathbf{Y} := \mathbf{Z}_L^0 \in \mathbb{R}^{n_L \times d}$ the output of the network, we consider the following matrix inference problem:

$$\text{Estimate } \mathbf{Z} := \{\mathbf{Z}_\ell\}_{\ell=0}^{L-1} \quad \text{given } \mathbf{Y} := \mathbf{Z}_L^0 \text{ and } \{\mathbf{W}_{2k-1}, \mathbf{B}_{2k-1}, \boldsymbol{\phi}_{2k}\}_{k=1}^{L/2}. \tag{2}$$

A key feature of the problem we consider here is that the unknowns, $\mathbf{Z}_\ell$, are *matrix-valued* with d columns with statistical dependencies between the columns. As we will see in section 2, the matrix-valued case applies to several problems of broad interest such as matrix imputation, multi-task and mixed regression problems, sketched clustering. We also show that via this formulation we can analyze the learning in two layer NNs under some architectural assumptions.

In many applications, the inference problem can be performed via minimization of an appropriate cost function. For example, suppose the network (1) has no noise $\mathbf{\Xi}_\ell$ for all layers except the final measurement layer, $\ell = L$. In this case, the $\mathbf{Z}_{L-1}^0 = \mathbf{g}(\mathbf{Z}_0^0)$ for some *deterministic function* $\mathbf{g}(\cdot)$ representing the action of the first $L-1$ layers. Inference can then be conducted via a minimization of the form,

$$\widehat{\mathbf{Z}}_{L-1} := \mathbf{g}\left(\arg\min_{\mathbf{Z}_0} H_L(\mathbf{Y}, \mathbf{Z}_{L-1}) + H_0(\mathbf{Z}_0), \text{ subject to } \mathbf{Z}_{L-1} = \mathbf{g}(\mathbf{Z}_0) \right) \tag{3}$$

where the term $H_L(\mathbf{Y}, \mathbf{Z}_{L-1})$ penalizes the prediction error and $H_0(\mathbf{Z}_0)$ is an (optional) regularizer on the network input. For maximum *a posteriori* (MAP) estimation one takes, $H_L(\mathbf{Y}, \mathbf{Z}_{L-1}) = -\log p(\mathbf{Y}|\mathbf{Z}_{L-1})$, and $H_0(\mathbf{Z}_0) = -\log p(\mathbf{Z}_0)$, where the output probability $p(\mathbf{Y}|\mathbf{Z}_{L-1})$ is defined from the last layer of model (1b): $\mathbf{Y} = \mathbf{Z}_L = \boldsymbol{\phi}_L(\mathbf{Z}_{L-1}, \mathbf{\Xi}_L)$. The minimization (3) can then be solved using a gradient-based method. Encouraging results in image reconstruction have been demonstrated in [4, 15, 18, 29, 37, 41, 45]. Markov-chain Monte Carlo algorithms and Langevin diffusion [7, 44] could also be employed for more complex inference tasks.

J. Stat. Mech. (2021) 124004

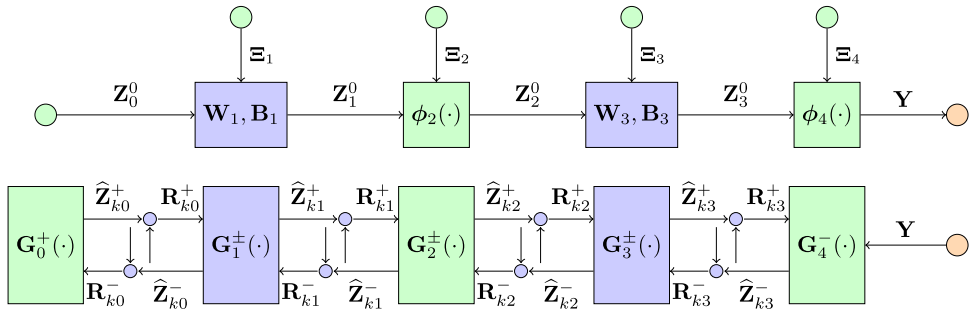

Figure 1. (Top) The signal flow graph for *true* values of matrix variables $\{\mathbf{Z}_\ell^0\}_{\ell=0}^3$, given in equation (1) where $\mathbf{Z}_\ell^0 \in \mathbb{R}^{n_\ell \times d}$. (Bottom) Signal flow graph of the ML-MVAMP procedure in algorithm 1. The variables with superscript + and − are updated in the forward and backward pass respectively. ML-MVAMP (algorithm 1) solves (2) by solving a sequence of simpler estimation problems over consecutive pairs $(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1})$.

However, rigorous analysis of these methods is difficult due to the non-convex nature of the optimization problem. To address this issue, recent works [12, 25, 34] have extended approximate message passing (AMP) methods to provide inference algorithms for the multi-layer networks. AMP was originally developed in [3, 9, 10, 17] for compressed sensing. Similar to other AMP-type results, the performance of multi-layer AMP-based inference can be precisely characterized in certain high-dimensional random instances. In addition, the mean-squared error (MSE) for inference of the algorithms match predictions for the Bayes-optimal inference predicted by various techniques from statistical physics [2, 14, 36]. Thus, AMP-based multi-layer inference provides a computationally tractable estimation framework with precise performance guarantees and testable conditions for optimality in certain high-dimensional random settings.

Prior multi-layer AMP works [12, 16, 26, 34] have considered the case of vector-valued quantities with $d = 1$. The main contribution of this paper is to consider the *matrix-valued* case when $d > 1$. To handle the case when $d > 1$, we extend the multi-layer vector approximate message passing (ML-VAMP) algorithm of [12, 34] to the matrix case. The ML-VAMP method is based on VAMP method of [35], which is closely related to expectation propagation [28, 38], expectation-consistent approximate inference [13, 32], S-AMP [6], and orthogonal AMP [24]. We will use 'multi-layer matrix VAMP (ML-Mat-VAMP)' when referring to the matrix extension of ML-VAMP.

Contributions. First, similar to the case of ML-VAMP, we analyze ML-Mat-VAMP in a large system limit (LSL), where $n_\ell \to \infty$ and d is fixed, under rotationally invariant random weight matrices $\mathbf{W}_\ell$. In this LSL, we prove that the MSE of the estimates of ML-Mat-VAMP can be exactly predicted by a deterministic set of equations called the *state evolution* (SE). The SE describes how the distribution of the true activations and pre-activations of the network as well as the estimated values generated by ML-Mat-VAMP evolve jointly from one iteration of the algorithm to the other. This extension of the SE equations to the matrix case is not trivial and requires considering correlation across multiple vectors. Indeed, in the case of ML-VAMP, the SE equations involve scalar quantities and 2×2 matrices. For ML-Mat-VAMP, the SE equations involve $d \times d$ and $2d \times 2d$ matrices.

J. Stat. Mech. (2021) 124004

Second, we show that the method can offer precise predictions in important estimation problems that are difficult to analyze via other means. The ML-VAMP was focused on deep reconstruction problems [4, 45]. The matrix version here can be applied to other classes of problems such as multi-task regression, matrix completion and learning the input layer of a NN. Even though these networks are typically shallow (just $L = 2$ layers), there are no existing methods that can provide the same types of precise results. For example, in the case of learning the input layer of a NN, our results can exactly predict the test error as a function of the noise statistics, activations, number of training sample and other key modeling parameters.

Notation. Boldface uppercase letters $\mathbf{X}$ denote matrices. $\mathbf{X}_{n:}$ refers to the nth row of $\mathbf{X}$. Random vectors are row-vectors. For a function $f : \mathbb{R}^{1\times m} \to \mathbb{R}^{1\times k}$, its row-wise extension is represented by $\mathbf{f} : \mathbb{R}^{N\times m} \to \mathbb{R}^{N\times k}$, i.e. $[\mathbf{f}(\mathbf{X})]_{n:} = f(\mathbf{X}_{n:})$. We denote the Jacobian matrix of f by $\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}) \in \mathbb{R}^{m\times k}$, so that $[\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x})]_{ij} = \frac{\partial f_i}{\partial \boldsymbol{x}_j}(\boldsymbol{x})$. For its row-wise extension $\mathbf{f}$, we denote by $\langle \frac{\partial \mathbf{f}}{\partial \mathbf{X}}(\mathbf{X}) \rangle$ the average Jacobian, i.e. $\frac{1}{N}\sum_{n=1}^{N} \frac{\partial f}{\partial \mathbf{X}_{n:}}(\mathbf{X}_{n:}) \in \mathbb{R}^{m\times k}$.

2. Example applications

As we describe next, the matrix estimation problem (2) is of broad interest and several interesting applications can be formulated under this framework. We share a few examples below.

2.1. Multi-task and mixed regression problems

A simple application of the matrix-valued multi-layer inference problem (2) is for *multi-task regression* [31]. Consider a generalized linear model of the form,

$$\mathbf{Y} = \boldsymbol{\phi}(\mathbf{XF}; \boldsymbol{\Xi}), \tag{4}$$

where $\mathbf{Y} \in \mathbb{R}^{N\times d}$ is a matrix of measured responses, $\mathbf{X} \in \mathbb{R}^{N\times p}$ is a known design matrix, $\mathbf{F} \in \mathbb{R}^{p\times d}$ are a set regression coefficients to be estimated, and $\boldsymbol{\Xi}$ is noise. The problem can be considered as d separate regression problems—one for each column. However, in some applications, these design 'tasks' are related in such a way that it benefits to *jointly* estimate the predictors. To do this, it is common to solve an optimization problem of the form

$$\arg\min_{\mathbf{F}} \left\{ \sum_{j=1}^{d}\sum_{i=1}^{N} L(y_{ij}, [\mathbf{XF}]_{ij}) + \lambda \sum_{k=1}^{p} \rho(\mathbf{F}_{k:}) \right\}, \tag{5}$$

where $L(\cdot)$ is a loss function, and $\rho(\cdot)$ is a regularizer that acts on the rows $\mathbf{F}_{k:}$ of $\mathbf{F}$ to couple the prediction coefficients across tasks. For example, the multi-task LASSO [31] uses loss $L(y, z) = (y - z)^2$ and regularization $\rho(\mathbf{F}_{k:}) = \|\mathbf{F}_{k:}\|_2$ to enforce row-sparsity in $\mathbf{F}$. In the compressive-sensing context, multi-task regression is known as the 'multiple measurement vector' (MMV) problem, with applications in MEG reconstruction [8], DoA estimation [42], and parallel MRI [22]. An AMP approach to the MMV problem

J. Stat. Mech. (2021) 124004

was developed in [47]. The multi-task model (4) can be immediately written as a multi-layer network (1) by setting: $\mathbf{Z}_0 := \mathbf{F}, \mathbf{W}_0 := \mathbf{X}, \mathbf{Z}_1 := \mathbf{W}_0\mathbf{Z}_0 = \mathbf{X}\mathbf{F}, \mathbf{Y} = \mathbf{Z}_2 := \boldsymbol{\phi}(\mathbf{Z}_1, \boldsymbol{\Xi})$. Also, by appropriately setting the prior $p(\mathbf{Z}_0)$, the multi-layer matrix MAP inference (3) will match the multi-task optimization (5).

In (5), the regularization couples the columns of $\mathbf{F}$ but the loss term couples its rows. In *mixed regression* problems, the loss couples the columns of $\mathbf{F}$. For example, consider designing predictors $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2]$ for *mixed linear regression* [46], i.e.

$$y_i = q_i \mathbf{x}_i^{\mathsf{T}} \mathbf{f}_1 + (1 - q_i)\mathbf{x}_i^{\mathsf{T}} \mathbf{f}_2 + v_i, \quad q_i \in \{0, 1\}, \tag{6}$$

where $i = 1, \ldots, N$ and the ith response comes from one of two linear models, but which model is not known. This setting can be modeled by a different output mapping: as before, set $\mathbf{Z}_0 := \mathbf{F}$, $\mathbf{Z}_1 = \mathbf{X}\mathbf{F}$ and let the noise in the output layer be $\boldsymbol{\Xi}_1 = [\mathbf{q}, \mathbf{v}]$ which includes the additive noise v_i in (6) and the random selection variable q_i. Then, we can write (6) via an appropriate function, $\mathbf{y} = \boldsymbol{\phi}_1(\mathbf{Z}_1, \boldsymbol{\Xi}_1)$.

2.2. Sketched clustering

A related problem arises in *sketched clustering* [19], where a massive dataset is non-linearly compressed down to a short vector $\mathbf{y} \in \mathbb{R}^n$, from which cluster centroids $\mathbf{f}_k \in \mathbb{R}^p$, for $k = 1, \ldots, d$, are then extracted. This problem can be approached via the optimization [20] $\min_{\boldsymbol{\alpha} \geqslant \mathbf{0}} \min_{\mathbf{F}} \sum_{i=1}^n \left| y_i - \sum_{j=1}^d \alpha_j \, \mathrm{e}^{\sqrt{-1}\mathbf{x}_i^{\mathsf{T}} \mathbf{f}_j} \right|^2$ where $\mathbf{x}_i \in \mathbb{R}^p$ are known i.i.d. Gaussian vectors. An AMP approach to sketched clustering was developed in [5]. For known $\boldsymbol{\alpha}$, the minimization corresponds to MAP estimation with the multi-layer matrix model with $\mathbf{Z}_0 = \mathbf{F}$, $\mathbf{W}_1 = \mathbf{X}$ $\mathbf{Z}_1 = \mathbf{X}\mathbf{F}$ and using the output mapping, $\boldsymbol{\phi}_1(\mathbf{Z}_1, \boldsymbol{\Xi}) := \sum_{j=1}^d \alpha_j \, \mathrm{e}^{\sqrt{-1}\mathbf{Z}_{1:,j}} + \boldsymbol{\Xi}$, where the exponential is applied elementwise and $\boldsymbol{\Xi}$ is i.i.d. Gaussian. The mapping ϕ_1 operates row-wise on $\mathbf{Z}_1$ and $\boldsymbol{\Xi}$.

2.3. Learning the input layer of a two-layer neural network

The matrix inference problem (2) can also be applied to learning the input layer weights in a two-layer NN. Let $\mathbf{X} \in \mathbb{R}^{N \times N_{\text{in}}}$ and $\mathbf{Y} \in \mathbb{R}^{N \times N_{\text{out}}}$ be training data corresponding to N data samples. Consider the two-layer NN model,

$$\mathbf{Y} = \sigma(\mathbf{X}\boldsymbol{F}_1)\boldsymbol{F}_2 + \boldsymbol{\Xi}, \tag{7}$$

with weight matrices $(\boldsymbol{F}_1, \boldsymbol{F}_2)$, componentwise activation function $\sigma(\cdot)$, and noise $\boldsymbol{\Xi}$. In (7), the bias terms are omitted for simplicity. We used the notation '$\boldsymbol{F}_\ell$' for the weights, instead of the standard notation '$\boldsymbol{W}_\ell$', to avoid confusion when (7) is mapped to the multi-layer inference network (2). Now, our critical assumption is that the weights in the second layer, $\boldsymbol{F}_2$, are known. The goal is to learn only the weights of the first layer, $\boldsymbol{F}_1 \in \mathbb{R}^{N_{\text{in}} \times N_{\text{hid}}}$, from a dataset of N samples $(\mathbf{X}, \mathbf{Y})$.

If the activation is ReLU, i.e. $\sigma(\boldsymbol{H}) = \max\{\boldsymbol{H}, 0\}$ and $\mathbf{Y}$ has a single column (i.e. scalar output per sample), and $\boldsymbol{F}_2$ has all positive entries, we can, without loss of generality, treat the weights $\boldsymbol{F}_2$ as fixed, since they can always be absorbed into the

J. Stat. Mech. (2021) 124004

weights $\boldsymbol{F}_1$. In this case, $\mathbf{y}$ and $\mathbf{F}_2$ are vectors and we can write the ith entry of $\mathbf{y}$ as

$$y_i = \sum_{j=1}^{d} F_{2j}\sigma([\mathbf{X}\boldsymbol{F}_1]_{ij}) + \xi_i = \sum_{j=1}^{d} \sigma([\mathbf{X}\boldsymbol{F}_1]_{ij}F_{2j}) + \xi_i. \tag{8}$$

Thus, we can assume, without loss of generality, that $\mathbf{F}_2$ is all ones. The parameterization (8) is sometimes referred to as the *committee machine* [40]. The committee machine has been recently studied by AMP methods [1] and mean-field methods [27] as a way to understand the dynamics of learning.

To pose the two-layer learning problem as multi-layer inference, define $\boldsymbol{Z}_0 := \boldsymbol{F}_1$, $\boldsymbol{W}_1 := \mathbf{X}$, $\boldsymbol{Z}_1 := \mathbf{X}\boldsymbol{F}_1, \boldsymbol{\Xi}_2 := \boldsymbol{\Xi}$, then $\boldsymbol{Y} = \boldsymbol{Z}_2$, where $\boldsymbol{Z}_2$ is the output of a two-layer inference network of the form in (1):

$$\mathbf{Y} = \boldsymbol{Z}_2 = \boldsymbol{\phi}_2(\boldsymbol{Z}_1, \boldsymbol{\Xi}_2) := \sigma(\boldsymbol{Z}_1)\boldsymbol{F}_2 + \boldsymbol{\Xi}_2. \tag{9}$$

Note that $\boldsymbol{W}_1$ is known. Also, since we have assumed that $\boldsymbol{F}_2$ is known, the function $\boldsymbol{\phi}_2$ is known. Finally, the function $\boldsymbol{\phi}_2$ is row-wise separable on both inputs. Thus, the problem of learning the input weights $\boldsymbol{F}_1$ is equivalent to learning the input $\boldsymbol{Z}_0$ of the network (9).

2.4. Model-based matrix completion

Consider an observed matrix $\mathbf{Y} = \mathbf{Z}_L \in \mathbb{R}^{N_L \times d}$ with missing entries $\Omega^c \in [N_L] \times [d]$. The problem is to impute the missing entries of $\mathbf{Y}$. This is an important problem in several applications ranging from recommendation systems, genomics, bioinformatics and more broadly analysis of tabular data. There have been several approaches to solving this data imputation problem, right from 0 imputation and mean imputation to more sophisticated techniques based on generative models.

Consider a generative model based on a multi-layer perceptron as in (1) such that the output $\mathbf{Z}_{L-1}$ models the uncorrupted data matrix. Then the imputation problem can be posed as the solution of the MAP optimization problem:

$$\underset{\{\mathbf{Z}_\ell\}_{\ell=0}^{L}}{\text{minimize}} \|\mathbf{Y} - \mathbf{Z}_{L-1}\|_\Omega^2 - \log \mathbb{P}(\mathbf{Z}_{L-1}, \mathbf{Z}_{L-2}, \ldots, \mathbf{Z}_0) \tag{10}$$

where $\|\mathbf{Y} - \mathbf{Z}_{L-1}\|_\Omega^2 = \sum_{(i,j)\in\Omega}((\mathbf{Y})_{ij} - (\mathbf{Z}_{L-1})_{ij})^2$. One can also similarly construct Bayes estimators such as $\mathbb{E}[\mathbf{Z}_{L-1}|\mathbf{Z}_L]$.

Traditional approaches to matrix completion have looked at regularized convex minimization schemes just like (10) where $-\log \mathbb{P}(\mathbf{Z}_{L-1}) = \|\mathbf{Z}_{L-1}\|_*$, which is the nuclear norm, or some other structure inducing convex norms. While the term $-\log \mathbb{P}(\ldots)$ in (10) can be thought of as a more general regularization term, this formulation allows for more general application problems with heterogeneous variables.

For example, in imputation of tabular data, it is often the case that some columns correspond to continuous valued variables, whereas other variables are discrete valued mod-

J. Stat. Mech. (2021) 124004

eling yes/no answers or count data. In such scenarios the $-\log \mathbb{P}(\mathbf{Z}_{L-1}, \ldots)$ allows more flexibility towards modeling using generalized linear models (GLMs) and other exponential family distributions for every column separately. One simple instance of (10) would be a generative model $-\log \mathbb{P}(\mathbf{Z}_{L-1}, \ldots, \mathbf{Z}_0)$ which is trained on some fully observed data $\mathbf{Z}_{L-1}$ using unsupervised learning methods such as variational autoencoders and generative adversarial networks.

3. Multi-layer matrix VAMP

3.1. MAP and MMSE inference

Observe that the equation (1) define a Markov chain over these signals and thus the posterior $p(\mathbf{Z}|\mathbf{Z}_L)$ factorizes as $p(\mathbf{Z}|\mathbf{Z}_L) \propto p(\mathbf{Z}_0)\prod_{\ell=1}^{L-1} p(\mathbf{Z}_\ell|\mathbf{Z}_{\ell-1})\, p(\mathbf{Y}|\mathbf{Z}_{L-1})$, where recall the notation $\mathbf{Z}$ from (2). The transition probabilities $p(\mathbf{Z}_\ell|\mathbf{Z}_{\ell-1})$ above are implicitly defined in equation (1) and depend on the statistics of noise terms $\mathbf{\Xi}_\ell$. We consider both MAP and minimum mean squared error (MMSE) estimation for this posterior:

$$\widehat{\mathbf{Z}}_{\mathsf{map}} = \arg\max_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{Z}_L) \qquad \widehat{\mathbf{Z}}_{\mathsf{mmse}} = \mathbb{E}[\mathbf{Z}|\mathbf{Z}_L] = \int \mathbf{Z} p(\mathbf{Z}|\mathbf{Z}_L) \mathrm{d}\mathbf{Z}. \tag{11}$$

3.2. Algorithm details

The ML-Mat-VAMP for approximately computing the MAP and MMSE estimates is similar to the ML-VAMP method in [12, 33]. The specific iterations of ML-Mat-VAMP algorithm are shown in algorithm 1. The algorithm produces estimates by a sequence of forward and backward pass updates denoted by superscripts + and − respectively. The estimates $\widehat{\mathbf{Z}}_\ell^\pm$ are constructed by solving sequential problems $\mathbf{Z} = \{\mathbf{Z}_\ell\}_{\ell=0}^{L-1}$ into a sequence of smaller problems each involving estimation of a single activation or preactivation $\mathbf{Z}_\ell$ via *estimation functions* $\{\mathbf{G}_\ell^\pm(\cdot)\}_{\ell=1}^{L-1}$ which are selected depending on whether one is interested in MAP or MMSE estimation.

To describe the estimation functions, we use the notation that, for a positive definite matrix $\mathbf{\Gamma}$, define the inner product $\langle \mathbf{A}, \mathbf{B}\rangle_\Gamma := \mathrm{Tr}(\mathbf{A}^{\mathrm{T}}\mathbf{B}\mathbf{\Gamma})$ and let $\|\mathbf{A}\|_\mathbf{\Gamma}$ denote the norm induced by this inner product. For $\ell = 1, \ldots, L-1$ define the approximate belief functions

$$b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}|\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1}^+, \mathbf{\Gamma}_\ell^-, \mathbf{\Gamma}_{\ell-1}^+) \propto p(\mathbf{Z}_\ell|\mathbf{Z}_{\ell-1}) \mathrm{e}^{-\frac{1}{2}\left\|\mathbf{Z}_\ell - \mathbf{R}_\ell^-\right\|_{\mathbf{\Gamma}_\ell^-}^2 - \frac{1}{2}\left\|\mathbf{Z}_{\ell-1} - \mathbf{R}_{\ell-1}^+\right\|_{\mathbf{\Gamma}_{\ell-1}^+}^2}, \tag{12}$$

where $\mathbf{Z}_\ell, \mathbf{R}_\ell^\pm \in \mathbb{R}^{n_\ell \times d}$ and $\mathbf{\Gamma}_\ell^\pm \in \mathbb{R}^{d\times d}$ for all $\ell = 0, 1, \ldots, L$. Define $b_0(\mathbf{Z}_0|\mathbf{R}_0^-, \mathbf{\Gamma}_0^-)$ and $b_L(\mathbf{Z}_{L-1}|\mathbf{R}_{L-1}^+, \mathbf{\Gamma}_{L-1}^+)$ similarly. The MAP and MMSE estimation functions are then given by the MAP and MMSE estimates for these belief densities,

$$\mathbf{G}_{\ell,\mathsf{map}}^\pm = (\widehat{\mathbf{Z}}_\ell^+, \widehat{\mathbf{Z}}_{\ell-1}^-) = \arg\max b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}) \quad \mathbf{G}_{\ell,\mathsf{mmse}}^\pm = (\widehat{\mathbf{Z}}_\ell^+, \widehat{\mathbf{Z}}_{\ell-1}^-) = \mathbb{E}[(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1})|b_\ell] \tag{13}$$

where the expectation is with respect to the normalized density proportional to b_ℓ. Thus, the ML-Mat-VAMP algorithm reduces the joint estimation of the vectors $(\mathbf{Z}_0, \ldots, \mathbf{Z}_{L-1})$

J. Stat. Mech. (2021) 124004

Algorithm 1. Multilayer Matrix VAMP (ML-Mat-VAMP).

Require: estimators $\mathbf{G}_0^+$, $\mathbf{G}_L^-$, $\{\mathbf{G}_\ell^\pm\}_{\ell=1}^{L-1}$.
Set $\mathbf{R}_{0\ell}^- = \mathbf{0} \in \mathbb{R}^{n_\ell \times d}$ and initialize $\{\mathbf{\Gamma}_{0\ell}^-\}_{\ell=0}^{L-1} \in \mathbb{R}_{\succ 0}^{d\times d}$.
for $k = 0, 1, \ldots, N_{\rm it} - 1$

// `Forward pass`

$\widehat{\mathbf{Z}}_{k0}^+ = \mathbf{G}_0^+(\mathbf{R}_{k0}^-, \mathbf{\Gamma}_{k0}^-)$

$\Lambda_{k0}^+ = \left\langle \frac{\partial \mathbf{G}_0^+}{\partial \mathbf{R}_0^-}(\mathbf{R}_{k0}^-, \mathbf{\Gamma}_{k0}^-) \right\rangle^{-1} \mathbf{\Gamma}_{k,0}^-$,

$\mathbf{\Gamma}_{k,0}^+ = \Lambda_{k,0}^+ - \mathbf{\Gamma}_{k,0}^-$

$\mathbf{R}_{k,0}^+ = (\widehat{\mathbf{Z}}_{k,0}^+ \Lambda_{k,0}^+ - \mathbf{R}_{k,0}^- \mathbf{\Gamma}_{k,0}^-)(\mathbf{\Gamma}_{k,0}^+)^{-1}$

for $\ell = 1, \ldots, L-1$ **do**

$\widehat{\mathbf{Z}}_{k\ell}^+ = \mathbf{G}_\ell^+(\mathbf{R}_{k\ell}^-, \mathbf{R}_{k,\ell-1}^+, \mathbf{\Gamma}_{k\ell}^-, \mathbf{\Gamma}_{k,\ell-1}^+)$

$\Lambda_{k\ell}^+ = \left\langle \frac{\partial \mathbf{G}_\ell^+}{\partial \mathbf{R}_\ell^-}(\ldots) \right\rangle^{-1} \mathbf{\Gamma}_{k\ell}^-$,

$\mathbf{\Gamma}_{k\ell}^+ = \Lambda_{k\ell}^+ - \mathbf{\Gamma}_{k\ell}^-$

$\mathbf{R}_{k\ell}^+ = (\widehat{\mathbf{Z}}_{k\ell}^+ \Lambda_{k\ell}^+ - \mathbf{R}_{k\ell}^- \mathbf{\Gamma}_{k\ell}^-)(\mathbf{\Gamma}_{k\ell}^+)^{-1}$

end for

// `Backward pass`

$\widehat{\mathbf{Z}}_{k,L-1}^- = \mathbf{G}_L^-(\mathbf{R}_{k,L-1}^+, \mathbf{\Gamma}_{k,L-1}^+)$

$\Lambda_{k,L-1}^- = \left\langle \frac{\partial \mathbf{G}_L^-}{\partial \mathbf{R}_{L-1}^+}(\mathbf{R}_{k,L-1}^+, \mathbf{\Gamma}_{k,L-1}^+) \right\rangle^{-1} \mathbf{\Gamma}_{k,L-1}^+$,

$\mathbf{\Gamma}_{k,L-1}^- = \Lambda_{k,L-1}^- - \mathbf{\Gamma}_{k,L-1}^+$

$\mathbf{R}_{k+1,L-1}^- = (\widehat{\mathbf{Z}}_{k,L-1}^- \Lambda_{k,L-1}^- - \mathbf{R}_{k,0}^+ \mathbf{\Gamma}_{k,0}^+)(\mathbf{\Gamma}_{k,0}^-)^{-1}$

for $\ell = L-1, \ldots, 1$ **do**

$\widehat{\mathbf{Z}}_{k+1,\ell-1}^- = \mathbf{G}_\ell^-(\mathbf{R}_{k+1,\ell}^-, \mathbf{R}_{k,\ell-1}^+, \mathbf{\Gamma}_{k+1,\ell}^-, \mathbf{\Gamma}_{k,\ell-1}^+)$

$\Lambda_{k+1,\ell-1}^- = \left\langle \frac{\partial \mathbf{G}_\ell^-}{\partial \mathbf{R}_{\ell-1}^+}(\ldots) \right\rangle^{-1} \mathbf{\Gamma}_{k,\ell-1}^+$,

$\mathbf{\Gamma}_{k+1,\ell}^- = \Lambda_{k\ell}^- - \mathbf{\Gamma}_{k\ell}^+$

$\mathbf{R}_{k+1,\ell-1}^- = (\widehat{\mathbf{Z}}_{k\ell}^- \Lambda_{k\ell}^- - \mathbf{R}_{k\ell}^+ \mathbf{\Gamma}_{k\ell}^+)(\mathbf{\Gamma}_{k+1,\ell}^-)^{-1}$

end for

end for

to a sequence of simpler estimations on sub-problems with terms $(\mathbf{Z}_{\ell-1}, \mathbf{Z}_\ell)$. We refer to these subproblems as denoisers and denote their solutions by $\mathbf{G}_\ell^\pm$, so that $\widehat{\mathbf{Z}}_\ell^+ = \mathbf{G}_\ell^+$ and $\widehat{\mathbf{Z}}_{\ell-1}^- = \mathbf{G}_\ell^-$ corresponding to lines 9 and 20 of algorithm 1. The denoisers $\mathbf{G}_0^+$ and $\mathbf{G}_L^-$, which provide updates to $\widehat{\mathbf{Z}}_0^+$ and $\widehat{\mathbf{Z}}_{L-1}^-$, are defined in a similar manner via b_0 and b_L, respectively.

The estimation functions (13) can be easily computed for the multi-layer matrix network. An important characteristic of these estimators is that they can be computed using maps which are row-wise separable over their inputs and hence are easily parallelizable. To simplify notation, we denote the precision parameters for denoisers $\mathbf{G}_\ell^\pm$ in the kth iteration by

$$\mathbf{\Theta}_{k\ell}^+ := (\mathbf{\Gamma}_{k\ell}^-, \mathbf{\Gamma}_{k,\ell-1}^+), \qquad \mathbf{\Theta}_{k\ell}^- := (\mathbf{\Gamma}_{k+1,\ell}^-, \mathbf{\Gamma}_{k,\ell-1}^+), \qquad \mathbf{\Theta}_{k0}^+ := \mathbf{\Gamma}_{k0}^-, \qquad \mathbf{\Theta}_{kL}^- := \mathbf{\Gamma}_{k,L-1}^+. \quad (14)$$

Non-linear layers. For ℓ even, since the rows of $\mathbf{\Xi}_\ell$ are i.i.d., the belief density $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}|\cdot)$ from (12) factors as a product across rows, $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}) = \prod_n b_\ell([\mathbf{Z}_\ell]_{n:}, [\mathbf{Z}_{\ell-1}]_{n:})$. Thus, the MAP and MMSE estimates (13) can be performed over d-dimensional variables where d is the number of entries in each row. There is no joint estimation across the different n_ℓ rows.

Linear layers. When ℓ is odd, the density $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}|\cdot)$ in (12) is a Gaussian. Hence, the MAP and MMSE estimates agree and can be computed via least squares. Although for linear layers $[\mathbf{G}_\ell^+, \mathbf{G}_\ell^-](\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1}^+, \mathbf{\Theta}_\ell)$ is not row-wise separable over $(\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1})$, it can

be computed using another row-wise denoiser $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-]$ via the singular value decomposition of the weight matrix $\mathbf{W}_\ell = \mathbf{V}_\ell \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}$ as follows. Note that the SVD is only needed to be performed once:

$$\begin{aligned}[\mathbf{G}_\ell^+, \mathbf{G}_\ell^-](\mathbf{R}_\ell, \mathbf{R}_{\ell-1}, \boldsymbol{\Theta}_\ell) &= \underset{\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}}{\arg\max} \|\mathbf{Z}_\ell - \mathbf{W}_\ell \mathbf{Z}_{\ell-1} - \mathbf{B}_\ell\|_{\mathbf{N}_\ell}^2 + \left\|\mathbf{Z}_\ell - \mathbf{R}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 \\ &\quad + \left\|\mathbf{Z}_{\ell-1} - \mathbf{R}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2 \\ &\overset{(a)}{=} \underset{\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}}{\arg\max} \left\|\mathbf{V}_\ell^\mathsf{T}\mathbf{Z}_\ell - \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}\mathbf{Z}_{\ell-1} - \mathbf{V}_\ell^\mathsf{T}\mathbf{B}_\ell\right\|_{\mathbf{N}_\ell}^2 \\ &\quad + \left\|\mathbf{V}_\ell^\mathsf{T}\mathbf{Z}_\ell - \mathbf{V}_\ell^\mathsf{T}\mathbf{R}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 + \left\|\mathbf{V}_{\ell-1}\mathbf{Z}_{\ell-1} - \mathbf{V}_{\ell-1}\mathbf{R}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2 \\ &\overset{(b)}{=} [\mathbf{V}_\ell^\mathsf{T}\widetilde{\mathbf{G}}_\ell^+, \mathbf{V}_{\ell-1}\widetilde{\mathbf{G}}_\ell^-](\mathbf{V}_\ell^\mathsf{T}\mathbf{R}_\ell, \mathbf{V}_{\ell-1}\mathbf{R}_{\ell-1}, \boldsymbol{\Theta}_\ell)\end{aligned}$$

where (a) follows from the rotational invariance of the norm, and (b) follows from the definition of denoiser $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-](\widetilde{\mathbf{R}}_\ell^-, \widetilde{\mathbf{R}}_{\ell-1}^+, \boldsymbol{\Theta}_\ell)$ given below

$$[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-] := \underset{\widetilde{\mathbf{Z}}_\ell, \widetilde{\mathbf{Z}}_{\ell-1}}{\arg\max} \left\|\widetilde{\mathbf{Z}}_\ell - \operatorname{diag}(\mathbf{S}_\ell)\widetilde{\mathbf{Z}}_{\ell-1} - \widetilde{\mathbf{B}}_\ell\right\|_{\mathbf{N}_\ell}^2 \left\|\widetilde{\mathbf{Z}}_\ell - \widetilde{\mathbf{R}}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 + \left\|\widetilde{\mathbf{Z}}_{\ell-1} - \widetilde{\mathbf{R}}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2. \quad (15)$$

Note that the optimization problem in (15), is decomposable accross the rows of variables $\widetilde{\mathbf{Z}}_\ell$ and $\widetilde{\mathbf{Z}}_{\ell-1}$, and hence $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-]$ operates row-wise on its inputs.

Fixed points. We note that the fixed points of the ML-Mat-VAMP algorithm can be shown to be Karush–Kuhn–Tucker points of the variational formulations of (11), omitted here due to lack of space. This is a direct extension of results from section 3 of [34]. In particular, we can show that the ML-Mat-VAMP in the MAP inference case is a preconditioned *Peaceman–Rachford splitting* ADMM type algorithm [39].

4. Analysis in the large system limit

We follow the analysis framework of the ML-VAMP work [12, 33], which is itself based on the original AMP analysis in [3]. This analysis is based on considering the asymptotics of certain large random problem instances. We essentially show that under certain assumptions, as the dimension goes to infinity the behavior of the ML-Mat-VAMP algorithm can be characterized by a set of equations that describe how the distribution of rows of hidden matrices evolve at each iteration of the algorithm for all the layers. Specifically, we consider a sequence of problems (1) indexed by N such that for each problem the dimensions $n_\ell(N)$ are growing so that $\lim_{N\to\infty} \frac{n_\ell}{N} = \beta_\ell \in (0, \infty)$ are scalar constants. Note that d is finite and does not grow with N.

Distributions of weight matrices. For $\ell = 1, 3, \ldots, L-1$, we assume that the weight matrices $\mathbf{W}_\ell$ are generated via the singular value decomposition,

J. Stat. Mech. (2021) 124004

$\mathbf{W}_\ell = \mathbf{V}_\ell \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}$ where $\mathbf{V}_\ell \in \mathbb{R}^{n_\ell \times n_\ell}$ are Haar distributed over orthonormal matrices and $\mathbf{S}_\ell = (s_{\ell,1}, \ldots, s_{\ell,\min\{n_\ell, n_{\ell-1}\}})$. We will describe the distribution of the components $\mathbf{S}_\ell$ momentarily.

Assumption on denoisers. We assume that the non-linear denoisers $\mathbf{G}^{\pm}_{2k}$ act row-wise on their inputs $(\mathbf{R}^-_{2k}, \mathbf{R}^+_{2k-1})$. Further these operators and their Jacobian matrices $\frac{\partial \mathbf{G}^+_{2k}}{\partial \mathbf{R}^-_{2k}}, \frac{\partial \mathbf{G}^-_{2k}}{\partial \mathbf{R}^+_{2k-1}}, \frac{\partial \mathbf{G}^+_{0}}{\partial \mathbf{R}^-_{0}}, \frac{\partial \mathbf{G}^-_{L}}{\partial \mathbf{R}^+_{L-1}}$ are *uniformly Lipschitz continuous*, the definition of which is provided in appendix B.

Assumption on initialization, true variables. The distribution of the remaining variables is described by a weak limit: for a matrix sequence $\boldsymbol{X} := \boldsymbol{X}(N) \in \mathbb{R}^{N\times d}$, by the notation $\boldsymbol{X} \overset{2}{\Rightarrow} X$ we mean that there exists a random variable X in $\mathbb{R}^d$ with $\mathbb{E}\|X\|^2 < \infty$ such that $\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^N \psi(\boldsymbol{X}_{i:}) = \mathbb{E}\,\psi(X)$ almost surely, for any bounded continuous function $\psi : \mathbb{R}^d \to \mathbb{R}$, as well as for quadratic functions $\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}$ for any $\boldsymbol{P} \in \mathbb{R}^{d\times d}_{\succeq 0}$. This is also referred to as Wasserstein-2 convergence [30]. For example, this property is satisfied for a random $\boldsymbol{X}$ with i.i.d. rows with bounded second moments, but is more general, since it applies to deterministic matrix sequences as well. More details on this weak limit are given in appendix B.

Let $\overline{\mathbf{B}}_\ell := \mathbf{V}_\ell^\intercal \mathbf{B}_\ell$, and $\overline{\mathbf{S}}_\ell \in \mathbb{R}^{n_\ell}$ be the zero-padded vector of singular values of $\mathbf{W}_\ell$, and let $\boldsymbol{\tau}^-_{0\ell} \in \mathbb{R}^{d\times d}_{\succ 0}$. Then we assume that the following empirical convergences hold. $(\boldsymbol{\Xi}_\ell, \mathbf{R}^-_{0\ell} - \mathbf{Z}^0_\ell) \overset{2}{\Rightarrow} (\Xi_\ell, Q^-_{0\ell})$ for even ℓ and $(\overline{\mathbf{S}}_\ell, \overline{\mathbf{B}}_\ell, \boldsymbol{\Xi}_\ell, \mathbf{V}_\ell^\top(\mathbf{R}^-_{0\ell} - \mathbf{Z}^0_\ell)) \overset{2}{\Rightarrow} (S_\ell, \overline{B}_\ell, \Xi_\ell, Q^-_{0\ell})$, for odd ℓ. Here $S_\ell \in \mathbb{R}_{\geqslant 0}$ is bounded, $\overline{B}_\ell \in \mathbb{R}^d$ is bounded, $\Xi_{2\ell-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{N}^{-1}_{2\ell-1})$, and $Q^-_{0\ell} \sim \mathcal{N}(\mathbf{0}, \overline{\boldsymbol{\Gamma}}^-_{0\ell})$, for $\ell = 0, 1, \ldots, L-1$ are all pairwise independent random variables. Additionally, we assume that $\mathbf{Z}^0_0 \overset{2}{\Rightarrow} Z^0$ and that the sequence of initial matrices $\{\boldsymbol{\Gamma}^-_{0\ell}\}$ satisfies the following pointwise convergence

$$\boldsymbol{\Gamma}^-_{0\ell}(N) \to \overline{\boldsymbol{\Gamma}}^-_{0\ell}, \quad \ell = 0, 1, \ldots, L-1. \tag{16}$$

4.1. Main result

The main result of this paper concerns the empirical distribution of the rows $[\widehat{\mathbf{Z}}^\pm_\ell]_{n:}, [\mathbf{R}^\pm_\ell]_{n:}$ of the iterates of algorithm 1. It characterizes the asymptotic behavior of these empirical distributions in terms of d-dimensional random vectors, which are either Gaussians or functions of Gaussians. Let $G^\pm_\ell$ denote maps $\mathbb{R}^{1\times d} \to \mathbb{R}^{1\times d}$, such that (13), i.e. $[\mathbf{G}^\pm_\ell(\mathbf{R}^-_\ell, \mathbf{R}^+_{\ell-1}, \boldsymbol{\Theta})]_{n:} = G^\pm_\ell([\mathbf{R}^-_\ell]_{n:}, [\mathbf{R}^+_{\ell-1}]_{n:}, \boldsymbol{\Theta})$. Having stated the requisite definitions and assumptions, we can now state our main result.

Theorem 1. *For a fixed iteration index $k \geqslant 0$, there exist deterministic matrices $\mathbf{K}^+_{k\ell} \in \mathbb{R}^{2\mathrm{d}\times 2d}_{\succ 0}$, and $\boldsymbol{\tau}^-_{k\ell}, \overline{\boldsymbol{\Gamma}}^+_{k\ell}$ and $\overline{\boldsymbol{\Gamma}}^-_{k\ell}, \in \mathbb{R}^{d\times d}_{\succ 0}$ such that for even ℓ:*

$$\left(\mathbf{Z}^0_{\ell-1}, \mathbf{Z}^0_\ell, \widehat{\mathbf{Z}}^-_{k,\ell-1}, \widehat{\mathbf{Z}}^+_{k\ell}\right) \overset{2}{\Rightarrow} \left(\mathsf{A}, \widetilde{\mathsf{A}}, G^-_\ell(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\boldsymbol{\Gamma}}^-_{k\ell}, \overline{\boldsymbol{\Gamma}}^+_{k,\ell-1}), G^+_\ell(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\boldsymbol{\Gamma}}^-_{k\ell}, \overline{\boldsymbol{\Gamma}}^+_{k,\ell-1})\right)$$

J. Stat. Mech. (2021) 124004

where $(\mathsf{A},\mathsf{B}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}^{+}_{k,\ell-1})$, $\mathsf{C} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}^{-}_{k\ell})$, $\widetilde{\mathsf{A}} = \phi_\ell(\mathsf{A}, \Xi_\ell)$ *and* $(\mathsf{A},\mathsf{B}),\mathsf{C}$ *are independent. For* $\ell = 0$, *the same result holds where the* first *and* third *terms are dropped, whereas for* $\ell = L$, *the* second *and* fourth *terms are dropped. Similarly, for odd* ℓ*:*

$$\left(\mathbf{V}^{\mathsf{T}}_{\ell-1}\mathbf{Z}^{0}_{\ell-1}, \mathbf{V}^{\mathsf{T}}_{\ell-1}\mathbf{Z}^{0}_{\ell}, \mathbf{V}_\ell\widehat{\mathbf{Z}}^{-}_{k,\ell-1}, \mathbf{V}_\ell\widehat{\mathbf{Z}}^{+}_{k\ell}\right) \overset{2}{\Rightarrow} \left(\mathsf{A}, \widetilde{\mathsf{A}}, G^{-}_\ell(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\boldsymbol{\Gamma}}^{-}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{+}_{k,\ell-1}), G^{+}_\ell(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\boldsymbol{\Gamma}}^{-}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{+}_{k,\ell-1})\right)$$

where $(\mathsf{A},\mathsf{B}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}^{+}_{k,\ell-1})$, $\mathsf{C} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}^{-}_{k\ell})$, $\widetilde{\mathsf{A}} = S_\ell\,\mathsf{A} + \overline{B}_\ell + \Xi_\ell$ *and* $(\mathsf{A},\mathsf{B}),\mathsf{C}$ *are independent.*

Furthermore for $\ell = 0, 1, \ldots, L-1$, *we have*

$$(\boldsymbol{\Gamma}^{\pm}_{k\ell}, \Lambda^{\pm}_{k\ell}) \xrightarrow{\text{a.s.}} (\overline{\boldsymbol{\Gamma}}^{\pm}_{k\ell}, \overline{\boldsymbol{\Lambda}}^{\pm}_{k\ell}).$$

The parameters in the distribution, $\{\mathbf{K}^{+}_{k\ell}, \boldsymbol{\tau}^{-}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{\pm}_{k\ell}, \overline{\Lambda}^{\pm}_{k\ell}\}$ are deterministic and can be computed via a set of recursive equations called the SE. The SE equations are provided in appendix A. The result is similar to those for ML-VAMP in [12, 34] except that the SE equations for ML-Mat-VAMP involve $d \times d$ and $2d \times 2d$ matrix terms; the ML-VAMP SE only requires scalar and 2×2 matrix terms. The result holds for both MAP inference and MMSE inference, the only difference is implicit, i.e. the choice of denoiser $\mathbf{G}_\ell(\cdot)$ from equation (13).

The importance of theorem 1 is that the rows of the iterates of the ML-Mat-VAMP algorithm ($\widehat{\mathbf{Z}}^{-}_{k,\ell-1}, \widehat{\mathbf{Z}}^{+}_{k\ell}$ in algorithm 1) and the rows of the corresponding true values, $\mathbf{Z}^{0}_{\ell-1}, \mathbf{Z}^{0}_{\ell}$, have a simple, asymptotic random vector description of a typical row. We will call this the 'row-wise' model. According to this model, for even ℓ, the rows of $\mathbf{Z}^{0}_{\ell-1}$ converge to a Gaussian $\mathsf{A} \in \mathbb{R}^d$ and the rows of $\mathbf{Z}^{0}_{\ell}$ converge to the output of the Gaussian through the row-wise function ϕ_ℓ, $\widetilde{\mathsf{A}} = \phi_\ell(\mathsf{A}, \Xi_\ell)$. Then the rows of the estimates $\widehat{\mathbf{Z}}^{-}_{k,\ell-1}, \widehat{\mathbf{Z}}^{+}_{k\ell}$ asymptotically approach the outputs of row-wise estimation function $G^{-}(\cdot)$ and $G^{+}(\cdot)$ supplied by A and $\widetilde{\mathsf{A}}$ corrupted with Gaussian noise. A similar convergence holds for odd ℓ.

This 'row-wise' model enables exact an analysis of the performance of the estimates at each iteration. For example, to compute a weighted MSE metric at iteration k, the convergence shows that,

$$\frac{1}{n_\ell}\left\|\widehat{\mathbf{Z}}^{+}_{k\ell} - \mathbf{Z}^{0}_{\ell}\right\|^2_{\mathbf{H}} \xrightarrow{\text{a.s.}} \mathbb{E}\|\mathbf{G}^{+}_\ell(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \boldsymbol{\Theta}_{k\ell}) - \widetilde{\mathsf{A}}\|^2_{\mathbf{H}},$$

for even ℓ and any positive semi-definite matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$. The norm on the left-hand above acts row-wise, $\|\mathbf{Z}\|^2_{\mathbf{H}} := \sum_i \|\mathbf{Z}_{i:}\|^2_{\mathbf{H}}$. Hence, this asymptotic MSE can be evaluated via expectations of d-dimensional variables from the SE. Similarly, one can obtain exact answers for any other row-wise performance metric of $\{(\widehat{\mathbf{Z}}^{\pm}_{k\ell}, \mathbf{Z}^{0}_{\ell})\}_\ell$ for any k.

J. Stat. Mech. (2021) 124004

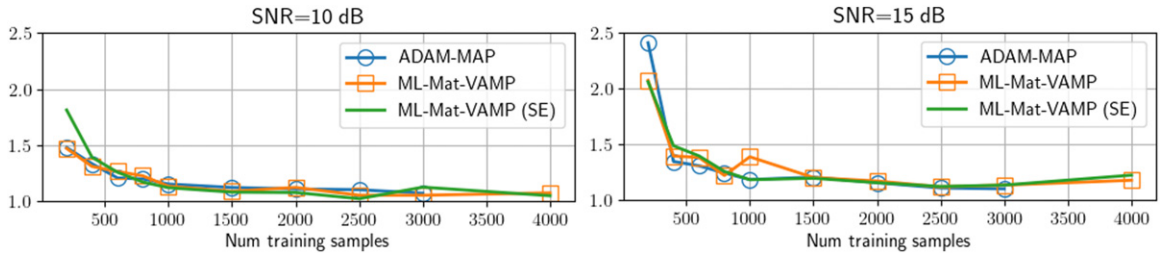

Figure 2. Test error in learning the first layer of a two layer NN using ADAM-based gradient descent, ML-Mat-VAMP and its SE prediction.

5. Numerical experiments

We consider the problem of learning the input layer of a two layer NN as described in section 2.3. We learn the weights $\mathbf{F}_1$ of the first layer of a two-layer network by solving problem (9). The LSL analysis in this case corresponds to the input size n_{in} and number of samples N going to infinity with the number of hidden units being fixed. Our experiment take $d = 4$ hidden units, $N_{\text{in}} = 100$ input units, $N_{\text{out}} = 1$ output unit, sigmoid activations and variable number of samples N. The weight vectors $\mathbf{F}_1$ and $\mathbf{F}_2$ are generated as i.i.d. Gaussians with zero mean and unit variance. The input $\mathbf{X}$ is also i.i.d. Gaussians with variance $1/N_{\text{in}}$ so that the average pre-activation has unit variance. Output noise is added at two levels of 10 and 15 dB relative to the mean. We generate 1000 test samples and a variable number of training samples that ranges from 200 to 4000. For each trial and number of training samples, we compare three methods: (i) MAP estimation where the MAP loss function is minimized by the ADAM optimizer [21] in the Keras package of Tensorflow; (ii) algorithm 1 run for 20 iterations and (iii) the SE prediction. The ADAM algorithm is run for 100 epochs with a learning rate = 0.01. The expectations in the SE are estimated via Monte-Carlo sampling (hence there is some variation).

Given an estimate $\widehat{\mathbf{F}}_1$ and true value $\mathbf{F}_1^0$, we can compute the test error as follows: given a new sample $\mathbf{x}$, the true and predicted pre-activations will be $\mathbf{z}_1 = (\mathbf{F}_1^0)^{\mathsf{T}}\mathbf{x}$ and $\widehat{\mathbf{z}}_1 = \widehat{\mathbf{F}}_1^{\mathsf{T}}\mathbf{x}$. Thus, if the new sample $\mathbf{x} \sim \mathcal{N}(0, \frac{1}{N_{\text{in}}}\mathbf{I})$, the true and predicted pre-activations, $(\mathbf{z}_1, \widehat{\mathbf{z}}_1)$, will be jointly Gaussian with covariance equal to the empirical $2d \times 2d$ covariance matrix of the rows of $\mathbf{F}_1^0$ and $\widehat{\mathbf{F}}_1$:

$$\mathbf{K} := \frac{1}{N_{\text{in}}} \sum_{k=1}^{N_{\text{in}}} \mathbf{u}_k^{\mathsf{T}} \mathbf{u}_k, \quad \mathbf{u}_k = \left[\mathbf{F}_{1,k:} \; \widehat{\mathbf{F}}_{1,k:} \right]. \tag{17}$$

From this covariance matrix, we can estimate the test error, $\mathbb{E}|y - \widehat{y}|^2 = \mathbb{E}|\mathbf{F}_2^{\mathsf{T}} (\sigma(\mathbf{z}_1) - \sigma(\widehat{\mathbf{z}}_1)|^2$, where the expectation is taken over the Gaussian $(\mathbf{z}_1, \widehat{\mathbf{z}}_1)$ with covariance $\mathbf{K}$. Also, since (17) is a row-wise operation, it can be predicted from the ML-Mat-

J. Stat. Mech. (2021) 124004

VAMP SE. Thus, the SE can also predict the asymptotic test error. The normalized test error for ADAM-MAP, ML-Mat-VAMP and the ML-Mat-VAMP SE are plotted in figure 2. The normalized test error is defined as the ratio of the MSE on the test samples to the optimal MSE. Hence, a normalized MSE of one is the minimum value.

Note that since ADAM and ML-Mat-VAMP are solving the same optimization problem, they perform similarly as expected. The main message of this paper is not to develop an algorithm that outperforms ADAM, but rather an algorithm that has theoretical guarantees. The key property of ML-Mat-VAMP is that its asymptotic behavior at all the iterations can be exactly predicted by the SE equations. In this example, figure 2 shows that the normalized test MSE predicted via SE (plotted in green) matches the normalized MSE of ML-Mat-VAMP estimates (plotted in orange).

6. Conclusions

We have developed a general framework for analyzing inference in multi-layer networks with matrix valued quantities in certain high-dimensional random settings. For learning the input layer of a two layer network, the methods enables precise predictions of the expected test error as a function of the parameter statistics, numbers of samples and noise level. This analysis can be valuable in understanding key properties such as generalization error, for example using ML-VAMP, Emami *et al* [11] characterizes the generalization error of GLMs under a variety of feature distributions and train-test mismatch. Future work will look to extend these to more complex networks.

Broader impact

In statistical physics, systems with a large number of degrees of freedom often admit a simplified macroscopic description. Modern NNs have thousands of hidden units and billions of free parameters; is there an analogous macroscopic description of the dynamics of multi-layer NNs? This paper identifies some of these macroscopic descriptions that can be used to analyze a large class of optimization problems (see section 2 for examples) arising in signal processing, data science, and machine learning. The techniques developed in this paper allow analyzing and understanding the fundamental limits of learning in 1 and 2 layer NNs which are basic building blocks in modern machine learning pipelines.

Acknowledgments

The work of P Schniter was supported by NSF Grant 1716388. The work of P Pandit, M Saharee-Ardakan and A K Fletcher was supported in part by the NSF Grants 1738285 and 1738286, ONR Grant N00014-15-1-2677. The work of S Rangan was supported in

J. Stat. Mech. (2021) 124004

part by NSF Grants 1116589, 1302336, and 1547332, NIST, SRC and the industrial affiliates of NYU Wireless.

Appendix A. State evolution equations

The SE equations given in algorithm 2 define an iteration indexed by k of constant matrices $\{\mathbf{K}^+_{k\ell}, \boldsymbol{\tau}^-_{kl}, \overline{\boldsymbol{\Gamma}}^{\pm}_{kl}\}_{\ell=0}^{L}$. These constants appear in the statement of the main result in theorem 1. The iterations in algorithm 2 also iteratively define a few $\mathbb{R}^{1\times d}$ valued random vectors $\{Q^0_\ell, P^0_\ell, Q^\pm_{k\ell}, P^\pm_{k\ell}\}$ which are either multivariate Gaussian or functions of multivariate Gaussians. In order to state algorithm 2, we need to define certain random variables and functions appearing therein which are described below. Let $\mathcal{L}_{\text{odd}} = \{1, 3, \ldots, L-1\}$ and $\mathcal{L}_{\text{even}} = \{2, 4, \ldots, L-2\}$.

Define $\{\overline{\boldsymbol{\Theta}}^{\pm}_{k\ell}\}$ similar to $\boldsymbol{\Theta}^{\pm}_{k\ell}$ from equation (14) using $\{\overline{\boldsymbol{\Gamma}}^{\pm}_{k\ell}\}$. Further, for $\ell = 1, 2, \ldots, L-1$ define

$$\overline{\boldsymbol{\Omega}}^{+}_{k\ell} := (\overline{\boldsymbol{\Lambda}}^{+}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{+}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{-}_{k\ell}), \qquad \overline{\boldsymbol{\Omega}}^{-}_{k\ell} := (\overline{\boldsymbol{\Lambda}}^{+}_{k,\ell-1}, \overline{\boldsymbol{\Gamma}}^{-}_{k,\ell-1}, \overline{\boldsymbol{\Gamma}}^{-}_{k,\ell-1}),$$

and $\overline{\boldsymbol{\Omega}}^{+}_{k0}$ and $\overline{\boldsymbol{\Omega}}^{-}_{kL}$. Now define random variables W_ℓ as

$$\begin{aligned} &W_0 = Z^0_0, \qquad W_L = (Y, \Xi_L), \qquad W_\ell = \Xi_\ell, \qquad \forall\, \ell \in \mathcal{L}_{\text{even}}, \\ &W_\ell = (S_\ell, \overline{B}_\ell, \Xi_\ell), \qquad\qquad\qquad\qquad\quad \forall\, \ell \in \mathcal{L}_{\text{odd}}. \end{aligned} \tag{18}$$

Define functions $\{f^0_\ell\}_{\ell=1}^{L}$ as

$$\begin{aligned} &f^0_\ell(P^0_{\ell-1}, W_\ell) := S_\ell P^0_{\ell-1} + \overline{B}_\ell + \Xi_\ell, \quad \forall\, \ell \in \mathcal{L}_{\text{odd}}, \\ &f^0_\ell(P^0_{\ell-1}, W_\ell) := \phi_\ell(P^0_{\ell-1}, \Xi_\ell), \quad \forall\, \ell \in \mathcal{L}_{\text{even}} \cup \{L\} \end{aligned} \tag{19}$$

and using (14) define functions $\{h^\pm_\ell, \}_{\ell=1}^{L}$, h^+_0 and h^-_L as

$$\begin{aligned} &h^\pm_\ell(P^0_{\ell-1}, P^+_{\ell-1}, Q^-_\ell, W_\ell, \boldsymbol{\Theta}^\pm_{k\ell}) = G^\pm_\ell(Q^-_\ell + Q^0_\ell, P^+_{\ell-1} + P^0_{\ell-1}, \boldsymbol{\Theta}^\pm_{k\ell}), \quad \forall\, \ell \in \mathcal{L}_{\text{even}}, \\ &h^\pm_\ell(P^0_{\ell-1}, P^+_{\ell-1}, Q^-_\ell, W_\ell, \boldsymbol{\Theta}^\pm_{k\ell}) = \widetilde{G}^\pm_\ell(Q^-_\ell + Q^0_\ell, P^+_{\ell-1} + P^0_{\ell-1}, \boldsymbol{\Theta}^\pm_{k\ell}), \quad \forall\, \ell \in \mathcal{L}_{\text{odd}} \\ &h^+_0(Q^-_0, W_0, \boldsymbol{\Theta}^+_{k0}) = G^+_0(Q^-_0 + W_0, \boldsymbol{\Theta}^+_{k0}), \\ &h^-_L(P^0_{L-1}, P^+_{L-1}, W_L, \boldsymbol{\Theta}^-_{kL}) = G^-_L(P^+_{L-1} + P^0_{L-1}, \boldsymbol{\Theta}^-_{kL}). \end{aligned} \tag{20}$$

Note that $[G^+_\ell, G^-_\ell]$ and $[\widetilde{G}^+_\ell, \widetilde{G}^-_\ell]$ are maps from $\mathbb{R}^{1\times d} \to \mathbb{R}^{1\times d}$ such that their row-wise extensions are the denoisers $[\mathbf{G}^+_\ell, \mathbf{G}^-_\ell]$ and $[\widetilde{\mathbf{G}}^+_\ell, \widetilde{\mathbf{G}}^-_\ell]$ respectively. Using (20) define functions $\{f^\pm_\ell\}_{\ell=1}^{L-1}$, f^+_0 and f^-_L as

$$\begin{aligned} &f^+_\ell(P^0_{\ell-1}, P^+_{\ell-1}, Q^-_\ell, W_\ell, \boldsymbol{\Omega}^+_{k\ell}) = \left[\left(h^+_\ell - Q^0_\ell\right)\Lambda^+_{k\ell} - Q^-_\ell \boldsymbol{\Gamma}^-_{k\ell}\right](\boldsymbol{\Gamma}^+_{k\ell})^{-1}, \\ &f^-_\ell(P^0_{\ell-1}, P^+_{\ell-1}, Q^-_\ell, W_\ell, \boldsymbol{\Omega}^-_{k\ell}) = \left[\left(h^-_\ell - P^0_{\ell-1}\right)\Lambda^-_{k,\ell-1} - P^+_{\ell-1}\boldsymbol{\Gamma}^+_{k,\ell-1}\right](\boldsymbol{\Gamma}^-_{k,\ell-1})^{-1}. \\ &f^+_0(Q^-_0, W_0, \boldsymbol{\Omega}^+_{k0}) = \left[(h^+_0 - W_0)\Lambda^+_{k0} - Q^-_0\boldsymbol{\Gamma}^-_{k0}\right](\boldsymbol{\Gamma}^+_{k0})^{-1}, \\ &f^-_L(P^0_{L-1}, P^+_{L-1}, W_L, \boldsymbol{\Omega}^-_{kL}) = \left[\left(h^-_L - P^0_{L-1}\right)\Lambda^-_{k,L-1} - P^+_{L-1}\boldsymbol{\Gamma}^+_{k,L-1}\right](\boldsymbol{\Gamma}^-_{k,L-1})^{-1}. \end{aligned} \tag{21}$$

Algorithm 2. SE for ML-Mat-VAMP (algorithm 1).

Require: functions $\{f_\ell^0\}$ from (19), $\{h_\ell^\pm\}$ from (20), and $\{f_\ell^\pm\}$ from (21). Perturbation random variables $\{W_\ell\}$ from (18). Initial random vectors $\{Q_{0\ell}^-\}_{\ell=0}^{L-1}$ with initial covariance matrices $\{\boldsymbol{\tau}_{0\ell}^-\}_{\ell=0}^{L-1}$ from section 4. Initial matrices $\{\overline{\boldsymbol{\Gamma}}_{0\ell}^-\}_{\ell=0}^{L}$ from (16).

// `Initial pass`
$Q_0^0 = W_0$, $\boldsymbol{\tau}_0^0 = \mathrm{Cov}(Q_0^0)$ and $P_0^0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_0^0)$
for $\ell = 1, \ldots, L-1$ **do**
  $Q_\ell^0 = f_\ell^0(P_{\ell-1}^0, W_\ell)$
  $P_\ell^0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_\ell^0)$, $\boldsymbol{\tau}_\ell^0 = \mathrm{Cov}(Q_\ell^0)$
end for

for $k = 0, 1, \ldots,$ **do**
  // `Forward pass`
  $\widehat{Q}_{k0}^+ = h_0^+(Q_{k0}^-, W_0, \overline{\boldsymbol{\Theta}}_{k0}^+)$
  $\overline{\boldsymbol{\Lambda}}_{k0}^+ = (\mathbb{E}\frac{\partial \widehat{Q}_{k0}^+}{\partial Q_0^-})^{-1}\overline{\boldsymbol{\Gamma}}_{k,0}^-$
  $\overline{\boldsymbol{\Gamma}}_{k0}^+ = \overline{\boldsymbol{\Lambda}}_{k0}^+ - \overline{\boldsymbol{\Gamma}}_{k0}^-$
  $Q_{k0}^+ = f_0^+(Q_{k0}^-, W_0, \overline{\boldsymbol{\Omega}}_{k0}^+)$
  $(P_0^0, P_{k0}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k0}^+)$, $\mathbf{K}_{k0}^+ := \mathrm{Cov}(Q_0^0, Q_{k0}^+)$
  for $\ell = 1, \ldots, L-1$ **do**
    $\widehat{Q}_{k\ell}^+ = h_\ell^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\boldsymbol{\Theta}}_{k\ell}^+)$
    $\overline{\boldsymbol{\Lambda}}_{k\ell}^+ = (\mathbb{E}\frac{\partial \widehat{Q}_{k\ell}^+}{\partial Q_{k\ell}^-})^{-1}\overline{\boldsymbol{\Gamma}}_{k\ell}^-$
    $\overline{\boldsymbol{\Gamma}}_{k\ell}^+ = \overline{\boldsymbol{\Lambda}}_{k\ell}^+ - \overline{\boldsymbol{\Gamma}}_{k\ell}^-$
    $Q_{k\ell}^+ = f_\ell^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\boldsymbol{\Omega}}_{k\ell}^+)$
    $(P_\ell^0, P_{k\ell}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k\ell}^+)$, $\mathbf{K}_{k\ell}^+ := \mathrm{Cov}(Q_\ell^0, Q_{k\ell}^+)$
  end for
  // `Backward pass`
  $\widehat{P}_{k+1,L-1}^- = h_L^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\boldsymbol{\Theta}}_{k+1,L}^-)$
  $\overline{\boldsymbol{\Lambda}}_{k+1,L}^- = (\mathbb{E}\frac{\partial \widehat{P}_{k+1,L-1}^-}{\partial P_{L-1}^+})^{-1}\overline{\boldsymbol{\Gamma}}_{kL}^+$
  $\overline{\boldsymbol{\Gamma}}_{k+1,L-1}^- = \overline{\boldsymbol{\Lambda}}_{k+1,L-1}^- - \overline{\boldsymbol{\Gamma}}_{k,L-1}^+$,
  $P_{k+1,L-1}^- = f_L^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\boldsymbol{\Omega}}_{k+1,L}^-)$
  $Q_{k+1,L-1}^- \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k+1,L-1}^-)$, $\boldsymbol{\tau}_{k+1,L-1}^- := \mathrm{Cov}(P_{k+1,L-1}^-)$
  for $\ell = L-2, \ldots, 0$ **do**
    $\widehat{P}_{k+1,\ell}^- = h_\ell^-(P_\ell^0, P_{k\ell}^+, Q_{k+1,\ell+1}^-, W_\ell, \overline{\boldsymbol{\Theta}}_{k+1,\ell}^-)$
    $\overline{\boldsymbol{\Lambda}}_{k+1,\ell}^- = (\mathbb{E}\frac{\partial \widehat{P}_{k+1,\ell}^-}{\partial P_{k,\ell}^+})^{-1}\overline{\boldsymbol{\Gamma}}_{k,\ell}^+$
    $\overline{\boldsymbol{\Gamma}}_{k+1,\ell}^- = \overline{\boldsymbol{\Lambda}}_{k+1,\ell}^- - \overline{\boldsymbol{\Gamma}}_{k,\ell}^+$,
    $P_{k+1,\ell}^- = f_\ell^-(P_\ell^0, P_{k\ell}^+, Q_{k+1,\ell+1}^-, W_\ell, \overline{\boldsymbol{\Omega}}_{k+1,\ell}^-)$
    $Q_{k+1,\ell}^- \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k+1,\ell}^-)$, $\boldsymbol{\tau}_{k+1,\ell}^- := \mathrm{Cov}(P_{k+1,\ell}^-)$
  end for
end for

Appendix B. Large system limit details

The analysis of algorithm 1 in the LSL is based on [3] and is by now standard in the theory of AMP-based algorithms. The goal is to characterize ensemble row-wise averages of iterates of the algorithm using *simpler* finite-dimensional random variables which are either Gaussians or functions of Gaussians. To that end, we start by defining some key terms needed in this analysis.

Definition 1 (pseudo-Lipschitz continuity). For a given $p \geqslant 1$, a map $\mathbf{g}: \mathbb{R}^{1\times d} \to \mathbb{R}^{1\times r}$ is called pseudo-Lipschitz of order p if for any $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^d$ we have,

$$\|\mathbf{g}(\mathbf{r}_1) - \mathbf{g}(\mathbf{r}_2)\| \leqslant C\|\mathbf{r}_1 - \mathbf{r}_2\| \left(1 + \|\mathbf{r}_1\|^{p-1} + \|\mathbf{r}_2\|^{p-1}\right).$$

Definition 2 (empirical convergence of rows of a matrix sequence). Consider a matrix-sequence $\{\mathbf{X}^{(N)}\}_{N=1}^{\infty}$ with $\mathbf{X}^{(N)} \in \mathbb{R}^{N\times d}$. For a finite $p \geqslant 1$, let $X \in (\mathbb{R}^d, \mathcal{R}^d)$ be a $\mathcal{R}^d$-measurable random variable with bounded moment $\mathbb{E}\|X\|_p^p < \infty$. We say the rows of matrix sequence $\{\mathbf{X}^{(N)}\}$ *converge empirically to X with pth order moments* if for all pseudo-Lipschitz continuous functions $f(\cdot)$ of order p,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{X}_{n:}^{(N)}) = \mathbb{E}[f(X)] \quad \text{a.s.} \tag{22}$$

Note that the sequence $\{\mathbf{X}^{(N)}\}$ could be random or deterministic. If it is random, however, then the quantities on the left-hand side are random sums and the almost sure convergence must take this randomness into account as well.

The above convergence is equivalent to requiring weak convergence as well as convergence of the pth moment, of the empirical distribution $\frac{1}{N}\sum_{n=1}^{N} \delta_{\mathbf{X}_{n:}^{(N)}}$ of the rows of $\mathbf{X}^{(N)}$ to X. This is also referred to convergence in the Wasserstein-p metric (chapter 6 in [43]).

In the case of $p = 2$, the condition is equivalent to requiring (22) to hold for all continuously bounded functions f as well as for all $f_q(\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}$ for all positive definite matrices $\boldsymbol{Q}$.

Definition 3 (uniform Lipschitz continuity). For a positive definite matrix $\boldsymbol{M}$, the map $\phi(\mathbf{r}; \boldsymbol{M}): \mathbb{R}^d \to \mathbb{R}^d$ is said to be uniformly Lipschitz continuous in $\mathbf{r}$ at $\boldsymbol{M} = \overline{\boldsymbol{M}}$ if there exist non-negative constants L_1, L_2 and L_3 such that for all $\mathbf{r} \in \mathbb{R}^d$

$$\|\phi(\mathbf{r}_1; \boldsymbol{M}_0) - \phi(\mathbf{r}_2; \boldsymbol{M}_0)\| \leqslant L_1\|\mathbf{r}_1 - \mathbf{r}_2\|$$
$$\|\phi(\mathbf{r}; \boldsymbol{M}_1) - \phi(\mathbf{r}; \boldsymbol{M}_2)\| \leqslant L_2(1 + \|\mathbf{r}\|)\rho(\boldsymbol{M}_1, \boldsymbol{M}_2)$$

for all $\boldsymbol{M}_i$ such that $\rho(\boldsymbol{M}_i, \overline{\boldsymbol{M}}) < L_3$ where ρ is a metric on the cone of positive semidefinite matrices.

We are now ready to prove theorem 1.

Appendix C. Proof of theorem 1

The proof of theorem 1 is a special case of a more general result on multi-layer recursions given in theorem 2. This result is stated in appendix D, and proved in appendix E. The

J. Stat. Mech. (2021) 124004

rest of this section identifies certain relevant quantities from theorem 1 in order to apply theorem 2.

Consider the SVD given of weight matrices $\mathbf{W}_\ell$ of the network given by,

$$\mathbf{W}_\ell = \mathbf{V}_\ell \, \mathrm{diag}(\mathbf{S}_\ell)\mathbf{V}_\ell - 1$$

as explained in section 4 of the main paper. We analyze algorithm 1 using *transformed* versions of the true signals $\mathbf{Z}_\ell^0$ and input errors $\mathbf{R}_\ell^\pm - \mathbf{Z}_\ell^0$ to the denoisers $\mathbf{G}_\ell^\pm$. For $\ell = 0, 2, \ldots, L-2$, define

$$\mathbf{q}_\ell^0 = \mathbf{Z}_\ell^0 \qquad \mathbf{q}_{\ell+1}^0 = \mathbf{V}_{\ell+1}^\top \mathbf{Z}_{\ell+1}^0 \tag{23a}$$

$$\mathbf{p}_\ell^0 = \mathbf{V}_\ell \mathbf{Z}_\ell^0 \qquad \mathbf{p}_{\ell+1}^0 = \mathbf{Z}_{\ell+1}^0 \tag{23b}$$

which are depicted in figure 3 (top). Similarly, define the following *transformed* versions of errors in the inputs $\mathbf{R}_\ell^\pm$ to the denoisers $\mathbf{G}_\ell^\pm$

$$\mathbf{q}_\ell^- = \mathbf{R}_\ell^- - \mathbf{Z}_\ell^0 \qquad \mathbf{q}_{\ell+1}^- = \mathbf{V}_{\ell+1}^\top (\mathbf{R}_{\ell+1}^- - \mathbf{Z}_{\ell+1}^0) \tag{24a}$$

$$\mathbf{p}_\ell^+ = \mathbf{V}_\ell (\mathbf{R}_\ell^+ - \mathbf{Z}_\ell^0) \qquad \mathbf{p}_{\ell+1}^+ = \mathbf{R}_{\ell+1}^+ - \mathbf{Z}_{\ell+1}^0. \tag{24b}$$

These quantities are depicted as inputs to function blocks $\mathbf{f}_\ell^\pm$ in figure 3 (middle). Define perturbation variables $\mathbf{w}_\ell$ as

$$\mathbf{w}_0 = \mathbf{Z}_0^0, \qquad \mathbf{w}_L = (\mathbf{Y}, \boldsymbol{\Xi}_L), \qquad \mathbf{w}_\ell = \boldsymbol{\Xi}_\ell, \quad \forall\, \ell \in \mathcal{L}_{\text{even}} \tag{25a}$$

$$\mathbf{w}_\ell = (\mathbf{S}_\ell, \overline{\mathbf{B}}_\ell, \boldsymbol{\Xi}_\ell), \quad \forall\, \ell \in \mathcal{L}_{\text{odd}}. \tag{25b}$$

Finally, we define $\mathbf{q}_\ell^+$ and $\mathbf{p}_\ell^-$ for $\ell = 1, 2, \ldots, L-1$ as

$$\mathbf{q}_\ell^+ = \mathbf{f}_\ell^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{\ell-1}^+, \mathbf{q}_\ell^-, \mathbf{w}_\ell, \Omega_\ell) \tag{26a}$$

$$\mathbf{p}_{\ell-1}^- = \mathbf{f}_\ell^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{\ell-1}^+, \mathbf{q}_\ell^-, \mathbf{w}_\ell, \Omega_\ell), \tag{26b}$$

which are outputs of function blocks in figure 3 (middle). Similarly, define the quantities $\mathbf{q}_0^+ = \mathbf{f}_0^+(\mathbf{q}_0^-, \mathbf{Z}_0, \Omega_0)$ and $\mathbf{p}_{L-1}^- = \mathbf{f}_L^+(\mathbf{p}_{L-1}^0, \mathbf{p}_{L-1}^+, \mathbf{Y}, \Omega_L)$.

Lemma 1. *Algorithm 1 is a special case of algorithm 3 with the definitions* $\{\mathbf{q}_\ell^0, \mathbf{p}_\ell^0, \mathbf{q}_\ell^\pm, \mathbf{p}_\ell^\pm\}_{\ell=0}^{L-1}$ *given in equations (23), (24), and (26), functions* $\mathbf{f}_\ell^\pm$ *are row-wise extensions of* $f_\ell^\pm$ *defined using equations (20) and (21).*

Lemma 2. *Assumptions 1 and 2 required for applying theorem 2 are satisfied by the conditions in theorem 1.*

Proof. The proofs of the above lemmas are identical to the case of $d = 1$, which was shown in [34]. For details see appendix F in [34]. □

Appendix D. General multi-layer recursions

To analyze algorithm 1, we consider a more general class of recursions as given in algorithm 3 and depicted in figure 3. The Gen-ML recursions generates (i) a set of *true matrices* $\mathbf{q}_\ell^0$ and $\mathbf{p}_\ell^0$ and (ii) *iterated matrices* $\mathbf{q}_{k\ell}^\pm$ and $\mathbf{p}_{k\ell}^\pm$. Each of these matrices have the same number of columns, denoted by d.

J. Stat. Mech. (2021) 124004

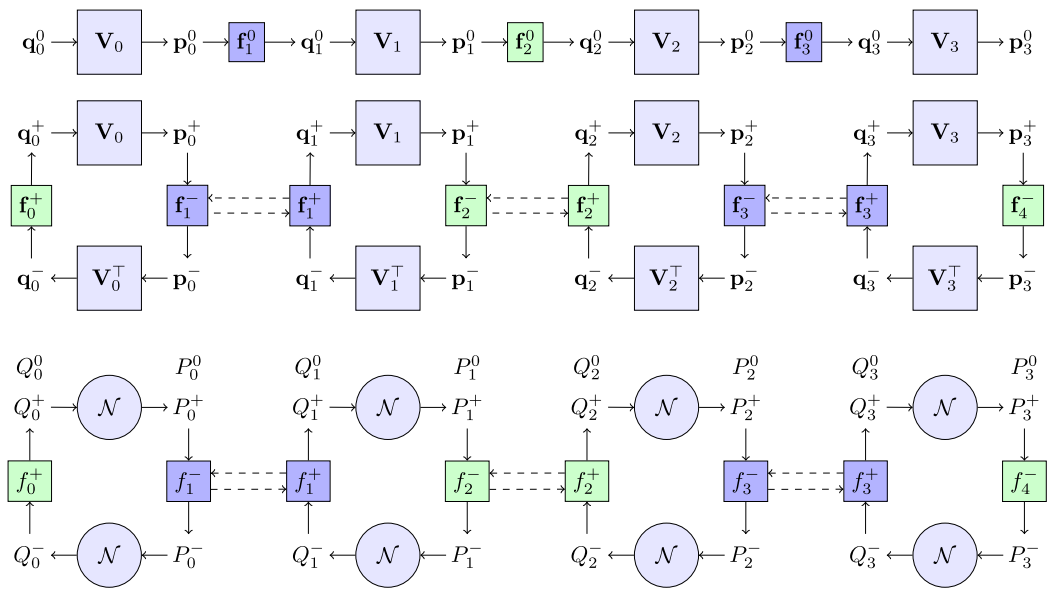

Figure 3. (Top) The equation (1) with equivalent quantities defined in (23), and $\mathbf{f}_\ell^0$ defined using (19). (Middle) The Gen-ML-Mat recursions in algorithm 3. These are also equivalent to ML-Mat-VAMP recursions from algorithm 1 (see lemma 1) if $\mathbf{q}^\pm, \mathbf{p}^\pm$ are as defined as in equations (24) and (26), and $\mathbf{f}_\ell^\pm$ given by equations (20) and (21). (Bottom) Quantities in the GEN-ML-SE recursions. These are also equivalent to ML-Mat-VAMP SE recursions from algorithm 2 (see lemma 1). The iteration indices k have been dropped for notational simplicity.

The true matrices are generated by a single forward pass, whereas the iterated matrices are generated via a sequence of forward and backward passes through a multi-layer system. In proving the SE for the ML-Mat-VAMP algorithm (algorithm 1, one would then associate the terms $\mathbf{q}_{k\ell}^\pm$ and $\mathbf{p}_{k\ell}^\pm$ with certain error quantities in the ML-Mat-VAMP recursions. To account for the effect of the parameters $\mathbf{\Gamma}_{k\ell}^\pm$ and $\Lambda_{k\ell}^\pm$ in ML-Mat-VAMP, the Gen-ML algorithm describes the parameter updates through a sequence of *parameter lists* $\Upsilon_{k\ell}^\pm$. The parameter lists are ordered lists of parameters that accumulate as the algorithm progresses. The true and iterated matrices from algorithm 3 are depicted in the signal flow graphs in figure 3 (top) and (middle), respectively. The iteration index k for the iterated vectors $\mathbf{q}_{k\ell}, \mathbf{p}_{k\ell}$ has been dropped for simplifying notation.

The functions $\mathbf{f}_\ell^0(\cdot)$ that produce the true matrices $\mathbf{q}_\ell^0, \mathbf{p}_\ell^0$ are called *initial matrix functions* and use the initial parameter list Υ_{01}^-. The functions $\mathbf{f}_{k\ell}^\pm(\cdot)$ that produce the matrices $\mathbf{q}_{k\ell}^+$ and $\mathbf{p}_{k\ell}^-$ are called the *matrix update functions* and use parameter lists $\Upsilon_{kl}^\pm$. The initial parameter lists Υ_{01}^- are assumed to be provided. As the algorithm progresses, new parameters $\lambda_{k\ell}^\pm$ are computed and then added to the lists in lines 12, 18, 25 and 31. The matrix update functions $\mathbf{f}_{k\ell}^\pm(\cdot)$ may depend on any sets of parameters accumulated in the parameter list. In lines 11, 17, 24 and 30, the new parameters $\lambda_{k\ell}^\pm$ are computed by: (1) computing average values $\mu_{k\ell}^\pm$ of *row-wise* functions $\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)$; and (2) taking functions $T_{k\ell}^\pm(\cdot)$ of the average values $\mu_{k\ell}^\pm$. Since the average values $\mu_{k\ell}^\pm$ represent statistics on the

J. Stat. Mech. (2021) 124004

Algorithm 3. General multi-layer matrix (Gen-ML-Mat) recursion.

Require: initial matrix functions $\{\mathbf{f}_\ell^0\}$. Matrix update functions $\{\mathbf{f}_{k\ell}^\pm(\cdot)\}$. Parameter statistic functions $\{\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)\}$. Parameter update functions $\{T_{k\ell}^\pm(\cdot)\}$. Orthogonal matrices $\{\mathbf{V}_\ell\}$. Perturbation variables $\{\mathbf{w}_\ell^\pm\}$. Initial matrices $\{\mathbf{q}_{0\ell}^-\}$. Initial parameter list Υ_{01}^-.

// Initial pass
$\mathbf{q}_0^0 = \mathbf{f}_0^0(\mathbf{w}_0),\ \mathbf{p}_0^0 = \mathbf{V}_0\mathbf{q}_0^0$
for $\ell = 1, \dots, L-1$ **do**
  $\mathbf{q}_\ell^0 = \mathbf{f}_\ell^0(\mathbf{p}_{\ell-1}^0, \mathbf{w}_\ell, \Upsilon_{01}^-)$
  $\mathbf{p}_\ell^0 = \mathbf{V}_\ell\mathbf{q}_\ell^0$
end for

for $k = 0, 1, \dots$ **do**
  // Forward pass
  $\lambda_{k0}^+ = T_{k0}^+(\mu_{k0}^+, \Upsilon_{0k}^-)$
  $\mu_{k0}^+ = \langle \boldsymbol{\varphi}_{k0}^+(\mathbf{q}_{k0}^-, \mathbf{w}_0, \Upsilon_{0k}^-) \rangle$
  $\Upsilon_{k0}^+ = (\Upsilon_{k1}^-, \lambda_{k0}^+)$
  $\mathbf{q}_{k0}^+ = \mathbf{f}_{k0}^+(\mathbf{q}_{k0}^-, \mathbf{w}_0, \Upsilon_{k0}^+)$
  $\mathbf{p}_{k0}^+ = \mathbf{V}_0\mathbf{q}_{k0}^+$
  for $\ell = 1, \dots, L-1$ **do**
    $\lambda_{k\ell}^+ = T_{k\ell}^+(\mu_{k\ell}^+, \Upsilon_{k,\ell-1}^+)$
    $\mu_{k\ell}^+ = \left\langle \boldsymbol{\varphi}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k,\ell-1}^+) \right\rangle$
    $\Upsilon_{k\ell}^+ = (\Upsilon_{k,\ell-1}^+, \lambda_{k\ell}^+)$
    $\mathbf{q}_{k\ell}^+ = \mathbf{f}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k\ell}^+)$
    $\mathbf{p}_{k\ell}^+ = \mathbf{V}_\ell\mathbf{q}_{k\ell}^+$
  end for
  // Backward pass
  $\lambda_{k+1,L}^- = T_{kL}^-(\mu_{kL}^-, \Upsilon_{k,L-1}^+)$
  $\mu_{kL}^- = \left\langle \boldsymbol{\varphi}_{kL}^-(\mathbf{p}_{k,L-1}^+, \mathbf{w}_L, \Upsilon_{k,L-1}^+) \right\rangle$
  $\Upsilon_{k+1,L}^- = (\Upsilon_{k,L-1}^+, \lambda_{k+1,L}^+)$
  $\mathbf{p}_{k+1,L-1}^- = \mathbf{f}_{kL}^-(\mathbf{p}_{L-1}^0, \mathbf{p}_{k,L-1}^+, \mathbf{w}_L, \Upsilon_{k+1,L}^-)$
  $\mathbf{q}_{k+1,L-1}^- = \mathbf{V}_{L-1}^\mathsf{T}\mathbf{p}_{k+1,L-1}$
  for $\ell = L-1, \dots, 1$ **do**
    $\lambda_{k+1,\ell}^- = T_{k\ell}^-(\mu_{k\ell}^-, \Upsilon_{k+1,\ell+1}^-)$
    $\mu_{k\ell}^- = \left\langle \boldsymbol{\varphi}_{k\ell}^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k+1,\ell}^-, \mathbf{w}_\ell, \Upsilon_{k+1,\ell+1}^-) \right\rangle$
    $\Upsilon_{k+1,\ell}^- = (\Upsilon_{k+1,\ell+1}^-, \lambda_{k+1,\ell}^-)$
    $\mathbf{p}_{k+1,\ell-1}^- = \mathbf{f}_{k\ell}^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k+1,\ell}^-, \mathbf{w}_\ell, \Upsilon_{k+1,\ell}^-)$
    $\mathbf{q}_{k+1,\ell-1}^- = \mathbf{V}_{\ell-1}^\mathsf{T}\mathbf{p}_{k+1,\ell-1}^-$
  end for
end for

rows of $\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)$, we will call $\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)$ the *parameter statistic functions*. We will call the $T_{k\ell}^\pm(\cdot)$ the *parameter update functions*. The functions $\mathbf{f}_\ell^0, \mathbf{f}_{k\ell}^\pm, \boldsymbol{\varphi}_\ell^\pm$ also take as input some perturbation vectors $\mathbf{w}_\ell$.

Similar to the analysis of the ML-Mat-VAMP algorithm, we consider the following LSL analysis of Gen-ML. Specifically, we consider a sequence of runs of the recursions indexed by N. For each N, let $N_\ell = N_\ell(N)$ be the dimension of the matrix signals $\mathbf{p}_\ell^\pm$ and $\mathbf{q}_\ell^\pm$ as we assume that $\lim_{N\to\infty} \frac{N_\ell}{N} = \beta_\ell \in (0,\infty)$ is a constant so that N_ℓ scales linearly with N. Note however that the number of columns of each of the matrices $\{\mathbf{q}_\ell^0, \mathbf{p}_\ell^0, \mathbf{q}_{k\ell}^\pm, \mathbf{p}_{k\ell}^\pm\}$ is equal to a finite integer $d > 0$, which remains fixed for all N. We then make the following assumptions. See appendix B for an overview of empirical convergence of sequences which we use in the assumptions described below.

Assumption 1. For vectors in the Gen-ML algorithm (algorithm 3), we assume:

(a) The matrices $\mathbf{V}_\ell$ are Haar distributed on the set of $N_\ell \times N_\ell$ orthogonal matrices and are independent from one another and from the matrices $\mathbf{q}_0^0$, $\mathbf{q}_{0\ell}^-$, perturbation variables $\mathbf{w}_\ell$.

(b) The rows of the initial matrices $\mathbf{q}_{0\ell}^-$, and perturbation variables $\mathbf{w}_\ell$ converge jointly empirically with limits,

$$\mathbf{q}_{0\ell}^- \overset{2}{\Rightarrow} Q_{0\ell}^-, \qquad \mathbf{w}_\ell \overset{2}{\Rightarrow} W_\ell, \tag{27}$$

where $Q_{0\ell}^-$ are random vectors in $\mathbb{R}^{1\times d}$ such that $(Q_{00}^-, \ldots, Q_{0,L-1}^-)$ is jointly Gaussian. For $\ell = 0, \ldots, L-1$, the random variables $W_\ell, P_{\ell-1}^0$ and $Q_{0\ell}^-$ are all independent. We also assume that the initial parameter list converges as

$$\lim_{N\to\infty} \Upsilon_{01}^-(N) \xrightarrow{\text{a.s.}} \overline{\Upsilon}_{01}^-, \tag{28}$$

to some list $\overline{\Upsilon}_{01}^-$. The limit (28) means that every element in the list $\lambda(N) \in \Upsilon_{01}^-(N)$ converges to a limit $\lambda(N) \to \overline{\lambda} \in \overline{\Upsilon}_{01}^-$ as $N \to \infty$ almost surely.

(c) The *matrix update functions* $\mathbf{f}_{k\ell}^\pm(\cdot)$ and *parameter update functions* $\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)$ act row-wise. For example, in the kth forward pass, at stage ℓ, we assume that for each output row n,

$$\left[\mathbf{f}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k\ell}^+)\right]_{n:} = f_{k\ell}^+(\mathbf{p}_{\ell-1,n:}^0, \mathbf{p}_{k,\ell-1,n:}^+, \mathbf{q}_{k\ell,n:}^-, \mathbf{w}_{\ell,n:}, \Upsilon_{k\ell}^+)$$

$$\left[\boldsymbol{\varphi}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k\ell}^+)\right]_{n:} = \varphi_{k\ell}^+(\mathbf{p}_{\ell-1,n:}^0, \mathbf{p}_{k,\ell-1,n:}^+, \mathbf{q}_{k\ell,n:}^-, \mathbf{w}_{\ell,n:}, \Upsilon_{k\ell}^+),$$

for some $\mathbb{R}^{1\times d}$-valued functions $f_{k\ell}^+(\cdot)$ and $\varphi_{k\ell}^+(\cdot)$. Similar definitions apply in the reverse directions and for the initial vector functions $\mathbf{f}_\ell^0(\cdot)$. We will call $f_{k\ell}^\pm(\cdot)$ the *matrix update row-wise functions* and $\varphi_{k\ell}^\pm(\cdot)$ the *parameter update row-wise functions*.

Next we define a set of *deterministic* constants $\{\mathbf{K}_{k\ell}^+, \boldsymbol{\tau}_{k\ell}^-, \overline{\mu}_{k\ell}^\pm, \overline{\Upsilon}_{kl}^\pm, \boldsymbol{\tau}_\ell^0\}$ and $\mathbb{R}^{1\times d}$-valued random vectors $\{Q_\ell^0, P_\ell^0, Q_{k\ell}^\pm, P_\ell^\pm\}$ which are recursively defined through algorithm 4, which we call the general multi-layer matrix (*Gen-ML-Mat*) SE. These recursions in algorithm closely mirror those in the Gen-ML-Mat algorithm (algorithm 3). The matrices $\mathbf{q}_{k\ell}^\pm$ and $\mathbf{p}_{k\ell}^\pm$ are replaced by random vectors $Q_{k\ell}^\pm$ and $P_{k\ell}^\pm$; the matrix and parameter update functions $\mathbf{f}_{k\ell}^\pm(\cdot)$ and $\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)$ are replaced by their row-wise functions $f_{k\ell}^\pm(\cdot)$ and $\varphi_{k\ell}^\pm(\cdot)$; and the parameters $\lambda_{k\ell}^\pm$ are replaced by their limits $\overline{\lambda}_{k\ell}^\pm$. We refer to $\{Q_\ell^0, P_\ell^0\}$ as *true random vectors* and $\{Q_{k\ell}^\pm, P_{kl}^\pm\}$ as *iterated random vectors*. The signal flow graph

Algorithm 4. Gen-ML-Mat SE.

Require: matrix update row-wise functions $f_\ell^0(\cdot)$ and $f_{k\ell}^{\pm}(\cdot)$, parameter statistic row-wise functions $\varphi_{k\ell}^{\pm}(\cdot)$, parameter update functions $T_{k\ell}^{\pm}(\cdot)$, initial parameter list limit: $\overline{\Upsilon}_{01}^{-}$, initial random variables W_ℓ, $Q_{0\ell}^{-}$, $\ell = 0, \ldots, L-1$.

// `Initial pass`
$Q_0^0 = f_0^0(W_0, \overline{\Upsilon}_{01}^{-}),\ P_0^0 \sim \mathcal{N}(0, \tau_0^0),\ \tau_0^0 = \mathbb{E}(Q_0^0)^2$
for $\ell = 1, \ldots, L-1$ **do**
  $Q_\ell^0 = f_\ell^0(P_{\ell-1}^0, W_\ell, \overline{\Upsilon}_{01}^{-})$
  $P_\ell^0 \sim \mathcal{N}(0, \tau_\ell^0),\ \tau_\ell^0 = \mathrm{Cov}(Q_\ell^0)$
end for

for $k = 0, 1, \ldots$ **do**
  // `Forward pass`
  $\overline{\lambda}_{k0}^{+} = T_{k0}^{+}(\overline{\mu}_{k0}^{+}, \overline{\Upsilon}_{0k}^{-})$
  $\overline{\mu}_{k0}^{+} = \mathbb{E}(\varphi_{k0}^{+}(Q_{k0}^{-}, W_0, \overline{\Upsilon}_{0k}^{-}))$
  $\overline{\Upsilon}_{k0}^{+} = (\overline{\Upsilon}_{k1}^{-}, \overline{\lambda}_{k0}^{+})$
  $Q_{k0}^{+} = f_{k0}^{+}(Q_{k0}^{-}, W_0, \overline{\Upsilon}_{k0}^{+})$
  $(P_0^0, P_{k0}^{+}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k0}^{+}),\ \mathbf{K}_{k0}^{+} = \mathrm{Cov}(Q_0^0, Q_{k0}^{+})$
  for $\ell = 1, \ldots, L-1$ **do**
    $\overline{\lambda}_{k\ell}^{+} = T_{k\ell}^{+}(\overline{\mu}_{k\ell}^{+}, \overline{\Upsilon}_{k,\ell-1}^{+})$
    $\overline{\mu}_{k\ell}^{+} = \mathbb{E}(\varphi_{k\ell}^{+}(P_{\ell-1}^0, P_{k,\ell-1}^{+}, Q_{k\ell}^{-}, W_\ell, \overline{\Upsilon}_{k,\ell-1}^{+}))$
    $\overline{\Upsilon}_{k\ell}^{+} = (\overline{\Upsilon}_{k,\ell-1}^{+}, \overline{\lambda}_{k\ell}^{+})$
    $Q_{k\ell}^{+} = f_{k\ell}^{+}(P_{\ell-1}^0, P_{k,\ell-1}^{+}, Q_{k\ell}^{-}, W_\ell, \overline{\Upsilon}_{k\ell}^{+})$
    $(P_\ell^0, P_{k\ell}^{+}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k\ell}^{+}),\ \mathbf{K}_{k\ell}^{+} = \mathrm{Cov}(Q_\ell^0, Q_{k\ell}^{+})$
  end for

  // `Backward pass`
  $\overline{\lambda}_{k+1,L}^{-} = T_{kL}^{-}(\overline{\mu}_{kL}^{-}, \overline{\Upsilon}_{k,L-1}^{+})$
  $\overline{\mu}_{kL}^{-} = \mathbb{E}(\varphi_{kL}^{-}(P_{L-1}^0, P_{k,L-1}^{+}, W_L, \overline{\Upsilon}_{k,L-1}^{+}))$
  $\overline{\Upsilon}_{k+1,L}^{-} = (\overline{\Upsilon}_{k,L-1}^{+}, \overline{\lambda}_{k+1,L}^{+})$
  $P_{k+1,L-1}^{-} = f_{kL}^{-}(P_{L-1}^0, P_{k,L-1}^{+}, W_L, \overline{\Upsilon}_{k+1,L}^{-})$
  $Q_{k+1,L-1}^{-} \sim \mathcal{N}(0, \tau_{k+1,L-1}^{-}),\ \tau_{k+1,L-1}^{-} = \mathrm{Cov}(P_{k+1,L-1}^{-})$
  for $\ell = L-1, \ldots, 1$ **do**
    $\overline{\lambda}_{k+1,\ell}^{-} = T_{k\ell}^{-}(\overline{\mu}_{k\ell}^{-}, \overline{\Upsilon}_{k+1,\ell+1}^{-})$
    $\overline{\mu}_{k\ell}^{-} = \mathbb{E}(\varphi_{k\ell}^{-}(P_{\ell-1}^0, P_{k,\ell-1}^{+}, Q_{k+1,\ell}^{-}, W_\ell, \overline{\Upsilon}_{k+1,\ell+1}^{-}))$
    $\overline{\Upsilon}_{k+1,\ell}^{-} = (\overline{\Upsilon}_{k+1,\ell+1}^{-}, \overline{\lambda}_{k+1,\ell}^{-})$
    $P_{k+1,\ell-1}^{-} = f_{k\ell}^{-}(P_{\ell-1}^0, P_{k,\ell-1}^{+}, Q_{k+1,\ell}^{-}, W_\ell, \overline{\Upsilon}_{k+1,\ell}^{-})$
    $Q_{k+1,\ell-1}^{-} \sim \mathcal{N}(0, \tau_{k+1,\ell-1}^{-}),\ \tau_{k+1,\ell-1}^{-} = \mathrm{Cov}(P_{k+1,\ell-1}^{-})$
  end for
end for

for the true and iterated random variables in algorithm 4 is given in the bottom panel of figure 3. The iteration index k for the iterated random variables $\{Q_{k\ell}^{\pm}, P_{kl}^{\pm}\}$ to simplify notation.

J. Stat. Mech. (2021) 124004

We also assume the following about the behavior of row-wise functions around the quantities defined in algorithm 4. The iteration index k has been dropped for simplifying notation.

Assumption 2. For row-wise functions f, φ and parameter update functions T we assume:

(a) $T^{\pm}_{k\ell}(\mu^{\pm}_{k\ell}, \cdot)$ are continuous at $\mu^{\pm}_{k\ell} = \overline{\mu}^{\pm}_{k\ell}$

(b) $f^{+}_{k\ell}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k\ell}, w_\ell, \Upsilon^{+}_{k\ell})$, $\frac{\partial f^{+}_{k\ell}}{\partial q^{-}_{k\ell}}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k\ell}, w_\ell, \Upsilon^{+}_{k\ell})$ and $\varphi^{+}_{k\ell}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k\ell}, w_\ell, \Upsilon^{+}_{k,\ell-1})$ are uniformly Lipschitz continuous in $(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k\ell}, w_\ell)$ at $\Upsilon^{+}_{k\ell} = \overline{\Upsilon}^{+}_{k\ell}$, $\Upsilon^{+}_{k,\ell-1} = \overline{\Upsilon}^{+}_{k,\ell-1}$. Similarly, $f^{-}_{k+1,\ell}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k+1,\ell}, w_\ell, \Upsilon^{-}_{k\ell})$, $\frac{\partial f^{-}_{k\ell}}{\partial p^{+}_{k,\ell-1}}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k+1,\ell}, w_\ell, \Upsilon^{-}_{k\ell})$, and $\varphi^{-}_{k\ell}(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k+1,\ell}, w_\ell, \Upsilon^{-}_{k+1,\ell+1})$ are uniformly Lipschitz continuous in $(p^{0}_{\ell-1}, p^{+}_{k,\ell-1}, q^{-}_{k+1,\ell}, w_\ell)$ at $\Upsilon^{-}_{k\ell} = \overline{\Upsilon}^{-}_{k\ell}$, $\Upsilon^{-}_{k+1,\ell+1} = \overline{\Upsilon}^{-}_{k+1,\ell+1}$.

(c) $f^{0}_{\ell}(p^{0}_{\ell-1}, w_\ell, \Upsilon^{-}_{01})$ are uniformly Lipschitz continuous in $(p^{0}_{k,\ell-1}, w_\ell)$ at $\Upsilon^{-}_{k+1,\ell} = \overline{\Upsilon}^{-}_{k+1,\ell}$.

(d) Matrix update functions $\mathbf{f}^{\pm}_{k\ell}$ are *asymptotically divergence free* meaning

$$\lim_{N\to\infty} \left\langle \frac{\partial \mathbf{f}^{+}_{k\ell}}{\partial \mathbf{q}^{-}_{k\ell}}(\mathbf{p}^{+}_{k,\ell-1}, \mathbf{q}^{-}_{k\ell}, \mathbf{w}_\ell, \overline{\Upsilon}^{+}_{k\ell}) \right\rangle = \mathbf{0}, \qquad \lim_{N\to\infty} \left\langle \frac{\partial \mathbf{f}^{-}_{k\ell}}{\partial \mathbf{p}^{+}_{k,\ell-1}}(\mathbf{p}^{+}_{k,\ell-1}, \mathbf{q}^{-}_{k+1,\ell}, \mathbf{w}_\ell, \overline{\Upsilon}^{-}_{k\ell}) \right\rangle = \mathbf{0} \tag{29}$$

We are now ready to state the general result regarding the empirical convergence of the true and iterated vectors from algorithm 3 in terms of random variables defined in algorithm 4.

Theorem 2. *Consider the iterates of the Gen-ML recursion (algorithm 3) and the corresponding random variables and parameter limits defined by the SE recursions (algorithm 4) under assumptions 1 and 2. Then,*

(a) For any fixed $k \geqslant 0$ and fixed $\ell = 1, \ldots, L-1$, the parameter list $\Upsilon^{+}_{k\ell}$ converges as

$$\lim_{N\to\infty} \Upsilon^{+}_{k\ell} = \overline{\Upsilon}^{+}_{k\ell} \tag{30}$$

almost surely. Also, the rows of $\boldsymbol{w}_\ell$, $\mathbf{p}^{0}_{\ell-1}$, $\mathbf{q}^{0}_{\ell}$, $\mathbf{p}^{+}_{0,\ell-1}, \ldots, \mathbf{p}^{+}_{k,\ell-1}$ and $\mathbf{q}^{\pm}_{0\ell}, \ldots, \mathbf{q}^{\pm}_{k\ell}$ almost surely jointly converge empirically with limits,

$$(\mathbf{p}^{0}_{\ell-1}, \mathbf{p}^{+}_{i,\ell-1}, \mathbf{q}^{-}_{j\ell}, \mathbf{q}^{0}_{\ell}, \mathbf{q}^{+}_{j\ell}) \overset{2}{\Rightarrow} (P^{0}_{\ell-1}, P^{+}_{i,\ell-1}, Q^{-}_{j\ell}, Q^{0}_{\ell}, Q^{+}_{j\ell}), \tag{31}$$

for all $0 \leqslant i, j \leqslant k$, where the variables $P^{0}_{\ell-1}$, $P^{+}_{i,\ell-1}$ and $Q^{-}_{j\ell}$ are zero-mean jointly Gaussian random variables independent of W_ℓ and with covariance matrix given by

$$\mathrm{Cov}(P^{0}_{\ell-1}, P^{+}_{i,\ell-1}) = \mathbf{K}^{+}_{i,\ell-1}, \qquad \mathbb{E}(Q^{-}_{j\ell})^2 = \boldsymbol{\tau}^{-}_{j\ell}, \qquad \mathbb{E}(P^{+\mathsf{T}}_{i,\ell-1} Q^{-}_{j\ell}) = \mathbf{0}, \qquad \mathbb{E}(P^{0\mathsf{T}}_{\ell-1} Q^{-}_{j\ell}) = \mathbf{0}, \tag{32}$$

and Q^{0}_{ℓ}, $Q^{+}_{j\ell}$ are the random variable in lines 4, 19, i.e.

$$Q^{0}_{\ell} = f^{0}_{\ell}(P^{0}_{\ell-1}, W_\ell), \qquad Q^{+}_{j\ell} = f^{+}_{j\ell}(P^{0}_{\ell-1}, P^{+}_{j,\ell-1}, Q^{-}_{j\ell}, W_\ell, \overline{\Upsilon}^{+}_{j\ell}). \tag{33}$$

J. Stat. Mech. (2021) 124004

An identical result holds for $\ell = 0$ with all the variables $\mathbf{p}^+_{i,\ell-1}$ and $P^+_{i,\ell-1}$ removed.

(b) For any fixed $k \geqslant 1$ and fixed $\ell = 1, \ldots, L-1$, the parameter lists $\Upsilon^-_{k\ell}$ converge as

$$\lim_{N\to\infty} \Upsilon^-_{k\ell} = \overline{\Upsilon}^-_{k\ell} \tag{34}$$

almost surely. Also, the rows of $\mathbf{w}_\ell$, $\mathbf{p}^0_{\ell-1}$, $\mathbf{p}^\pm_{0,\ell-1}, \ldots, \mathbf{p}^\pm_{k-1,\ell-1}$, and $\mathbf{q}^-_{0\ell}, \ldots, \mathbf{q}^-_{k\ell}$ almost surely jointly converge empirically with limits,

$$(\mathbf{p}^0_{\ell-1}, \mathbf{p}^+_{i,\ell-1}, \mathbf{q}^-_{j\ell}, \mathbf{p}^-_{j,\ell-1}) \overset{2}{\Rightarrow} (P^0_{\ell-1}, P^+_{i,\ell-1}, Q^-_{j\ell}, P^-_{j,\ell-1}), \tag{35}$$

for all $0 \leqslant i \leqslant k-1$ and $0 \leqslant j \leqslant k$, where the variables $P^0_{\ell-1}$, $P^+_{i,\ell-1}$ and $Q^-_{j\ell}$ are zero-mean jointly Gaussian random variables independent of W_ℓ and with covariance matrix given by equation (32) and $P^-_{j\ell}$ is the random variable in line 32:

$$P^-_{j\ell} = f^-_{j\ell}(P^0_{\ell-1}, P^+_{j-1,\ell-1}, Q^-_{j\ell}, W_\ell, \overline{\Upsilon}^-_{j\ell}). \tag{36}$$

An identical result holds for $\ell = L$ with all the variables $\mathbf{q}^-_{j\ell}$ and $Q^-_{j\ell}$ removed.

For $k = 0$, $\Upsilon^-_{01} \to \overline{\Upsilon}^-_{01}$ almost surely, and the rows $\{(\mathbf{w}_{\ell,n:}, \mathbf{p}^0_{\ell-1,n:}, \mathbf{q}^-_{j\ell,n:})\}_{n=1}^N$ empirically converge to independent random variables $(W_\ell, P^0_{\ell-1}, Q^-_{0\ell})$.

Proof. Appendix E is dedicated to proving this result. □

Appendix E. Proof of theorem 2

The proof proceeds using mathematical induction. It largely mimics the proof for the case of $d = 1$ which were the convergence results in theorem 5 in [34]. However, in the case of $d > 1$, we observe that several quantities which were scalars in proving theorem 5 in [34] are now matrices. Due to the non-commutativity of these matrix quantities, we re-state the whole prove, while modifying the requisite matrix terms appropriately.

E.1. Overview of the induction sequence

The proof is similar to that of theorem 4 in [35], which provides a SE analysis for VAMP on a single-layer network. The critical challenge here is to extend that proof to multi-layer recursions. Many of the ideas in the two proofs are similar, so we highlight only the key differences between the two.

Similar to the SE analysis of VAMP in [35], we use an induction argument. However, for the multi-layer proof, we must index over both the iteration index k and layer index ℓ. To this end, let $\mathcal{H}^+_{k\ell}$ and $\mathcal{H}^-_{k\ell}$ be the hypotheses:

- $\mathcal{H}^+_{k\ell}$: the hypothesis that theorem 2(a) is true for a given k and ℓ, where $0 \leqslant \ell \leqslant L-1$.
- $\mathcal{H}^-_{k\ell}$: the hypothesis that theorem 2(b) is true for a given k and ℓ, where $1 \leqslant \ell \leqslant L$.

We prove these hypotheses by induction via a sequence of implications,

$$\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^L \cdots \Rightarrow \mathcal{H}^-_{k1} \Rightarrow \mathcal{H}^+_{k0} \Rightarrow \cdots \Rightarrow \mathcal{H}^+_{k,L-1} \Rightarrow \mathcal{H}^-_{k+1,L} \Rightarrow \cdots \Rightarrow \mathcal{H}^-_{k+1,1} \Rightarrow \cdots, \tag{37}$$

beginning with the hypotheses $\{\mathcal{H}^-_{0\ell}\}$ for all $\ell = 1, \ldots, L-1$.

J. Stat. Mech. (2021) 124004

E.2. Base case: proof of $\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^{L}$

The base case corresponds to the hypotheses $\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^{L}$. Note that theorem 2(b) states that for $k=0$, we need $\Upsilon^-_{01} \to \overline{\Upsilon}^-_{01}$ almost surely, and $\{(\mathbf{w}_{\ell,n:}, \mathbf{p}^0_{\ell-1,n:}, \mathbf{q}^-_{j\ell,n:})\}_{n=1}^{N}$ empirically converge to independent random variables $(W_\ell, P^0_{\ell-1}, Q^-_{0\ell})$. These follow directly from equations (27) and (28) in assumption 1(a).

E.3. Inductive step: proof of $\mathcal{H}^+_{k,\ell+1}$

Fix a layer index $\ell = 1, \ldots, L-1$ and an iteration index $k = 0, 1, \ldots$. We show the implication $\cdots \implies \mathcal{H}^+_{k,\ell+1}$ in (37). All other implications can be proven similarly using symmetry arguments.

Definition 4 (induction hypothesis)**.** The hypotheses prior to $\mathcal{H}^+_{k,\ell+1}$ in the sequence (37), but not including $\mathcal{H}^+_{k,\ell+1}$, are true.

The inductive step then corresponds to the following result.

Lemma 3. *Under the induction hypothesis, $\mathcal{H}^+_{k,\ell+1}$ holds.*

Before proving the inductive step in lemma 3, we prove two intermediate lemmas. Let us start by defining some notation. Define $\mathbf{P}^+_{k\ell} := [\mathbf{p}^+_{0\ell} \cdots \mathbf{p}^+_{k\ell}] \in \mathbb{R}^{N_\ell \times (k+1)d}$, be a matrix whose column blocks are the first $k+1$ values of the matrix $\mathbf{p}^+_\ell$. We define the matrices $\mathbf{P}^-_{k\ell}$, $\mathbf{Q}^+_{k\ell}$ and $\mathbf{Q}^-_{k\ell}$ in a similar manner with values of $\mathbf{p}^-_\ell$, $\mathbf{q}^+_\ell$ and $\mathbf{q}^-_\ell$ respectively.

Note that except the initial matrices $\{\mathbf{w}_\ell, \mathbf{q}^-_{0\ell}\}_{\ell=1}^{L}$, all later iterates in algorithm 3 are random due to the randomness of $\mathbf{V}_\ell$. Let $\mathfrak{G}^\pm_{k\ell}$ denote the collection of random variables associated with the hypotheses, $\mathcal{H}^\pm_{k\ell}$. That is, for $\ell = 1, \ldots, L-1$,

$$\mathfrak{G}^+_{k\ell} := \left\{\mathbf{w}_\ell, \mathbf{p}^0_{\ell-1}, \mathbf{P}^+_{k,\ell-1}, \mathbf{q}^0_\ell, \mathbf{Q}^-_{k\ell}, \mathbf{Q}^+_{k\ell}\right\},$$
$$\mathfrak{G}^-_{k\ell} := \left\{\mathbf{w}_\ell, \mathbf{p}^0_{\ell-1}, \mathbf{P}^+_{k-1,\ell-1}, \mathbf{q}^0_\ell, \mathbf{Q}^-_{k\ell}, \mathbf{P}^-_{k,\ell-1}\right\}.$$

For $\ell = 0$ and $\ell = L$ we set, $\mathfrak{G}^+_{k0} := \left\{\mathbf{w}_0, \mathbf{Q}^-_{k0}, \mathbf{Q}^+_{k0}\right\}$, $\quad \mathfrak{G}^-_{kL} := \left\{\mathbf{w}_L, \mathbf{p}^0_{L-1}, \mathbf{P}^+_{k-1,L-1}, \mathbf{P}^-_{k,L-1}\right\}$.

Let $\overline{\mathfrak{G}}^+_{k\ell}$ be the sigma algebra generated by the union of all the sets $\mathfrak{G}^\pm_{k'\ell'}$ as they have appeared in the sequence (37) up to and including the final set $\mathfrak{G}^+_{k\ell}$. Thus, the sigma algebra $\overline{\mathfrak{G}}^+_{k\ell}$ contains all *information* produced by algorithm 3 immediately *before* line 20 in layer ℓ of iteration k. Note also that the random variables in algorithm 4 immediately before defining $P^+_{k,\ell}$ in line 20 are all $\overline{\mathfrak{G}}^+_{k\ell}$ measurable.

Observe that the matrix $\mathbf{V}_\ell$ in algorithm 3 appears only during matrix-vector multiplications in lines 20 and 32. If we define the matrices, $\mathbf{A}_{k\ell} := \left[\mathbf{p}^0_\ell, \mathbf{P}^+_{k-1,\ell}\ \mathbf{P}^-_{k\ell}\right]$, $\mathbf{B}_{k\ell} := \left[\mathbf{q}^0_\ell, \mathbf{Q}^+_{k-1,\ell}\ \mathbf{Q}^-_{k\ell}\right]$, all the matrices in the set $\overline{\mathfrak{G}}^+_{k\ell}$ will be unchanged for all matrices $\mathbf{V}_\ell$ satisfying the linear constraints

$$\mathbf{A}_{k\ell} = \mathbf{V}_\ell \mathbf{B}_{k\ell}. \tag{38}$$

Hence, the conditional distribution of $\mathbf{V}_\ell$ given $\overline{\mathfrak{G}}^+_{k\ell}$ is precisely the uniform distribution on the set of orthogonal matrices satisfying (38). The matrices $\mathbf{A}_{k\ell}$ and $\mathbf{B}_{k\ell}$ are of dimensions $N_\ell \times (2k+2)d$. From lemmas 3 and 4 in [35], this conditional distribution is

J. Stat. Mech. (2021) 124004

given by

$$\mathbf{V}_\ell|_{\overline{\mathfrak{G}}^+_{k\ell}} \stackrel{d}{=} \mathbf{A}_{k\ell}(\mathbf{A}_{k\ell}^\mathsf{T}\mathbf{A}_{k\ell})^{-1}\mathbf{B}_{k\ell}^\mathsf{T} + \mathbf{U}_{\mathbf{A}^\perp_{k\ell}}\widetilde{\mathbf{V}}_\ell\mathbf{U}_{\mathbf{B}^\perp_{k\ell}}^\mathsf{T}, \tag{39}$$

where $\mathbf{U}_{\mathbf{A}^\perp_{k\ell}}$ and $\mathbf{U}_{\mathbf{B}^\perp_{k\ell}}$ are $N_\ell \times (N_\ell - (2k+2)\,d)$ matrices whose columns are an orthonormal basis for $\text{Range}(\mathbf{A}_{k\ell})^\perp$ and $\text{Range}(\mathbf{B}_{k\ell})^\perp$. The matrix $\widetilde{\mathbf{V}}_\ell$ is Haar distributed on the set of $(N_\ell - (2k+2)\,d) \times (N_\ell - (2k+2)\,d)$ orthogonal matrices and is independent of $\overline{\mathfrak{G}}^+_{k\ell}$.

Next, similar to the proof of theorem 4 in [35], we can use (39) to write the conditional distribution of $\mathbf{p}^+_{k\ell}$ (from line 20 of algorithm 3) given $\overline{\mathfrak{G}}^+_{k\ell}$ as a sum of two terms

$$\mathbf{p}^+_{k\ell}|_{\overline{\mathfrak{G}}^+_{k\ell}} = \mathbf{V}_\ell|_{\overline{\mathfrak{G}}^+_{k\ell}} \qquad \mathbf{q}^+_{k\ell} \stackrel{d}{=} \mathbf{p}^{+\text{det}}_{k\ell} + \mathbf{p}^{+\text{ran}}_{k\ell}, \tag{40a}$$

$$\mathbf{p}^{+\text{det}}_{k\ell} := \mathbf{A}_{k\ell}(\mathbf{B}_{k\ell}^\mathsf{T}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^\mathsf{T}\mathbf{q}^+_{k\ell} \tag{40b}$$

$$\mathbf{p}^{+\text{ran}}_{k\ell} := \mathbf{U}_{\mathbf{B}^\perp_k}\widetilde{\mathbf{V}}_\ell^\mathsf{T}\mathbf{U}_{\mathbf{A}^\perp_k}^\mathsf{T}\mathbf{q}^+_{k\ell} \tag{40c}$$

where we call $\mathbf{p}^{+\text{det}}_{k\ell}$ the *deterministic* term and $\mathbf{p}^{+\text{ran}}_{k\ell}$ the *random* term. The next two lemmas characterize the limiting distributions of the deterministic and random terms.

Lemma 4. *Under the induction hypothesis, the rows of the 'deterministic' term* $\mathbf{p}^{+\text{det}}_{k\ell}$ *along with the rows of the matrices in* $\overline{\mathfrak{G}}^+_{k\ell}$ *converge empirically. In addition, there exists constant* $d \times d$ *matrices* $\beta^+_{0\ell}, \ldots, \beta^+_{k-1,\ell}$ *such that*

$$\mathbf{p}^{+\text{det}}_{k\ell} \stackrel{2}{\Rightarrow} P^{+\text{det}}_{k\ell} := P^0_\ell\beta^0_\ell + \sum_{i=0}^{k-1} P^+_{i\ell}\beta_{i\ell}, \tag{41}$$

where $P^{+\text{det}}_{k\ell} \in \mathbb{R}^{1\times d}$ *is the limiting random vector for the rows of* $\mathbf{p}^{\text{det}}_{k\ell}$.

Proof. The proof is similar that of lemma 6 in [35], but we go over the details as there are some important differences in the multi-layer matrix case. Define $\widetilde{\mathbf{P}}^+_{k-1,\ell} = \left[\mathbf{p}^0_\ell,\ \mathbf{P}^+_{k-1,\ell}\right]$, $\widetilde{\mathbf{Q}}^+_{k-1,\ell} = \left[\mathbf{q}^0_\ell,\ \mathbf{Q}^+_{k-1,\ell}\right]$, which are the matrices in $\mathbb{R}^{N_\ell\times(k+1)d}$. We can then write $\mathbf{A}_{k\ell}$ and $\mathbf{B}_{k\ell}$ from (38) as

$$\mathbf{A}_{k\ell} := \left[\widetilde{\mathbf{P}}^+_{k-1,\ell}\ \mathbf{P}^-_{k\ell}\right], \qquad \mathbf{B}_{k\ell} := \left[\widetilde{\mathbf{Q}}^+_{k-1,\ell}\ \mathbf{Q}^-_{k\ell}\right]. \tag{42}$$

We first evaluate the asymptotic values of various terms in (40b). By definition of $\mathbf{B}_{k\ell}$ in (42),

$$\mathbf{B}_{k\ell}^\mathsf{T}\mathbf{B}_{k\ell} = \begin{bmatrix} (\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\widetilde{\mathbf{Q}}^+_{k-1,\ell} & (\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\mathbf{Q}^-_{k\ell} \\ (\mathbf{Q}^-_{k\ell})^\mathsf{T}\widetilde{\mathbf{Q}}^+_{k-1,\ell} & (\mathbf{Q}^-_{k\ell})^\mathsf{T}\mathbf{Q}^-_{k\ell} \end{bmatrix}.$$

We can then evaluate the asymptotic values of these terms as follows: for $0 \leqslant i, j \leqslant k-1$ the asymptotic value of the $(i+2, j+2)^{\text{nd}}d \times d$ block of the matrix $(\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\widetilde{\mathbf{Q}}^+_{k-1,\ell}$ is

J. Stat. Mech. (2021) 124004

$$\lim_{N\to\infty}\frac{1}{N_\ell}\left[(\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\widetilde{\mathbf{Q}}^+_{k-1,\ell}\right]_{i+2,j+2}\overset{(a)}{=}\lim_{N\to\infty}\frac{1}{N_\ell}(\mathbf{q}^+_{i\ell})^\mathsf{T}\mathbf{q}^+_{j\ell}$$
$$=\lim_{N\to\infty}\frac{1}{N_\ell}\sum_{n=1}^{N_\ell}[\mathbf{q}^+_{i\ell}]_{n:}[\mathbf{q}^+_{j\ell}]^\mathsf{T}_{n:}\overset{(b)}{=}\mathbb{E}\left[Q^{+\mathsf{T}}_{i\ell}Q^+_{j\ell}\right]$$

where (a) follows since the $(i+2)$th column block of $\widetilde{\mathbf{Q}}^+_{k-1,\ell}$ is $\mathbf{q}^+_{i\ell}$, and (b) follows due to the empirical convergence assumption in (31). Also, since the first column block of $\widetilde{\mathbf{Q}}^+_{k-1,\ell}$ is $\mathbf{q}^0_\ell$, we obtain that

$$\lim_{N_\ell\to\infty}\frac{1}{N_\ell}(\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\widetilde{\mathbf{Q}}^+_{k-1,\ell}=\mathbf{R}^+_{k-1,\ell}\quad\text{and}\quad\lim_{N_\ell\to\infty}\frac{1}{N_\ell}(\mathbf{Q}^-_{k\ell})^\mathsf{T}\mathbf{Q}^-_{k\ell}=\mathbf{R}^-_{k\ell},\tag{43}$$

where $\mathbf{R}^+_{k-1,\ell}\in\mathbb{R}^{(k+1)d\times(k+1)d}$ is the covariance matrix of $\left[Q^0_\ell\ Q^+_{0\ell}\ \dots Q^+_{k-1,\ell}\right]$, and $\mathbf{R}^-_{k\ell}\in\mathbb{R}^{(k+1)d\times(k+1)d}$ is the covariance matrix of $[Q^-_{0\ell}\ Q^-_{1\ell}\ \dots Q^-_{k\ell}]$. For the matrix $(\widetilde{\mathbf{Q}}^+_{k-1,\ell})^\mathsf{T}\mathbf{Q}^-_{k\ell}$, first observe that the limit of the divergence free condition (29) implies

$$\mathbb{E}\left[\frac{\partial f^+_{i\ell}(P^+_{i,\ell-1},Q^-_{i\ell},W_\ell,\overline{\Upsilon}_{i\ell})}{\partial Q^-_{i\ell}}\right]=\lim_{N_\ell\to\infty}\left\langle\frac{\partial\mathbf{f}^+_{i\ell}(\mathbf{p}^+_{i,\ell-1},\mathbf{q}^-_{i\ell},\mathbf{w}_\ell,\overline{\Upsilon}^+_{i\ell})}{\partial\mathbf{q}^-_{i\ell}}\right\rangle=\mathbf{0},\tag{44}$$

for any i. Also, by the induction hypothesis $\mathcal{H}^+_{k\ell}$,

$$\mathbb{E}(P^{+\mathsf{T}}_{i,\ell-1}Q^-_{j\ell})=\mathbf{0},\qquad\mathbb{E}(P^{0\mathsf{T}}_{\ell-1}Q^-_{j\ell})=\mathbf{0},\tag{45}$$

for all $0\leqslant i,j\leqslant k$. Therefore using (33), the cross-terms $\mathbb{E}(Q^{+\mathsf{T}}_{i\ell}Q^-_{j\ell})$ are given by

$$\begin{aligned}\mathbb{E}(f^+_{i\ell}(P^0_{\ell-1},P^+_{i,\ell-1},Q^-_{i\ell},W_\ell,\overline{\Upsilon}_{i\ell})^\mathsf{T}Q^-_{j\ell})\overset{(a)}{=}&\,\mathbb{E}\left[\frac{\partial f^+_{i\ell}(P^0_{\ell-1},P^+_{i,\ell-1},Q^-_{i\ell},W_\ell,\overline{\Upsilon}^+_{i\ell})}{\partial P^0_{\ell-1}}\right]\mathbb{E}(P^{0\mathsf{T}}_{\ell-1}Q^-_{j\ell})\\&+\mathbb{E}\left[\frac{\partial f^+_{i\ell}(P^0_{\ell-1},P^+_{i,\ell-1},Q^-_{i\ell},W_\ell,\overline{\Upsilon}^+_{i\ell})}{\partial P^+_{i,\ell-1}}\right]\mathbb{E}(P^{+\mathsf{T}}_{i,\ell-1}Q^-_{j\ell})\\&+\mathbb{E}\left[\frac{\partial f^+_{i\ell}(P^0_{\ell-1},P^+_{i,\ell-1},Q^-_{i\ell},W_\ell,\overline{\Upsilon}^+_{i\ell})}{\partial Q^-_{i\ell}}\right]\mathbb{E}(Q^{-\mathsf{T}}_{i\ell}Q^-_{j\ell})\overset{(b)}{=}\mathbf{0},\end{aligned}\tag{46}$$

(a) follows from a multivariate version of Stein's lemma (equation (2) in [23]); and (b) follows from (44), and (45). Consequently,

$$\lim_{N_\ell\to\infty}\frac{1}{N_\ell}\mathbf{B}^\mathsf{T}_{k\ell}\mathbf{B}_{k\ell}=\begin{bmatrix}\mathbf{R}^+_{k-1,\ell}&\mathbf{0}\\\mathbf{0}&\mathbf{R}^-_{k\ell}\end{bmatrix},\quad\text{and}\quad\lim_{N_\ell\to\infty}\frac{1}{N_\ell}\mathbf{B}^\mathsf{T}_{k\ell}\mathbf{q}^+_{k\ell}=\begin{bmatrix}\mathbf{b}^+_{k\ell}\\\mathbf{0}\end{bmatrix},\tag{47}$$

where $\mathbf{b}^+_{k\ell}:=\left[\mathbb{E}(Q^{+\mathsf{T}}_{0\ell}Q^+_{k\ell})\ \mathbb{E}(Q^{+\mathsf{T}}_{1\ell}Q^+_{k\ell})\ \dots\ \mathbb{E}(Q^{+\mathsf{T}}_{k-1,\ell}Q^+_{k\ell})\right]^\mathsf{T}$, is the matrix of correlations. We again have $\mathbf{0}$ in the second term because $\mathbb{E}[Q^{+\mathsf{T}}_{i\ell}Q^-_{j\ell}]=\mathbf{0}$ for all $0\leqslant i,j\leqslant k$. Hence

J. Stat. Mech. (2021) 124004

we have

$$\lim_{N_\ell\to\infty}(\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{q}_{k\ell}^{+}=\begin{bmatrix}\boldsymbol{\beta}_{k\ell}^{+}\\ \mathbf{0}\end{bmatrix},\qquad \boldsymbol{\beta}_{k\ell}^{+}:=\left[\mathbf{R}_{k\text{-}1,\ell}^{+}\right]^{-1}\mathbf{b}_{k\ell}^{+}. \tag{48}$$

Therefore, $\mathbf{p}_{k\ell}^{+\det}$ equals

$$\begin{aligned}\mathbf{A}_{k\ell}(\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{q}_{k\ell}^{+}&=\left[\widetilde{\mathbf{P}}_{k\text{-}1,\ell}^{+}\ \mathbf{P}_{k,\ell}^{-}\right]\begin{bmatrix}\boldsymbol{\beta}_{k\ell}^{+}\\ \mathbf{0}\end{bmatrix}+O\left(\frac{1}{N_\ell}\right)\\&=\mathbf{p}_{\ell}^{0}\beta_{\ell}^{0}+\sum_{i=0}^{k\text{-}1}\mathbf{p}_{i\ell}^{+}\beta_{i\ell}^{+}+O\left(\frac{1}{N_\ell}\right),\end{aligned} \tag{49}$$

where β_ℓ^0 and $\beta_{i\ell}^+$ are $d\times d$ block matrices of $\boldsymbol{\beta}_{k\ell}^+$ and the term $O(\frac{1}{N_\ell})$ means a matrix sequence, $\boldsymbol{\varphi}(N)\in\mathbb{R}^{N_\ell}$ such that $\lim_{N\to\infty}\frac{1}{N}\|\boldsymbol{\varphi}(N)\|^2=0$. A continuity argument then shows the empirical convergence (41). □

Lemma 5. *Under the induction hypothesis, the components of the 'random' term* $\mathbf{p}_{k\ell}^{+\mathrm{ran}}$ *along with the components of the vectors in* $\overline{\mathfrak{G}}_{k\ell}^{+}$ *almost surely converge empirically. The components of* $\mathbf{p}_{k\ell}^{+\mathrm{ran}}$ *converge as*

$$\mathbf{p}_{k\ell}^{+\mathrm{ran}}\overset{2}{\Rightarrow}U_{k\ell}, \tag{50}$$

where $U_{k\ell}$ *is a zero mean Gaussian random vector in* $\mathbb{R}^{1\times d}$ *independent of the limiting random variables corresponding to the variables in* $\overline{\mathfrak{G}}_{k\ell}^{+}$.

Proof. The proof is identical to that of lemmas 7 and 8 in [35]. □

We are now ready to prove lemma 3.

Proof of lemma 3. Using the partition (40a) and lemmas 4 and 5, we see that the components of the vector sequences in $\overline{\mathfrak{G}}_{k\ell}^{+}$ along with $\mathbf{p}_{k\ell}^{+}$ almost surely converge jointly empirically, where the components of $\mathbf{p}_{k\ell}^{+}$ have the limit

$$\mathbf{p}_{k\ell}^{+}=\mathbf{p}_{k\ell}^{\det}+\mathbf{p}_{k\ell}^{\mathrm{ran}}\overset{2}{\Rightarrow}P_{\ell}^{0}\beta_{\ell}^{0}+\sum_{i=0}^{k\text{-}1}P_{i\ell}^{+}\beta_{i\ell}^{+}+U_{k\ell}=:P_{k\ell}^{+}. \tag{51}$$

Note that the above Wasserstein-2 convergence can be shown using the same arguments involved in showing that if $X_N|\mathcal{F}\overset{d}{\Longrightarrow}X|\mathcal{F}$, and $Y_N|\mathcal{F}\overset{d}{\Longrightarrow}c$, then $(X_N,Y_N)|\mathcal{F}\overset{d}{\Longrightarrow}(X,c)|\mathcal{F}$ for some constant c and sigma-algebra $\mathcal{F}$.

We first establish the Gaussianity of $P_{k\ell}^{+}$. Observe that by the induction hypothesis, $\mathcal{H}_{k,\ell+1}^{-}$ holds whereby $(P_\ell^0,P_{0\ell}^+,\ldots,P_{k\text{-}1,\ell}^+,Q_{0,\ell+1}^-,\ldots,Q_{k,\ell+1}^-)$, is jointly Gaussian. Since U_k is Gaussian and independent of $(P_\ell^0,P_{0\ell}^+,\ldots,P_{k-1,\ell}^+,Q_{0,\ell+1}^-,\ldots,Q_{k,\ell+1}^-)$, we can conclude from (51) that $(P_\ell^0,P_{0\ell}^+,\ldots,P_{k\text{-}1,\ell}^+,P_{k\ell}^+,Q_{0,\ell+1}^-,\ldots,Q_{k,\ell+1}^-)$ is jointly Gaussian.

We now need to prove the correlations of this jointly Gaussian random vector are as claimed by $\mathcal{H}_{k,\ell+1}^{+}$. Since $\mathcal{H}_{k,\ell+1}^{-}$ is true, we know that (32) is true for all $i=0,\ldots,k-1$ and $j=0,\ldots,k$ and $\ell=\ell+1$. Hence, we need only to prove the additional identity for

J. Stat. Mech. (2021) 124004

$i = k$, namely the equations: $\mathrm{Cov}(P_\ell^0, P_{k\ell}^+)^2 = \mathbf{K}_{k\ell}^+$ and $\mathbb{E}(P_{k\ell}^+ Q_{j,\ell+1}^-) = 0$. First observe that

$$\mathbb{E}(P_{k\ell}^{+\mathsf{T}} P_{k\ell}^+)^2 \overset{(a)}{=} \lim_{N_\ell\to\infty} \frac{1}{N_\ell} \mathbf{p}_{k\ell}^{+\mathsf{T}} \mathbf{p}_{k\ell}^+ \overset{(b)}{=} \lim_{N_\ell\to\infty} \frac{1}{N_\ell} \mathbf{q}_{k\ell}^{+\mathsf{T}} \mathbf{q}_{k\ell}^+ \overset{(c)}{=} \mathbb{E}\big(Q_{k\ell}^{+\mathsf{T}} Q_{k\ell}^+\big)^2$$

where (a) follows from the fact that the rows of $\mathbf{p}_{k\ell}^+$ converge empirically to $P_{k\ell}^+$; (b) follows from line 20 in algorithm 3 and the fact that $\mathbf{V}_\ell$ is orthogonal; and (c) follows from the fact that the rows of $\mathbf{q}_{k\ell}^+$ converge empirically to $Q_{k\ell}^+$ from hypothesis $\mathcal{H}_{k,\ell}^+$. Since $\mathbf{p}_\ell^0 = \mathbf{V}_\ell \mathbf{q}^0$, we similarly obtain that $\mathbb{E}(P_\ell^{0\mathsf{T}} P_{k\ell}^+) = \mathbb{E}(Q_\ell^{0\mathsf{T}} Q_{k\ell}^+)$, $\mathbb{E}(P_\ell^{0\mathsf{T}} P_\ell^0) = \mathbb{E}(Q_\ell^{0\mathsf{T}} Q_\ell^0)$, from which we conclude

$$\mathrm{Cov}(P_\ell^0, P_{k\ell}^+) = \mathrm{Cov}(Q_\ell^0, Q_{k\ell}^+) =: \mathbf{K}_{k\ell}^+, \tag{52}$$

where the last step follows from the definition of $\mathbf{K}_{k\ell}^+$ in line 20 of algorithm 4. Finally, we observe that for $0 \leqslant j \leqslant k$

$$\mathbb{E}(P_{k\ell}^{+\mathsf{T}} Q_{j,\ell+1}^-) \overset{(a)}{=} \beta_\ell^{0\mathsf{T}} \mathbb{E}(P_\ell^{0\mathsf{T}} Q_{j,\ell+1}^-) + \sum_{i=0}^{k-1} \beta_{i\ell}^{+\mathsf{T}} \mathbb{E}(P_{i\ell}^{+\mathsf{T}} Q_{j,\ell+1}^-) + \mathbb{E}(U_{k\ell}^{\mathsf{T}} Q_{j,\ell+1}^-) \overset{(b)}{=} \mathbf{0}, \tag{53}$$

where (a) follows from (51) and, in (b), we used the fact that $\mathbb{E}(P_\ell^{0\mathsf{T}} Q_{j,\ell+1}^-) = \mathbf{0}$ and $\mathbb{E}(P_{i\ell}^{+\mathsf{T}} Q_{j,\ell+1}^-) = \mathbf{0}$ since (32) is true for $i \leqslant k-1$ corresponding to $\mathcal{H}_{k,\ell+1}^-$ and $\mathbb{E}(U_{k\ell}^{\mathsf{T}} Q_{j,\ell+1}^-) = \mathbf{0}$ since $U_{k\ell}$ is independent of $\overline{\mathfrak{G}}_{k\ell}^+$, and $Q_{j,\ell+1}^-$ is $\overline{\mathfrak{G}}_{k\ell}^+$ measurable. Thus, with (52) and (53), we have proven all the correlations in (32) corresponding to $\mathcal{H}_{k,\ell+1}^+$.

Next, we prove the convergence of the parameter lists $\Upsilon_{k,\ell+1}^+$ to $\overline{\Upsilon}_{k,\ell+1}^+$. Since $\Upsilon_{k\ell}^+ \to \overline{\Upsilon}_{k\ell}^+$ due to hypothesis $\mathcal{H}_{k\ell}^+$, and $\varphi_{k,\ell+1}^+(\cdot)$ is uniformly Lipschitz continuous, we have that $\lim_{N\to\infty} \mu_{k,\ell+1}^+$ from line 17 in algorithm 3 converges almost surely as

$$\begin{aligned}\lim_{N\to\infty} \big\langle \boldsymbol{\varphi}_{k,\ell+1}^+(\mathbf{p}_\ell^0, \mathbf{p}_{k\ell}^+, \mathbf{q}_{k,\ell+1}^-, \mathbf{w}_{\ell+1}, \overline{\Upsilon}_{k\ell}^+)\big\rangle &= \mathbb{E}\left[\varphi_{k,\ell+1}^+(P_\ell^0, P_{k\ell}^+, Q_{k,\ell+1}^-, W_{\ell+1}, \overline{\Upsilon}_{k\ell}^+)\right] \\ &= \overline{\mu}_{k,\ell+1}^+, \end{aligned} \tag{54}$$

where $\overline{\mu}_{k,\ell+1}^+$ is the value in line 17 in algorithm 4. Since $T_{k,\ell+1}^+(\cdot)$ is continuous, we have that $\lambda_{k,\ell+1}^+$ in line 18 in algorithm 3 converges as $\lim_{N\to\infty} \lambda_{k,\ell+1}^+ = T_{k,\ell+1}^+(\overline{\mu}_{k,\ell+1}^+, \overline{\Upsilon}_{k\ell}^+) =: \overline{\lambda}_{k,\ell+1}^+$, from line 18 in algorithm 4. Therefore, we have the limit

$$\lim_{N\to\infty} \Upsilon_{k,\ell+1}^+ = \lim_{N\to\infty} (\Upsilon_{k,\ell}^+, \lambda_{k,\ell+1}^+) = (\overline{\Upsilon}_{k,\ell}^+, \overline{\lambda}_{k,\ell+1}^+) = \overline{\Upsilon}_{k,\ell+1}^+, \tag{55}$$

which proves the convergence of the parameter lists stated in $\mathcal{H}_{k,\ell+1}^+$. Finally, using (55), the empirical convergence of the matrix sequences $\mathbf{p}_\ell^0$, $\mathbf{p}_{k\ell}^+$ and $\mathbf{q}_{k,\ell+1}^-$ and the uniform Lipschitz continuity of the update function $f_{k,\ell+1}^+(\cdot)$ we obtain that $\mathbf{q}_{k,\ell+1}^+$ equals

$$\mathbf{f}_{k,\ell+1}^+(\mathbf{p}_\ell^0, \mathbf{p}_{k\ell}^-, \mathbf{q}_{k,\ell+1}^-, \mathbf{w}_{\ell+1}, \Upsilon_{k,\ell+1}^+) \overset{2}{\Rightarrow} f_{k,\ell+1}^+(P_\ell^0, P_{k\ell}^-, Q_{k,\ell+1}^-, W_{\ell+1}, \overline{\Upsilon}_{k,\ell+1}^+) =: Q_{k,\ell+1}^+,$$

which proves the claim (33) for $\mathcal{H}_{k,\ell+1}^+$. This completes the proof. □

J. Stat. Mech. (2021) 124004

An overview of the iterates in algorithm 3 is depicted in figure 3 (top) and (middle). Theorem 2 shows that the rows of the iterates of algorithm 3 converge empirically with second order moments to random variables defined in algorithm 4. The random variables defined in algorithm 4 are depicted in figure 3 (bottom).

References

[1] Aubin B, Antoine M, Krzakala F, Macris N, Zdeborová L *et al* 2018 The committee machine: computational to statistical gaps in learning a two-layers neural network *Advances in Neural Information Processing Systems* pp 3223–34

[2] Barbier J, Krzakala F, Macris N, Miolane L and Zdeborová L 2019 Optimal errors and phase transitions in high-dimensional generalized linear models *Proc. Natl Acad. Sci. USA* **116** 5451–60

[3] Bayati M and Montanari A 2011 The dynamics of message passing on dense graphs, with applications to compressed sensing *IEEE Trans. Inf. Theory* **57** 764–85

[4] Bora A, Jalal A, Price E and Dimakis A G 2017 Compressed sensing using generative models *Proc. ICML*

[5] Byrne E, Chatalic A, Gribonval R and Schniter P 2019 Sketched clustering via hybrid approximate message passing *IEEE Trans. Signal Process.* **67** 4556–69

[6] Cakmak B, Winther O and Fleury B H 2014 S-AMP: approximate message passing for general matrix ensembles *Proc. IEEE ITW*

[7] Cheng X, Chatterji N S, Abbasi-Yadkori Y, Bartlett P L and Jordan M I 2018 Sharp convergence rates for Langevin dynamics in the nonconvex setting (arXiv:1805.01648)

[8] Cotter S F, Rao B D, Kjersti Engan K and Kreutz-Delgado K 2005 Sparse solutions to linear inverse problems with multiple measurement vectors *IEEE Trans. Signal Process.* **53** 2477–88

[9] Donoho D L, Maleki A and Montanari A 2009 Message-passing algorithms for compressed sensing *Proc. Natl Acad. Sci.* **106** 18914–9

[10] Donoho D L, Maleki A and Montanari A 2010 Message passing algorithms for compressed sensing *Proc. of IEEE Information Theory Workshop* pp 1–5

[11] Emami M, Sahraee-Ardakan M, Pandit P, Rangan S and Fletcher A K 2020 Generalization error of generalized linear models in high dimensions (arXiv:2005.00180)

[12] Fletcher A K, Rangan S and Schniter P 2018 Inference in deep networks in high dimensions *Proc. of IEEE Int. Symp. on Information Theory*

[13] Fletcher A K, Sahraee-Ardakan M, Rangan S and Schniter P 2016 Expectation consistent approximate inference: generalizations and convergence *Proc. of IEEE Int. Symp. on Information Theory* pp 190–4

[14] Gabrié M, Manoel A, Luneau C, Barbier J, Macris N, Krzakala F and Zdeborová L 2018 Entropy and mutual information in models of deep neural networks *Proc. NIPS*

[15] Hand P and Voroninski V 2017 Global guarantees for enforcing deep generative priors by empirical risk (arXiv:1705.07576)

[16] He H, Wen C-K and Jin S 2017 Generalized expectation consistent signal recovery for nonlinear measurements *2017 IEEE Int. Symp. on Information Theory (ISIT)* (IEEE) pp 2333–7

[17] Kabashima Y 2003 A CDMA multiuser detection algorithm on the basis of belief propagation *J. Phys. A: Math. Gen.* **36** 11111

[18] Kabkab M, Samangouei P and Chellappa R 2018 Task-aware compressed sensing with generative adversarial networks *32nd AAAI Conf. on Artificial Intelligence*

[19] Keriven N, Bourrier A, Gribonval R and Pérez P 2017 Sketching for large-scale learning of mixture models *Inf. Inference* A **7** 447–508

[20] Keriven N, Tremblay N, Traonmilin Y and Gribonval R 2017 Compressive k-means *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE) pp 6369–73

[21] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)

[22] Liang D, Ying L and Liang F 2009 Parallel MRI acceleration using M-FOCUSS *Proc. of Int. Conf. on Bioinformatics and Biomedical Engineering* (IEEE) pp 1–4

[23] Liu J S 1994 Siegel's formula via Stein's identities *Stat. Probab. Lett.* **21** 247–51

[24] Ma J and Ping L 2017 Orthogonal AMP *IEEE Access* **5** 2020–33

[25] Manoel A, Krzakala F, Mézard M and Zdeborová L 2017 Multi-layer generalized linear estimation *Proc. of IEEE Int. Symp. on Information Theory* pp 2098–102

[26] Manoel A, Krzakala F, Varoquaux G, Thirion B and Zdeborová L 2018 Approximate message-passing for convex optimization with non-separable penalties (arXiv:1809.06304)
[27] Mei S, Montanari A and Nguyen P-M 2018 A mean field view of the landscape of two-layer neural networks *Proc. Natl Acad. Sci. USA* **115** E7665–71
[28] Minka T P 2001 Expectation propagation for approximate Bayesian inference *Proc. UAI* pp 362–9
[29] Mixon D G and Villar S 2018 Sunlayer: stable denoising with generative networks (arXiv:1803.09319)
[30] Montanari A, Ruan F, Sohn Y and Yan J 2019 The generalization error of max-margin linear classifiers: high-dimensional asymptotics in the overparametrized regime (arXiv:1911.01544)
[31] Obozinski G, Taskar B and Jordan M 2006 Multi-task feature selection *Technical Report* Statistics Department, UC Berkeley p 2
[32] Opper M and Winther O 2005 Expectation consistent approximate inference *J. Mach. Learn. Res.* **6** 2177–204
[33] Pandit P, Sahraee M, Rangan S and Fletcher A K 2019 Asymptotics of MAP inference in deep networks *Proc. of IEEE Int. Symp. on Information Theory* pp 842–6
[34] Pandit P, Sahraee-Ardakan M, Rangan S, Schniter P and Fletcher A K 2020 Inference with deep generative priors in high dimensions *IEEE J. Sel. Areas Inf. Theory* **1** 336
[35] Rangan S, Schniter P and Fletcher A K 2019 Vector approximate message passing *IEEE Trans. Inf. Theory* **65** 6664–84
[36] Reeves G 2017 Additivity of information in multilayer networks via additive Gaussian noise transforms *Proc. of Allerton Conf. on Communication, Control & Computing* pp 1064–70
[37] Shah V and Hegde C 2018 Solving linear inverse problems using GAN priors: an algorithm with provable guarantees *IEEE Int. Conf. on Acoustics, Speech and Signal Processing* pp 4609–13
[38] Takeuchi K 2017 Rigorous dynamics of expectation-propagation-based signal recovery from unitarily invariant measurements *Proc. of IEEE Int. Symp. on Information Theory* pp 501–5
[39] Themelis A and Patrinos P 2020 Douglas–Rachford splitting and ADMM for nonconvex optimization: tight convergence results *SIAM J. Optim.* **30** 149–81
[40] Tresp V 2000 A Bayesian committee machine *Neural Comput.* **12** 2719–41
[41] Tripathi S, Lipton Z C and Nguyen T Q 2018 Correction by projection: denoising images with generative adversarial networks (arXiv:1803.04477)
[42] Tzagkarakis G, Milioris D and Tsakalides P 2010 Multiple-measurement Bayesian compressed sensing using GSM priors for DOA estimation *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing* (IEEE) pp 2610–3
[43] Villani C 2008 *Optimal Transport: Old and New* vol 338 (Berlin: Springer) (https://doi.org/10.1007/978-3-540-71050-9)
[44] Welling M and Yee W T 2011 Bayesian learning via stochastic gradient Langevin dynamics *Proc. of 28th Int. Conf. on Machine Learning* pp 681–8
[45] Yeh R, Chen C, Lim T Y, Hasegawa-Johnson M and Do M N 2016 Semantic image inpainting with perceptual and contextual losses (arXiv:1607.07539)
[46] Yi X, Caramanis C and Sanghavi S 2014 Alternating minimization for mixed linear regression *Int. Conf. on Machine Learning* pp 613–21
[47] Ziniel J and Schniter P 2013 Efficient high-dimensional inference in the multiple measurement vector problem *IEEE Trans. Signal Process.* **61** 340–54

J. Stat. Mech. (2021) 124004