Multi-Layer Generalized Linear Estimation

Andre Manoel, Florent Krzakala, Marc Mézard, Lenka Zdeborová

Multi-Layer Generalized Linear Estimation

Andre Manoel
Neurospin, CEA
Université Paris-Saclay

Florent Krzakala
LPS ENS, CNRS
PSL, UPMC & Sorbonne Univ.

Marc Mézard
Ecole Normale Supérieure
PSL Research University

Lenka Zdeborová
IPhT, CNRS, CEA
Université Paris-Saclay

***Abstract*—We consider the problem of reconstructing a signal from multi-layered (possibly) non-linear measurements. Using non-rigorous but standard methods from statistical physics we present the Multi-Layer Approximate Message Passing (ML-AMP) algorithm for computing marginal probabilities of the corresponding estimation problem and derive the associated state evolution equations to analyze its performance. We also give the expression of the asymptotic free energy and the minimal information-theoretically achievable reconstruction error. Finally, we present some applications of this measurement model for compressed sensing and perceptron learning with structured matrices/patterns, and for a simple model of estimation of latent variables in an auto-encoder.**

In many natural and engineered systems, the interactions between sets of variables in different subsystems involve multiple layers of interdependencies. This is for instance the case in the neural networks developed in deep learning [1], the hierarchical models used in statistical inference [2], and the multiplex networks considered in complex systems [3]. It is therefore fundamental to generalize our theoretical and algorithmic tools to deal with these *multi-layer* setups. Our goal in this paper is to develop such a generalization of the cavity/replica approach that originated in statistical physics [4] and that has been shown to be quite successful for studying generalized linear estimation with randomly chosen mixing, leading in particular to the computation of the mutual information (or equivalently free energy) and minimum achievable mean-squared error for CDMA and compressed sensing [5]–[8]. This methodology is also closely related to the approximate message passing (AMP) algorithm, originally known in physics as Thouless-Anderson-Palmer (TAP) equations [9]–[14].

We present in section I a multi-layer generalized linear measurement (ML-GLM) model with random weights at each layer and consider the Bayesian inference of the signal measured by the ML-GLM. We derive AMP for ML-GLM and, using non-rigorous but standard methods from statistical physics, analyze its behavior by deriving the associated state evolution. We also present the expression for the associated free energy (or mutual information) and the optimal information-theoretically mean squared error (MMSE). We compare the MMSE with the MSE achieved by AMP and describe the associated phase transitions.

I. Problem statement

ML-GLM model: Consider L known matrices $W^{(1)}$, $W^{(2)}$, $\dots$, $W^{(L)}$ of dimension $W^{(\ell)} \in \mathbb{R}^{n_{\ell-1}\times n_\ell}$. A control parameter which will be important is the ratio of the number of rows to columns in each of these matrices, $\alpha_\ell = n_{\ell-1}/n_\ell$. The components of each of these matrices are drawn independently at random, from a probability distribution $P_{W^{(\ell)}}$ having zero mean and variance $1/n_\ell$. We consider a signal $\boldsymbol{x}\in\mathbb{R}^{n_L}$ with elements x_i, $i = 1,\dots,n_L$ sampled independently from a distribution $P_X(x_i)$. We then collect n_0 observations $\boldsymbol{y} \in \mathbb{R}^{n_0}$ of the signal $\boldsymbol{x}$ as

$$\boldsymbol{y} = f^{(1)}_{\xi^1}(W^{(1)} f^{(2)}_{\xi^2}(W^{(2)} \cdots f^{(L)}_{\xi^L}(W^{(L)}\boldsymbol{x}))), \tag{1}$$

where the so-called *activation functions* $f^{(\ell)}_{\xi^\ell}$, $\ell = 1,\dots,L$, are applied element-wise. These functions can be deterministic or stochastic and are, in general, non-linear. Assuming $f^{(\ell)}_{\xi}(z)$ depends on a variable z and some noise ξ distributed with $P^{(\ell)}(\xi)$, we can define the probability distribution of the output of the function $h = f^{(\ell)}_{\xi}(z)$ as

$$P^{(\ell)}_{\text{out}}(h|z) = \int \mathrm{d}\xi\, P^{(\ell)}(\xi)\, \delta(h - f^{(\ell)}_{\xi}(z))\,, \tag{2}$$

where $\delta(.)$ is the Dirac function. P_{out} is then interpreted as a noisy channel through which the variable z is observed, h being the observation. With the above definition of P_{out} we can rewrite eq. (1) in an equivalent form introducing *hidden* auxiliary variables $h^{(\ell)} \in \mathbb{R}^{n_\ell}$ for $\ell = 1,\dots,L-1$ as

$$\begin{aligned} y_\mu &\sim P^{(1)}_{\text{out}}\Big(y_\mu\Big|\textstyle\sum_{i=1}^{n_1} W^{(1)}_{\mu i} h^{(1)}_i\Big), \\ h^{(1)}_\mu &\sim P^{(2)}_{\text{out}}\Big(h^{(1)}_\mu\Big|\textstyle\sum_{i=1}^{n_2} W^{(2)}_{\mu i} h^{(2)}_i\Big), \\ &\vdots \\ h^{(L-1)}_\mu &\sim P^{(L)}_{\text{out}}\Big(h^{(L-1)}_\mu\Big|\textstyle\sum_{i=1}^{n_L} W^{(L)}_{\mu i} x_i\Big), \\ x_\mu &\sim P_X(x_\mu), \end{aligned} \tag{3}$$

The inference problem of interest in this paper is the MMSE estimation of the signal $\boldsymbol{x}$ from the knowledge of the observation $\boldsymbol{y}$ and the matrices $W^{(\ell)}$, for all $\ell = 1,\dots,L$. This inference is done through the computation of marginals of the corresponding posterior distribution $P(\boldsymbol{x}|\boldsymbol{y})$.

Using the Bayes theorem and the above definition of the hidden variables $h^{(\ell)} \in \mathbb{R}^{n_l}$, the posterior is written as

$$\begin{aligned} P(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{Z(\boldsymbol{y})} \prod_{\mu=1}^{n_L} P_X(x_\mu) \int \prod_{\ell=1}^{L-1}\prod_{\mu=1}^{n_\ell} \mathrm{d}h^{(\ell)}_\mu \\ \prod_{\mu=1}^{n_1} P^{(1)}_{\text{out}}\Big(y_\mu\Big|\sum_{i=1}^{n_1} W^{(1)}_{\mu i} h^{(1)}_i\Big) \prod_{\mu=1}^{n_L} P^{(L)}_{\text{out}}\Big(h^{(L-1)}_\mu\Big|\sum_{i=1}^{n_L} W^{(L)}_{\mu i} x_i\Big) \\ \prod_{\ell=2}^{L-1}\prod_{\mu=1}^{n_\ell} P^{(\ell)}_{\text{out}}\Big(h^{(\ell-1)}_\mu\Big|\sum_{i=1}^{n_\ell} W^{(\ell)}_{\mu i} h^{(\ell)}_i\Big). \end{aligned} \tag{4}$$

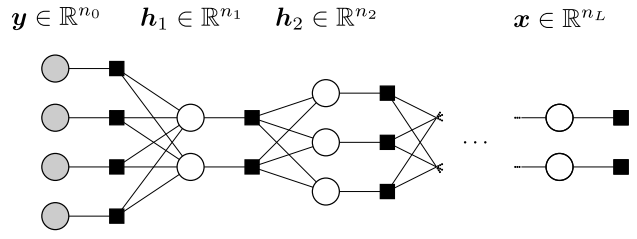

Fig. 1. Factor graph of the multi-layer generalized linear estimation problem (4). Shaded circles correspond to observations $\boldsymbol{y}$, empty circles to hidden variables $\boldsymbol{h}$ and the signal $\boldsymbol{x}$ to be inferred. Squares represent the activation functions relating the variables via eqs. (3).

We focus here on the "Bayes-optimal inference" where the generative model and all its parameters are known, i.e. the only unknown variables are $\boldsymbol{x}$ and the $\boldsymbol{h}^{(\ell)}$, $\ell = 1, \dots, L-1$. In this case, in order to minimize the expected mean-squared error between the ground truth value of $\boldsymbol{x}$ and its estimator $\hat{\boldsymbol{x}}$ one needs to compute averages of the marginals of the posterior distribution.

As usual, computing the marginals of the posterior (4) is in general intractable. In this paper we develop an analysis of the posterior and its marginals that is asymptotically exact, in the sense that, with high probability, the estimated marginal probabilities differ from the true ones by a factor that goes to zero in the "thermodynamic" limit $n_\ell \to \infty$ for every $\ell = 0, \dots, L$ (at fixed ratios $\alpha_\ell = n_{\ell-1}/n_\ell = O(1)$). Our analysis is based on an AMP-type algorithm and an analysis of the corresponding Bethe/replica free energy.

Context: The ML-GLM model considered in this paper has a range of applications and is related to other models considered in the literature. It is similar to the deep exponential families of [2], [15] with fixed known random weights. It can also be seen as the decoder-side of an autoencoder neural network with fixed known weights (corresponding to a randomly generated hierarchy of features), the goal being to infer the vector of latent variables that is the closest to some ground-truth values used to generate the data. One of the main assets of the considered ML-GLM is that the activation functions in each of the layers can be non-linear, as is the case in deep neural networks, and our analysis handles these non-linearities. When the intermediate layers are linear, the ML-GLM can be interpreted as a model for compressed sensing with a structured measurement matrix obtained as a product of random matrices. Similarly, when the external layer has a threshold activation function and all the other activation functions are linear, the ML-GLM can be seen as a single layer perceptron that aims to classify correlated patterns obtained by products of random matrices.

We anticipate that the algorithm and theory developed in this paper will find applications to other learning problems. A natural direction of future work is to prove both the state evolution of the algorithm and the Bethe free energy for the ML-GLM rigorously, perhaps along the lines of [7], [8], [16].

II. Algorithm and analysis

ML-AMP: We consider a probability distribution $P(\boldsymbol{x}, \{\boldsymbol{h}^{(\ell)}\}_{\ell=1}^{L-1} | \boldsymbol{y})$ defined as the posterior distribution (4) without the integral over the auxiliary variables $\{\boldsymbol{h}^{(\ell)}\}_{\ell=1}^{L-1}$, and represented by the graphical model depicted in Fig. 1. We derive ML-AMP by first writing the belief propagation equations [17] for this graphical model. As every factor relates to many variables and every variable is contained in many factors, we use the central limit theorem to keep track of only the means and variances of the messages. Furthermore, we express the iterations in terms of node-variables instead of messages, giving rise to the so called Onsager reaction terms. This derivation was presented for the case of single-layer generalized linear estimation in e.g. [13], [14]. The resulting multi-layer-AMP (ML-AMP) algorithm then iteratively updates estimators of the means $\hat{\boldsymbol{h}}^{(\ell)}$ and of the variances $\boldsymbol{\sigma}^{(\ell)}, \ell = 1, \dots, L$ of the auxiliary variables $\boldsymbol{h}^{(\ell)}, \ell = 1, \dots, L-1$ and of the signal $\boldsymbol{x}$. For simplicity of the multi-level notation we denote $\hat{\boldsymbol{x}} = \hat{\boldsymbol{h}}^{(L)}$, and similarly for the variance.

We first write the ML-AMP update equations for an intermediate layer ℓ and then specify how they change for the very first ($\ell = 1$) and the very last ($\ell = L$) layers. At each layer $1 \geq \ell \geq L$, there are two vectors, $\boldsymbol{V}^{(\ell)} \in \mathbb{R}^{n_{\ell-1}}$ and $\boldsymbol{\omega}^{(\ell)} \in \mathbb{R}^{n_{\ell-1}}$, associated with the factor nodes, and two vectors, $\boldsymbol{A}^{(\ell)} \in \mathbb{R}^{n_l}$ and $\boldsymbol{B}^{(\ell)} \in \mathbb{R}^{n_l}$, associated with the variable nodes. Their update reads

$$\begin{aligned}
V_\mu^{(\ell)}(t) &= \sum_i \left[W_{\mu i}^{(\ell)}\right]^2 \sigma_i^{(\ell)}(t), \\
\omega_\mu^{(\ell)}(t) &= \sum_i W_{\mu i}^{(\ell)} \hat{h}_i^{(\ell)}(t) - V_\mu^{(\ell)}(t) g_\mu^{(\ell)}(t-1), \\
A_i^{(\ell)}(t) &= -\sum_\mu \left[W_{\mu i}^{(\ell)}\right]^2 \partial_\omega g_\mu^{(\ell)}(t), \\
B_i^{(\ell)}(t) &= \sum_\mu W_{\mu i}^{(\ell)} g_\mu^{(\ell)}(t) + A_i^{(\ell)}(t) \hat{h}_i^{(\ell)}(t).
\end{aligned} \tag{5}$$

To define functions $g_\mu^{(\ell)}(t)$, $\partial_\omega g_\mu^{(\ell)}(t)$ and to state how to use eqs. (5) to update the estimators $\hat{h}_i^{(\ell)}(t)$ and variances $\sigma_i^{(\ell)}(t)$, we need to define an auxiliary function $\mathcal{Z}^{(\ell)}$ for $2 \geq l \geq L$ as

$$\mathcal{Z}^{(\ell)}(A^{(\ell-1)}, B^{(\ell-1)}, V^{(\ell)}, \omega^{(\ell)}) \equiv \frac{1}{\sqrt{2\pi V^{(\ell)}}} \int \mathrm{d}h \, \mathrm{d}z \, P_{\text{out}}^{(\ell)}(h|z) \, e^{-\frac{1}{2}A^{(\ell-1)}h^2 + B^{(\ell-1)}h} \, e^{-\frac{(z-\omega^{(\ell)})^2}{2V^{(\ell)}}}.$$

With this definition, the estimators of marginal mean of the auxiliary variables $\hat{h}_i^{(\ell)}, 1 \geq \ell \geq L-1$, and the quantity $g_\mu^{(\ell)}, 2 \geq \ell \geq L$, is computed as

$$\begin{aligned}
g_\mu^{(\ell)}(t) &= \partial_\omega \log \mathcal{Z}^{(\ell)}(A_\mu^{(\ell-1)}, B_\mu^{(\ell-1)}, V_\mu^{(\ell)}, \omega_\mu^{(\ell)}), \\
\hat{h}_i^{(\ell)}(t+1) &= \partial_B \log \mathcal{Z}^{(\ell+1)}(A_i^{(\ell)}, B_i^{(\ell)}, V_i^{(\ell+1)}, \omega_i^{(\ell+1)}),
\end{aligned} \tag{6}$$

where the quantities on the right hand side of (6) are evaluated at time index t.

The output function $g_\mu^{(1)}$ in the first layer is obtained as in the standard AMP algorithm for generalized linear estimation:

$$\mathcal{Z}^{(1)}(y, V^{(1)}, \omega^{(1)}) = \int \mathrm{d}z \, P_{\text{out}}^{(1)}(y|z) \, \frac{e^{-\frac{(z-\omega^{(1)})^2}{2V^{(1)}}}}{\sqrt{2\pi V^{(1)}}}, \tag{7}$$

procedure ML-AMP
  initialize $g_\mu^{(\ell)} = 0 \; \forall(\mu,\ell)$
  initialize $h_i^{(\ell)} = 0, \sigma_i^{(\ell)} = 1 \; \forall(i,\ell)$
  for $t \leftarrow 1$ **to** $t_{\max}$ **do**
    for $\ell \leftarrow 1$ **to** L **do**
      compute $V_\mu^{(\ell)}, \omega_\mu^{(\ell)} \; \forall\mu$ using (5)
      compute $g_\mu^{(\ell)}, \partial_\omega g_\mu^{(\ell)} \; \forall\mu$ using (6)
      compute $A_i^{(\ell)}, B_i^{(\ell)} \; \forall i$ using (5)
    end for
    compute $\hat{h}_i^{(\ell)}, \sigma_i^{(\ell)} \; \forall(i,\ell)$ using (6)
  end for
end procedure

$$g_\mu^{(1)}(t) = \partial_\omega \log \mathcal{Z}^{(1)}(y_\mu, V_\mu^{(1)}(t), \omega_\mu^{(1)}(t)) \; . \tag{8}$$

In the last layer, the estimator $\hat{h}_i^{(L)} = \hat{x}_i$ is obtained from

$$\begin{aligned} &\mathcal{Z}^{(L+1)}(A^{(L)}, B^{(L)}) = \int \mathrm{d}h \, P_X(h) \, e^{-\frac{1}{2}A^{(L)}h^2 + B^{(L)}h}, \\ &\hat{h}_i^{(L)}(t+1) = \partial_B \log \mathcal{Z}^{(L+1)}(A_i^{(L)}(t), B_i^{(L)}(t)). \end{aligned} \tag{9}$$

Finally, the expressions $\partial_\omega g_\mu^{(\ell)}$ and $\sigma_i^{(\ell)}$ in (5) are defined as: $\partial_\omega g_\mu^{(\ell)}(t) = \partial_\omega^2 \log \mathcal{Z}^{(\ell)}(t)$ and $\sigma_i^{(\ell)}(t+1) = \partial_B^2 \log \mathcal{Z}^{(\ell+1)}(t)$.

Note that the ML-AMP is closely related to AMP for generalized linear estimation [13]. The form of ML-AMP is the one we would obtain if we treated each layer separately, while defining an effective prior P_X^{eff} depending on the variables of the next layer, and an effective output channel $P_{\text{out}}^{\text{eff}}$ depending on the variables of the preceding layer:

$$\begin{aligned} P_X^{\text{eff}}(h^{(\ell)} | V^{(\ell+1)}, \omega^{(\ell+1)}) &= \int \mathrm{d}z \, P_{\text{out}}^{(\ell+1)}(h^{(\ell)}|z) \frac{e^{-\frac{(z-\omega^{(\ell+1)})^2}{2V^{(\ell+1)}}}}{\sqrt{2\pi V^{(\ell+1)}}}, \\ P_{\text{out}}^{\text{eff}}(z^{(\ell)} | A^{(\ell-1)}, B^{(\ell-1)}) &= \int \mathrm{d}h \, P_{\text{out}}^{(\ell)}(h|z^{(\ell)}) \, e^{-\frac{1}{2}A^{(\ell-1)}h^2 + B^{(\ell-1)}h}. \end{aligned}$$

State evolution: A very useful property of AMP, compared to other algorithms commonly used for estimating marginals of posterior distributions such as (4), is that its performance can be traced analytically in the thermodynamic limit using the so-called *state evolution* (SE) [13], [16]. This is a version of the density evolution [4] for dense graphs. It is known as the cavity method in statistical physics [4], where it is in general non-rigorous.

Specifically, the cavity method implies that at each layer ℓ, the overlap between the ground true $\boldsymbol{h}^{(\ell)}$ and its estimate $\hat{\boldsymbol{h}}^{(\ell)}(t)$ provided by the ML-AMP algorithm at iteration t concentrates around an "overlap" $m^{(\ell)}(t)$

$$m^{(\ell)}(t) \overset{n_\ell \to \infty}{=} \frac{1}{n_\ell} \sum_{i=1}^{n_\ell} h_i^{(\ell)} \hat{h}_i^{(\ell)}(t) \, . \tag{10}$$

In the case of the Bayes-optimal inference of $\boldsymbol{x}$, the state evolution for the ML-AMP algorithm is written also in terms of a parameter $\hat{m}^{(\ell)}$, defined as the value around which the components of $\boldsymbol{A}^{(\ell)}$ concentrate.

For the intermediate layers, $2 \geq \ell \geq L$, the SE reads

$$\begin{aligned} \hat{m}^{(\ell)}(t) &= -\alpha_\ell \, \mathbb{E}_{h,z,b,w}^{(\ell)} \partial_\omega g^{(\ell)}\big(\hat{m}^{(\ell-1)}(t), b, \rho_\ell - m^{(\ell)}(t), w\big), \\ m^{(\ell-1)}(t+1) &= \mathbb{E}_{h,z,b,w}^{(\ell)} h \, \hat{h}^{(\ell-1)}(\hat{m}^{(\ell-1)}(t), b, \rho_\ell - m^{(\ell)}(t), w), \end{aligned} \tag{11}$$

where the scalar functions $\partial_\omega g^{(\ell)}$ and $\hat{h}^{(\ell)}$ are defined in eq. (6). The quantity ρ_ℓ is given by the second moment of the components of the vector $\boldsymbol{h}^{(\ell)}$. It can be computed from the knowledge of P_X and $P_{\text{out}}^{(k)}$ for $k = \ell, \dots, L$. The expectations are taken over the joint distribution

$$\begin{aligned} P^{(\ell)}(h,z,b,w) = P_{\text{out}}^{(\ell)}(h|z) \mathcal{N}(z; w, \rho_\ell - m^{(\ell)}) \times \\ \mathcal{N}(b; \hat{m}^{(\ell-1)} h, \hat{m}^{(\ell-1)}) \mathcal{N}(w; 0, m^{(\ell)}). \end{aligned} \tag{12}$$

In the above distribution, h, z, b, w are random variables representing respectively: the ground truth hidden variables at layer $(\ell-1)$, the components of the estimator of $W^{(\ell)}\boldsymbol{h}^{(\ell)}$, the components of $\boldsymbol{B}^{(\ell-1)}$, and the components of $\boldsymbol{\omega}^{(\ell)}$. Note that the probability distribution (12) is similar to the one appearing in the single-layer G-AMP algorithm of [13] with the exception that, in $P_{\text{out}}^{(\ell)}(h|z)$, a known measurement is replaced by the distribution of the hidden variable $h^{(\ell-1)}$ at the previous layer. This makes the multi-layer SE quite intuitive for readers well familiar with the SE for G-AMP.

At the first (leftmost) and last (rightmost) layers, the SE order parameters are given by the same fixed point equations as in the single-layer G-AMP setting, that is

$$\begin{aligned} \hat{m}^{(1)}(t) &= -\alpha_1 \, \mathbb{E}_{y,z,w} \, \partial_\omega g^{(1)}(y, \rho_1 - m^{(1)}(t), w), \\ m^{(L)}(t+1) &= \mathbb{E}_{x,b} \, x \, \hat{h}^{(L)}(\hat{m}^{(L)}(t), b), \end{aligned} \tag{13}$$

where expectations are taken over $P(y,z,w) = P_{\text{out}}^{(1)}(y|z) \mathcal{N}(w; 0, m^{(1)}) \mathcal{N}(z; w, \rho_1 - m^{(1)})$ and $P(x,b) = P_X(x) \mathcal{N}(b; \hat{m}^{(L)} x, \hat{m}^{(L)})$ respectively. Thus, if $L = 1$, the state evolution equations of ML-AMP reduce to that of the standard G-AMP algorithm.

The state evolution are iterative equations. We initialize $m^{(\ell)}(t=0)$ close to zero (or otherwise, corresponding to the initialization of the ML-AMP algorithm), then compute $\hat{m}^{(\ell)}(t=0)$ for all layers. We then compute m for the next time step for all layers, then $\hat{m}$ at the same time step and all the layers, etc. Finally to obtain the mean-squared error on $h^{(\ell)}$ from the state evolution, we evaluate $\text{MSE}^{(\ell)} = \rho_\ell - m^{(\ell)}$.

III. Free energy and phase transitions

We define the free energy of the posterior (4) as

$$\phi = -\lim_{\{n_\ell\} \to \infty} \frac{1}{n_L} \log Z(\boldsymbol{y}) \; , \tag{14}$$

where $\lim_{\{n_\ell\} \to \infty}$ denotes the thermodynamic limit. This free energy is self-averaging, meaning that the above limit is with high probability independent of the realization of the random variable $\boldsymbol{y}$, it only depends on the parameters of the model α_ℓ, L, P_X, and $P_{\text{out}}^{(\ell)}$. A computation analogous to the one that leads from belief propagation to the ML-AMP algorithm and its SE can be used to rewrite the Bethe free energy [17] into a single instance free energy evaluated using the fixed points of the ML-AMP algorithm. Using averaging analogous to the one of state evolution one then rewrites this into the

so-called replica symmetric free energy

$$\phi_{\rm RS}(\boldsymbol{m},\hat{\boldsymbol{m}}) = \frac{1}{2}\sum_{\ell=1}^{L}\tilde{\alpha}_\ell m^{(\ell)}\hat{m}^{(\ell)} - \tilde{\alpha}_L\mathcal{I}^{(L+1)}(\hat{m}^{(L)}) \\ -\sum_{\ell=2}^{L}\tilde{\alpha}_{\ell-1}\mathcal{I}^{(\ell)}(m^{(\ell)},\hat{m}^{(\ell-1)}) - \tilde{\alpha}_0\mathcal{I}^{(1)}(m^{(1)}). \tag{15}$$

with $\tilde{\alpha}_\ell = n_\ell/n_L$, and

$$\mathcal{I}^{(L+1)}(\hat{m}^{(L)}) = \mathbb{E}_{x,b}\,\log\mathcal{Z}^{(L+1)}(\hat{m}^{(L)},b),$$
$$\mathcal{I}^{(\ell)}(m^{(\ell)},\hat{m}^{(\ell-1)}) = \mathbb{E}^{(\ell)}_{h,z,b,w}\log\mathcal{Z}^{(\ell)}(\hat{m}^{(\ell-1)},b,\rho_\ell - m^{(\ell)},w),$$
$$\mathcal{I}^{(1)}(m^{(1)}) = \mathbb{E}_{y,z,w}\,\log\mathcal{Z}^{(1)}(y,\rho_1 - m^{(1)},w).$$

One can check, by computing derivatives of this free energy with respect to m_ℓ and $\hat{m}_\ell$, that the stationary points of $\phi_{\rm RS}(\boldsymbol{m},\hat{\boldsymbol{m}})$ are fixed points of the state evolution equations (11) and (13). Let us now use (11) and (13) to express $\hat{\boldsymbol{m}}$ in terms of $\boldsymbol{m}$ and consider the free energy $\phi_{\rm RS}(\boldsymbol{m})$ only as a function of the overlaps $\boldsymbol{m}$ (10). Define

$$\phi_{\rm RS} \equiv \min_{\boldsymbol{m}}\,\phi_{\rm RS}(\boldsymbol{m}), \\ \boldsymbol{m}_{\rm IT} \equiv \operatorname*{argmin}_{\boldsymbol{m}}\,\phi_{\rm RS}(\boldsymbol{m}). \tag{16}$$

In the setting of Bayes-optimal inference, the replica symmetric free energy $\phi_{\rm RS}$ is equal to the free energy (14). At the same time the minimum mean squared error of the Bayes-optimal estimation of $\boldsymbol{x}$ is given by

$$\text{MMSE} = \rho_L - m^{(L)}_{\rm IT}\,, \tag{17}$$

where $\boldsymbol{m}_{\rm IT}$ is defined in (16) and ρ_L is the second moment of P_X. This has been recently proven for the single layer linear estimation [7], [8], and based on statistical physics arguments we conjecture it to be true also in the present problem.

We divide the region of parameters of the present problem into three phases. If the MMSE is not low enough (defined in a way depending on the application) we say that inference of the signal $\boldsymbol{x}$ is information-theoretically *impossible*. If the MMSE is sufficiently low and the ML-AMP algorithm analyzed via state evolution matches it, then we say inference is *easy*. Finally, and most interestingly, if MMSE is sufficiently low, but ML-AMP achieves worse MSE we talk about a region of algorithmically *hard* inference. Defining an algorithmically hard region by the performance of one given algorithm might seem in general unjustified. However, in the case of single layer generalized linear estimation we know of no other polynomial algorithm that would outperform AMP in the thermodynamic limit for random W. This leads us to the above definition of the hard phase also for the present multi-layer problem.

IV. Results for selected problems

We now focus on three examples of two-layer models and draw their phase diagrams that indicate for which layer sizes (i.e. for which values of α_ℓ) reconstruction of the signal is information-theoretically possible. We also run the ML-AMP on single instances sampled from the model and illustrate that the mean-squared error it reaches after convergence matches the one predicted by the state evolution.

Sparse linear regression using correlated data: Among the simplest non-trivial cases of the ML-GLM model is a two-layer analogue of sparse linear regression (SLR) defined as

$$\boldsymbol{y} = W^{(1)}(W^{(2)}\boldsymbol{x} + \mathcal{N}(0,\Delta_2)) + \mathcal{N}(0,\Delta_1). \tag{18}$$

where the vector $\boldsymbol{x}$ we seek to infer is sparse, $P_X(\boldsymbol{x}) = \prod_{i=1}^{n_2}[\rho\,\mathcal{N}(x_i;0,1) + (1-\rho)\,\delta(x_i)]$, and $W^{(\ell)}_{\mu i} \sim \mathcal{N}(0,1/n_\ell)$.

When $\Delta_2 = 0$ the model (18) is equivalent to a SLR with a structured data matrix $\Phi = W^{(1)}W^{(2)}$, a problem previously studied by [18]–[22]. Interestingly, the state evolution equations (11) have at their fixed point $\hat{m}^{(2)} = R\left(-(\rho_2 - m^{(2)})/\Delta_1\right)/\Delta_1$, where R gives the R-transform of $\Phi^T\Phi$ (see e.g. [23]). Together with eq. (13) these are the same equations obtained by adaptive approaches in [18]–[22]. This confirms that the adaptative methods are exact in the case of matrix product, as was already noted for the Hopfield model [24].

In Fig. 2 (left) we draw the phase diagram as a function of $\alpha \equiv \alpha_1\alpha_2$ and α_2. Remarkably, in the noiseless case, the phase diagram of the problem with structured matrix Φ is reduced to the statement that both α and α_2 need to be larger than the corresponding threshold known for the usual compressed sensing problem [14].

For $\Delta_2 > 0$ this simple mapping does not hold and, as far as we know, the ML-AMP and its SE analysis give new results. We compared numerically the performance of ML-AMP to the VAMP of [22]. Whereas for $\Delta_2 = 0$ the two algorithm agree (within errorbars), for $\Delta_2 > 0$ the ML-AMP gives a distinguishably lower MSE.

Perceptron learning with correlated patterns: A lot of work has been dedicated to learning and generalization in a perceptron with binary weights [25], [26], defined by:

$$\boldsymbol{y} = \text{sgn}(\Phi\boldsymbol{x}) \tag{19}$$

with $\boldsymbol{x} \sim \prod_{i=1}^{n_2}[\frac{1}{2}\delta(x_i-1) + \frac{1}{2}\delta(x_i+1)]$. These works focused on random patterns where the elements of Φ are iid.

It was recently argued that learning and generalization of combinatorially structured patterns, defined as $\Phi = W^{(1)}W^{(2)}$, is best studied using multilayer networks and presents major differences [24]. Our analysis of this case in Fig. 2 (center) quantifies how many extra samples are needed so that a binary perceptron is able to correctly classify combinatorially correlated patterns with respect to random ones.

Two-layer decoder: The most exciting potential applications for the present results perhaps lie in the realm of deep neural networks where models such as ML-GLM with learned weights $W^{(\ell)}$ are used. A crucial ingredient of such neural networks are non-linear activation functions present among the layers. These activation functions can be seen as noisy channels, e.g. in the context of the decoder-side of an auto-encoder with known weights $W^{(\ell)}$: one is interested in how well a vector of latent variables can be reconstructed when y

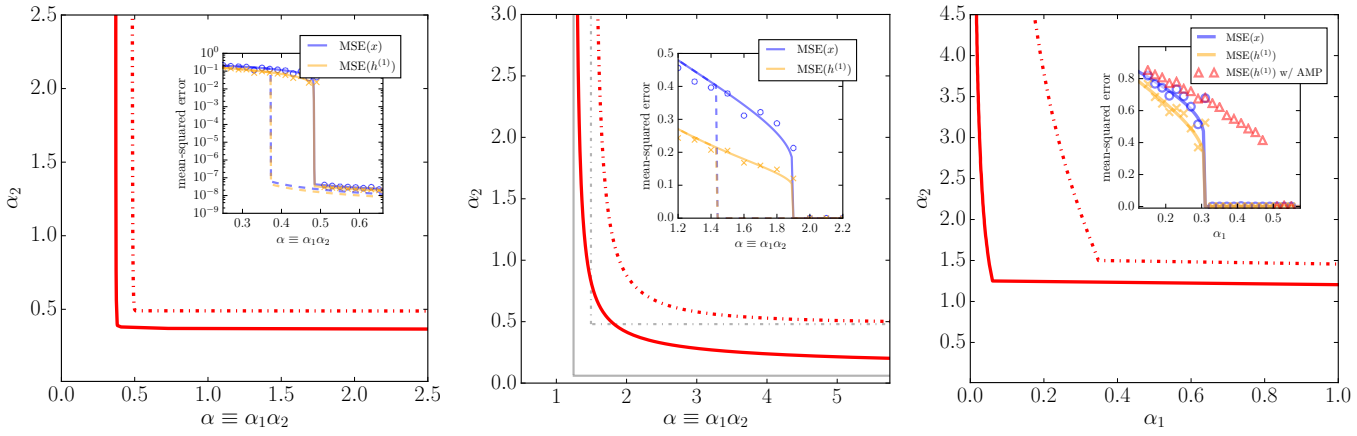

Fig. 2. **Main panels:** Phase diagram for sparse linear regression (left) and perceptron (center) with correlated data/patterns, defined by (18) and (19) respectively, and for the two-layer decoder (right), problem (20). Model parameters in SLR are $\rho = 0.3$, $\Delta_2 = 0$ and $\Delta_1 = 10^{-8}$, and in the two-layer decoder $\Delta_1 = \Delta_2 = 10^{-8}$. The ML-AMP algorithm succeeds to reconstruct the signal with very small MSE above the dotted red line. Between the two lines reconstruction is information-theoretically possible, but ML-AMP does not achieve it. Below the full red line good reconstruction is impossible. The grey lines plotted for the perceptron is a comparison with the case of random patterns. **Insets:** Comparisons between the MSE predicted by the state evolution (lines) and that provided by ML-AMP on a single instance with $n_\ell = 2000$ (symbols), $\alpha_2 = 1.0$ for the left and center figures, and $\alpha_2 = 2.0$ for the right. Dashed lines indicate the MMSE. Red triangles in the right inset compare to the performance of normal AMP in solving the decoder problem in the first layer assuming a binary i.i.d. prior on $h^{(1)}$, this works for $\alpha_1 \gtrsim 0.48$. ML-AMP takes into account correlations in $h^{(1)}$ and performs better.

is observed at the output. In Fig. 2 (right) we draw a phase diagram for the following example

$$\boldsymbol{y} = W^{(1)}\,\mathrm{sgn}(W^{(2)}\boldsymbol{x} + \mathcal{N}(0,\Delta_2)) + \mathcal{N}(0,\Delta_1), \qquad (20)$$

with $\boldsymbol{x} \sim \prod_{i=1}^{n_2}[\frac{1}{2}\delta(x_i-1)+\frac{1}{2}\delta(x_i+1)]$. Our results illustrate that the ML-AMP algorithm provides better results than a layer-wise estimation with ordinary AMP, because it takes into account correctly the correlations among the hidden variables.

Acknowledgment

This work has been supported by the ERC under the European Union's FP7 Grant Agreement 307087-SPARCS.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] R. Ranganath, D. Tran, and D. M. Blei, "Hierarchical Variational Models," in *International Conference on Machine Learning (ICML)*, 2016.

[3] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, 2010.

[4] M. Mézard and A. Montanari, *Information, physics, and computation*. Oxford University Press, 2009.

[5] T. Tanaka, "A statistical-mechanics approach to large-system analysis of cdma multiuser detectors," *IEEE Trans. Info. Theory*, vol. 48, no. 11, pp. 2888–2910, 2002.

[6] Y. Wu and S. Verdú, "Optimal phase transitions in compressed sensing," *IEEE Trans. Info. Theory*, vol. 58, no. 10, pp. 6241–6263, 2012.

[7] J. Barbier, M. Dia, N. Macris, and F. Krzakala, "The mutual information in random linear estimation," *arXiv preprint arXiv:1607.02335*, 2016.

[8] G. Reeves and H. D. Pfister, "The replica-symmetric prediction for compressed sensing with gaussian matrices is exact," in *2016 IEEE International Symposium on Information Theory (ISIT)*, 2016, p. 665.

[9] D. J. Thouless, P. W. Anderson, and R. G. Palmer, "Solution of 'solvable model of a spin glass'," *Philosophical Magazine*, vol. 35, p. 593, 1977.

[10] M. Mézard, "The space of interactions in neural networks: Gardner's computation with the cavity method," *J. Phys. A: Math. & Th.*, vol. 22, no. 12, p. 2181, 1989.

[11] Y. Kabashima, "Inference from correlated patterns: a unified theory for perceptron learning and linear vector channels," *J. Phys.: Conference Series*, vol. 95, p. 012001, 2008.

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

[13] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *2011 IEEE International Symposium on Information Theory (ISIT)*, Jul. 2011, pp. 2168–2172.

[14] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices," *J. Stat Mech.: Theory and Experiment*, vol. 2012, no. 08, p. P08009, 2012.

[15] R. Ranganath, L. Tang, L. Charlin, and D. Blei, "Deep Exponential Families," in *AISTATS*, 2015, pp. 762–771.

[16] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 764–785, 2011.

[17] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *Exploring artificial intelligence in the new millennium*, vol. 8, pp. 236–239, 2003.

[18] K. Takeda, S. Uda, and Y. Kabashima, "Analysis of cdma systems that are characterized by eigenvalue spectrum," *EPL*, vol. 76, p. 1193, 2006.

[19] A. M. Tulino, G. Caire, S. Verdú, and S. Shamai, "Support Recovery With Sparsely Sampled Free Random Matrices," *IEEE Trans. Info. Theory*, vol. 59, no. 7, pp. 4243–4271, Jul. 2013.

[20] Y. Kabashima and M. Vehkaperä, "Signal recovery using expectation consistent approximation for linear observations," in *2014 IEEE International Symposium on Information Theory (ISIT)*, Jun. 2014, p. 226.

[21] B. Çakmak, O. Winther, and B. H. Fleury, "S-AMP: Approximate message passing for general matrix ensembles," in *2014 IEEE Information Theory Workshop (ITW)*, Nov. 2014, pp. 192–196.

[22] S. Rangan, P. Schniter, and A. Fletcher, "Vector Approximate Message Passing," Oct. 2016, arXiv: 1610.03082.

[23] A. M. Tulino and S. Verdú, *Random Matrix Theory and Wireless Communications*. Now Publishers Inc, 2004.

[24] M. Mézard, "Mean-field message-passing equations in the Hopfield model and its generalizations," *arXiv:1608.01558*, 2016.

[25] E. Gardner and B. Derrida, "Three unfinished works on the optimal storage capacity of networks," *Journal of Physics A: Mathematical and General*, vol. 22, no. 12, p. 1983, 1989.

[26] A. Engel and C. Van den Broeck, *Statistical mechanics of learning*. Cambridge University Press, 2001.