Matrix Completion from Fewer Entries: Spectral Detectability and Rank Estimation

Alaa Saade[1] **and Florent Krzakala**[1,2]
[1] Laboratoire de Physique Statistique, CNRS & École Normale Supérieure, Paris, France.
[2]Sorbonne Universités, Université Pierre et Marie Curie Paris 06, F-75005, Paris, France

Lenka Zdeborová
Institut de Physique Théorique, CEA Saclay and CNRS UMR 3681, 91191 Gif-sur-Yvette, France

Abstract

The completion of low rank matrices from few entries is a task with many practical applications. We consider here two aspects of this problem: detectability, i.e. the ability to estimate the rank r reliably from the fewest possible random entries, and performance in achieving small reconstruction error. We propose a spectral algorithm for these two tasks called MaCBetH (for Matrix Completion with the Bethe Hessian). The rank is estimated as the number of negative eigenvalues of the Bethe Hessian matrix, and the corresponding eigenvectors are used as initial condition for the minimization of the discrepancy between the estimated matrix and the revealed entries. We analyze the performance in a random matrix setting using results from the statistical mechanics of the Hopfield neural network, and show in particular that MaCBetH efficiently detects the rank r of a large $n \times m$ matrix from $C(r)r\sqrt{nm}$ entries, where $C(r)$ is a constant close to 1. We also evaluate the corresponding root-mean-square error empirically and show that MaCBetH compares favorably to other existing approaches.

Matrix completion is the task of inferring the missing entries of a matrix given a subset of known entries. Typically, this is possible because the matrix to be completed has (at least approximately) low rank r. This problem has witnessed a burst of activity, see e.g. [1, 2, 3], motivated by many applications such as collaborative filtering [1], quantum tomography [4] in physics, or the analysis of a covariance matrix [1]. A commonly studied model for matrix completion assumes the matrix to be exactly low rank, with the known entries chosen uniformly at random and observed without noise. The most widely considered question in this setting is how many entries need to be revealed such that the matrix can be completed exactly in a computationally efficient way [1, 3]. While our present paper assumes the same model, the main questions we investigate are different.

The first question we address is *detectability*: how many random entries do we need to reveal in order to be able to estimate the rank r reliably. This is motivated by the more generic problem of detecting structure (in our case, low rank) hidden in partially observed data. It is reasonable to expect the existence of a region where exact completion is hard or even impossible yet the rank estimation is tractable. A second question we address is what is the minimum achievable root-mean-square error (RMSE) in estimating the unknown elements of the matrix. In practice, even if exact reconstruction is not possible, having a procedure that provides a very small RMSE might be quite sufficient.

In this paper we propose an algorithm called MaCBetH that gives the best known empirical performance for the two tasks above when the rank r is small. The rank in our algorithm is estimated as the number of negative eigenvalues of an associated Bethe Hessian matrix [5, 6], and the corresponding eigenvectors are used as an initial condition for the local optimization of a cost function commonly considered in matrix completion (see e.g. [3]). In particular, in the random matrix setting, we show

1

that MaCBetH detects the rank of a large $n \times m$ matrix from $C(r)r\sqrt{nm}$ entries, where $C(r)$ is a small constant, see Fig. 2, and $C(r) \to 1$ as $r \to \infty$. The RMSE is evaluated empirically and, in the regime close to $C(r)r\sqrt{nm}$, compares very favorably to existing approache such as OptSpace [3].

This paper is organized as follows. We define the problem and present generally our approach in the context of existing work in Sec. 1. In Sec. 2 we describe our algorithm and motivate its construction via a spectral relaxation of the Hopfield model of neural network. Next, in Sec. 3 we show how the performance of the proposed spectral method can be analyzed using, in parts, results from spin glass theory and phase transitions, and rigorous results on the spectral density of large random matrices. Finally, in Sec. 4 we present numerical simulations that demonstrate the efficiency of MaCBetH. Implementations of our algorithms in the Julia and Matlab programming languages are available at the SPHINX webpage `http://www.lps.ens.fr/~krzakala/WASP.html`.

1 Problem definition and relation to other work

Let $\mathcal{M}^{\text{true}}$ be a rank-r matrix such that

$$\mathcal{M}^{\text{true}} = XY^{\mathsf{T}}, \tag{1}$$

where $X \in \mathbb{R}^{n \times r}$ and $Y \in \mathbb{R}^{m \times r}$ are two (unknown) tall matrices. We observe only a small fraction of the elements of $\mathcal{M}^{\text{true}}$, chosen uniformly at random. We call E the subset of observed entries, and $\mathcal{M}$ the (sparse) matrix supported on E whose nonzero elements are the revealed entries of $\mathcal{M}^{\text{true}}$. The aim is to reconstruct the rank r matrix $\mathcal{M}^{\text{true}} = XY^{\mathsf{T}}$ given $\mathcal{M}$. An important parameter which controls the difficulty of the problem is $\epsilon = |E|/\sqrt{nm}$. In the case of a square matrix $\mathcal{M}$, this is the average number of revealed entries per line or column.

In our numerical examples and theoretical justifications we shall generate the low rank matrix $\mathcal{M}^{\text{true}} = XY^{\mathsf{T}}$, using tall matrices X and Y with iid Gaussian elements, we call this the random matrix setting. The MaCBetH algorithm is, however, non-parametric and does not use any prior knowledge about X and Y. The analysis we perform applies to the limit $n \to \infty$ while $m/n = \alpha = O(1)$ and $r = O(1)$.

The matrix completion problem was popularized in [1] who proposed nuclear norm minimization as a convex relaxation of the problem. The algorithmic complexity of the associated semidefinite programming is, however, $O(n^2 m^2)$. A low complexity procedure to solve the problem was later proposed by [7] and is based on singular value decomposition (SVD). A considerable step towards theoretical understanding of matrix completion from few entries was made in [3] who proved that with the use of *trimming* the performance of SVD-based matrix completion can be improved and a RMSE proportional to $\sqrt{nr/|E|}$ can be achieved. The algorithm of [3] is referred to as OptSpace, and empirically it achieves state-of-the-art RMSE in the regime of very few revealed entries.

OptSpace proceeds in three steps [3]. First, one trims the observed matrix $\mathcal{M}$ by setting to zero all rows (resp. columns) with more revealed entries than twice the average number of revealed entries per row (resp. per column). Second, a singular value decompositions is performed on the matrix and only the first r components are kept. When the rank r is unknown it is estimated as the index for which the ratio between two consecutive singular values has a minimum. Third, a local minimization of the discrepancy between the observed entries and the estimate is performed. The initial condition for this minimization is given by the first r left and right singular vectors from the second step.

In this work we improve upon OptSpace by replacing the first two steps by a different spectral procedure that detects the rank and provides a better initial condition for the discrepancy minimization. Our method leverages on recent progress made in the task of detecting communities in the stochastic block model [8, 5] with spectral methods. Both in community detection and matrix completion, traditional spectral methods fail in the very sparse regime due to the existence of spurious large eigenvalues (or singular values) corresponding to localized eigenvectors [8, 3]. The authors of [8, 5, 9] showed that using the non-backtracking matrix or the closely related Bethe Hessian as a basis for the spectral method in community detection provides reliable rank estimation and better inference performance. The present paper provides an analogous improvement for the matrix completion problem. In particular, we shall analyze the algorithm using tools from spin glass theory in statistical mechanics, and show that there exists a phase transition between a phase where it is able to detect the rank, and a phase where it is unable to do so.

2 Algorithm and motivation

2.1 The MaCBetH algorithm

A standard approach to the completion problem (see e.g. [3]) is to minimize the cost function

$$\min_{X,Y} \sum_{(ij)\in E} [\mathcal{M}_{ij} - (XY^{\mathsf{T}})_{ij}]^2 \tag{2}$$

over $X \in \mathbb{R}^{n\times r}$ and $Y \in \mathbb{R}^{m\times r}$. This function is non-convex, and global optimization is hard. One therefore resorts to a local optimization technique with a careful choice of the initial conditions X_0, Y_0. In our method, given the matrix $\mathcal{M}$, we consider a weighted bipartite undirected graph with adjacency matrix $A \in \mathbb{R}^{(n+m)\times(n+m)}$

$$A = \begin{pmatrix} 0 & \mathcal{M} \\ \mathcal{M}^T & 0 \end{pmatrix}. \tag{3}$$

We will refer to the graph thus defined as $\mathcal{G}$. We now define the Bethe Hessian matrix $\mathrm{H}(\beta) \in \mathbb{R}^{(n+m)\times(n+m)}$ to be the matrix with elements

$$\mathrm{H}_{ij}(\beta) = \left(1 + \sum_{k\in\partial i} \sinh^2 \beta A_{ik}\right) \delta_{ij} - \frac{1}{2} \sinh(2\beta A_{ij}), \tag{4}$$

where β is a parameter that we will fix to a well-defined value β_{SG} depending on the data, and ∂i stands for the neighbors of i in the graph $\mathcal{G}$. Expression (4) corresponds to the matrix introduced in [5], applied to the case of graphical model (6). The MaCBetH algorithm that is the main subject of this paper is then, given the matrix A, which we assume to be centered:

Algorithm (MaCBetH)

1. Numerically solve for the value of $\hat{\beta}_{\mathrm{SG}}$ such that $F(\hat{\beta}_{\mathrm{SG}}) = 1$, where

$$F(\beta) := \frac{1}{\sqrt{nm}} \sum_{(i,j)\in E} \tanh^2(\beta \mathcal{M}_{ij}). \tag{5}$$

2. Build the Bethe Hessian $\mathrm{H}(\hat{\beta}_{\mathrm{SG}})$ following eq. (4).

3. Compute *all* its negative eigenvalues $\lambda_1, \cdots, \lambda_{\hat{r}}$ and corresponding eigenvectors $v_1, \cdots, v_{\hat{r}}$. $\hat{r}$ is our estimate for the rank r. Set X_0 (resp. Y_0) to be the first n lines (resp. the last m lines) of the matrix $[v_1\ v_2\ \cdots\ v_{\hat{r}}]$.

4. Perform local optimization of the cost function (2) with rank $\hat{r}$ and initial condition X_0, Y_0.

In step 1, $\hat{\beta}_{\mathrm{SG}}$ is an approximation of the optimal value of β, for which $\mathrm{H}(\beta)$ has a maximum number of negative eigenvalues (see section 3). Instead of this approximation, β can be chosen in such a way as to maximize the number of negative eigenvalues. We however observed numerically that the algorithm is robust to some imprecision on the value of $\hat{\beta}_{\mathrm{SG}}$. In step 2 we could also use the non-backtracking matrix weighted by $\tanh \beta \mathcal{M}_{ij}$, it was shown in [5] that the spectrum of the Bethe Hessian and the non-backtracking matrix are closely related. In the next section, we will motivate and analyze this algorithm (in the setting where $\mathcal{M}^{\mathrm{true}}$ was generated from element-wise random X and Y) and show that in this case MaCBetH is able to infer the rank whenever $\epsilon > \epsilon_c$. Fig. 1 illustrates the spectral properties of the Bethe Hessian that justify this algorithm: the spectrum is composed of a few informative negative eigenvalues, well separated from the bulk (which remains positive). In particular, as observed in [8, 5], it avoids the spurious eigenvalues with localized eigenvectors that make trimming necessary in the case of [3]. This algorithm is computationally efficient as it is based on the eigenvalue decomposition of a sparse, symmetric matrix.

2.2 Motivation from a Hopfield model

We shall now motivate the construction of the MaCBetH algorithm from a graphical model perspective and a spectral relaxation. Given the observed matrix $\mathcal{M}$ from the previous section, we consider

the following graphical model

$$P(\{s\}, \{t\}) = \frac{1}{Z} \exp\left(\beta \sum_{(i,j)\in E} \mathcal{M}_{ij} s_i t_j\right),\tag{6}$$

where the $\{s_i\}_{1\le i\le n}$ and $\{t_j\}_{1\le j\le m}$ are binary variables, and β is a parameter controlling the strength of the interactions. This model is a (generalized) Hebbian Hopfield model on a bipartite sparse graph, and is therefore known to have r modes (up to symmetries) correlated with the lines of X and Y [10]. To study it, we can use the standard Bethe approximation which is widely believed to be exact for such problems on large random graphs [11, 12]. In this approximation the means $\mathbb{E}(s_i), \mathbb{E}(t_j)$ and moments $\mathbb{E}(s_i t_j)$ of each variable are approximated by the parameters b_i, c_j and ξ_{ij} that minimize the so-called Bethe free energy $F_{\text{Bethe}}(\{b_i\}, \{c_j\}, \{\xi_{ij}\})$ that reads

$$F_{\text{Bethe}}(\{b_i\}, \{c_j\}, \{\xi_{ij}\}) = -\sum_{(i,j)\in E} \mathcal{M}_{ij} \xi_{ij} + \sum_{(i,j)\in E} \sum_{s_i, t_j} \eta\Big(\frac{1 + b_i s_i + c_j t_j + \xi_{ij} s_i t_j}{4}\Big)$$
$$+ \sum_{i=1}^{n}(1 - d_i)\sum_{s_i}\eta\Big(\frac{1 + b_i s_i}{2}\Big) + \sum_{j=1}^{m}(1 - d_j)\sum_{t_j}\eta\Big(\frac{1 + c_j t_j}{2}\Big),\tag{7}$$

where $\eta(x) := x \ln x$, and d_i, d_j are the degrees of nodes i and j in the graph $\mathcal{G}$. Neural network models such as eq. (6) have been extensively studied over the last decades (see e.g. [12, 13, 14, 15, 16] and references therein) and the phenomenology, that we shall review briefly here, is well known. In particular, for β small enough, the global minimum of the Bethe free energy corresponds to the so-called *paramagnetic state*

$$\forall i, j, \qquad b_i = c_j = 0, \ \xi_{ij} = \tanh\left(\beta\mathcal{M}_{ij}\right).\tag{8}$$

As we increase β, above a certain value β_{R}, the model enters a *retrieval* phase, where the free energy has local minima correlated with the factors X and Y. There are r local minima, called *retrieval states* $(\{b_i^l\}, \{c_j^l\}, \{\xi_{ij}^l\})$ indexed by $l = 1, \cdots, r$ such that, in the large n, m limit,

$$\forall l = 1 \cdots r, \qquad \frac{1}{n}\sum_{i=1}^{n} X_{i,l} b_i^l > 0, \quad \frac{1}{m}\sum_{j=1}^{m} Y_{j,l} c_j^l > 0 \,.\tag{9}$$

These retrieval states are therefore convenient initial conditions for the local optimization of eq. (2), and we expect their number to tell us the correct rank. Increasing β above a critical value β_{SG} the system eventually enters a spin glass phase, marked by the appearance of many spurious minima.

It would be tempting to continue the Bethe approach leading to belief propagation, but we shall instead consider a simpler spectral relaxation of the problem, following the same strategy as used in [5, 6] for graph clustering. First, we use the fact that the paramagnetic state (8) is always a stationary point of the Bethe free energy, for any value of β [17, 18]. In order to detect the retrieval states, we thus study its stability by looking for negative eigenvalues of the Hessian of the Bethe free energy evaluated at the paramagnetic state (8). At this point, the elements of the Hessian involving one derivative with respect to ξ_{ij} vanish, while the block involving two such derivatives is a diagonal positive definite matrix [5, 17]. The remaining part is the matrix called Bethe Hessian in [5] (which however considers a different graphical model than (6)). Eigenvectors corresponding to its negative eigenvalues are thus expected to give an approximation of the retrieval states (9). The picture exposed in this section is summarized in Figure 1 and motivates the MaCBetH algorithm.

Note that a similar approach was used in [16] to detect the retrieval states of a Hopfield model using the weighted non-backtracking matrix [8], which linearizes the belief propagation equations rather than the Bethe free energy, resulting in a larger, non-symmetric matrix. The Bethe Hessian, while mathematically closely related, is also simpler to handle in practice.

3 Analysis of performance in detection

We now show how the performance of MaCBetH can be analyzed, and the spectral properties of the matrix characterized using both tools from statistical mechanics and rigorous arguments.
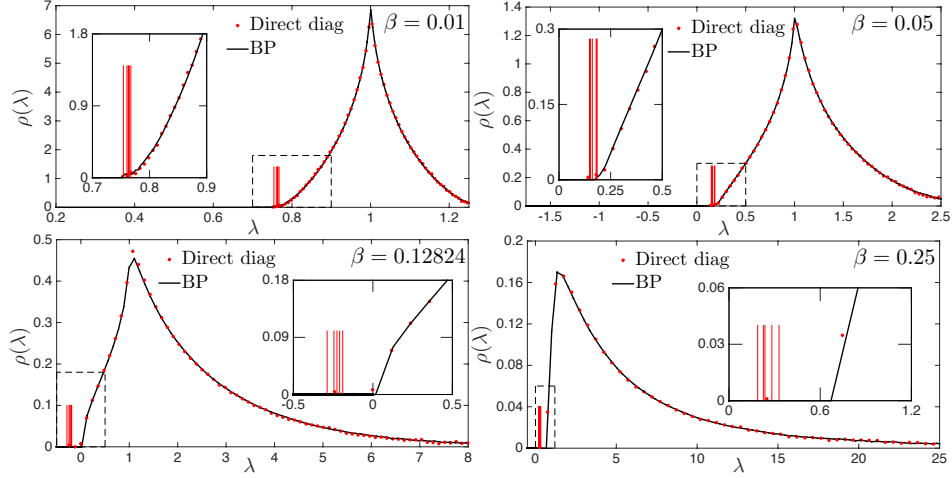
Figure 1: Spectral density of the Bethe Hessian for various values of the parameter β. Red dots are the result of the direct diagonalisation of the Bethe Hessian for a rank $r = 5$ and $n = m = 10^4$ matrix, with $\epsilon = 15$ revealed entries per row on average. The black curves are the solutions of (18) computed with belief propagation on a graph of size 10^5. We isolated the 5 smallest eigenvalues, represented as small bars for convenience, and the inset is a zoom around these smallest eigenvalues. For β small enough (top plots), the Bethe Hessian is positive definite, signaling that the paramagnetic state (8) is a local minimum of the Bethe free energy. As β increases, the spectrum is shifted towards the negative region and has 5 negative eigenvalues at the approximate value of $\hat{\beta}_{\mathrm{SG}} = 0.12824$ (to be compared to $\beta_R = 0.0832$ for this case) evaluated by our algorithm (lower left plot). These eigenvalues, corresponding to the retrieval states (9), become positive and eventually merge in the bulk as β is further increased (lower right plot), while the bulk of uninformative eigenvalues remains at all values of β in the positive region.

3.1 Analysis of the phase transition

We start by investigating the phase transition above which our spectral method will detect the correct rank. Let $x_p = (x_p^l)_{1 \le l \le r}, y_p = (y_p^l)_{1 \le l \le r}$ be random vectors with the same empirical distribution as the lines of X and Y respectively. Using the statistical mechanics correspondence between the negative eigenvalues of the Bethe Hessian and the appearance of phase transitions in model (6), we can compute the values β_R and β_{SG} where instabilities towards, respectively, the retrieval states and the spurious glassy states, arise. We have repeated the computations of [13, 14, 15, 16] in the case of model (6), using the cavity method [12]. We refer the reader interested in the technical details of the statistical mechanics approach to neural networks to [14, 15, 16].

Following a standard computation for locating phase transitions in the Bethe approximation (see e.g. [12, 19]), the stability of the paramagnetic state (8) towards these two phases can be monitored in terms of the two following parameters:

$$\lambda(\beta) = \lim_{s \to \infty} \mathbb{E}\Big[\prod_{p=1}^{s} \tanh^2\Big(\beta\sum_{l=1}^{r} x_p^l y_p^l\Big) \tanh^2\Big(\beta\sum_{l=1}^{r} x_{p+1}^l y_p^l\Big)\Big]^{\frac{1}{2s}}, \tag{10}$$

$$\mu(\beta) = \lim_{s \to \infty} \mathbb{E}\Big[\prod_{p=1}^{s} \tanh\Big(\beta|x_p^1 y_p^1| + \beta\sum_{l=2}^{r} x_p^l y_p^l\Big) \tanh\Big(\beta|x_{p+1}^1 y_p^1| + \beta\sum_{l=2}^{r} x_{p+1}^l y_p^l\Big)\Big]^{\frac{1}{2s}}, \tag{11}$$

where the expectation is over the distribution of the vectors x_p, y_p. The parameter $\lambda(\beta)$ controls the sensitivity of the paramagnetic solution to random noise, while $\mu(\beta)$ measures its sensitivity to a perturbation in the direction of a retrieval state. β_{SG} and β_R are defined implicitly as $\epsilon\lambda(\beta_{\mathrm{SG}}) = 1$ and $\epsilon\mu(\beta_R) = 1$, i.e. the value beyond which the perturbation diverges. The existence of a retrieval phase is equivalent to the condition $\beta_{\mathrm{SG}} > \beta_R$, so that there exists a range of values of β where the retrieval states exist, but not the spurious ones. If this condition is met, by setting $\beta = \beta_{\mathrm{SG}}$ in our algorithm, we ensure the presence of meaningful negative eigenvalues of the Bethe Hessian.

We define the critical value of $\epsilon = \epsilon_c$ such that $\beta_{\mathrm{SG}} > \beta_R$ if and only if $\epsilon > \epsilon_c$. In general, there is no closed-form formula for this critical value, which is defined implicitly in terms of the functions λ and μ. We thus computed ϵ_c numerically using a population dynamics algorithm [12] and the results for $C(r) = \epsilon_c/r$ are presented on Figure 2. Quite remarkably, with the definition $\epsilon = |E|/\sqrt{nm}$, the critical value ϵ_c does not depend on the ratio m/n, only on the rank r.

In the limit of large ϵ and r it is possible to obtain a simple closed-form formula. In this case the observed entries of the matrix become jointly Gaussian distributed, and uncorrelated, and therefore independent. Expression (10) then simplifies to

$$\lambda(\beta) =_{r \to \infty} \mathbb{E}\left[\tanh^2\left(\beta \sum_{l=1}^{r} x^l y^l \right) \right]. \quad (12)$$

Note that the MaCBetH algorithm uses an empirical estimator $F(\beta) \simeq \epsilon\lambda(\beta)$ (5) of this quantity to compute an approximation $\hat{\beta}_{\mathrm{SG}}$ of β_{SG} purely from the revealed entries. In the large r, ϵ regime, both $\beta_{\mathrm{SG}}, \beta_R$ decay to 0, so that we can further approximate

$$1 = \epsilon\lambda(\beta_{\mathrm{SG}}) \sim_{r \to \infty} \epsilon r \beta_{\mathrm{SG}}^2 \mathbb{E}[x^2]\mathbb{E}[y^2], \quad (13)$$

$$1 = \epsilon\mu(\beta_R) \sim_{r \to \infty} \epsilon\beta_R \sqrt{\mathbb{E}[x^2]\mathbb{E}[y^2]}, \quad (14)$$
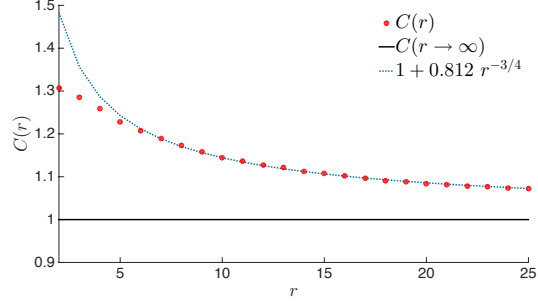

Figure 2: Location of the critical value as a function of the rank r. MaCBetH is able to estimate the correct rank from $|E| > C(r)r\sqrt{nm}$ known entries. We used a population dynamics algorithm with a population of size 10^6 to compute the functions λ and μ from (10,11). The dotted line is a fit suggesting that $C(r) - 1 = O(r^{-3/4})$.
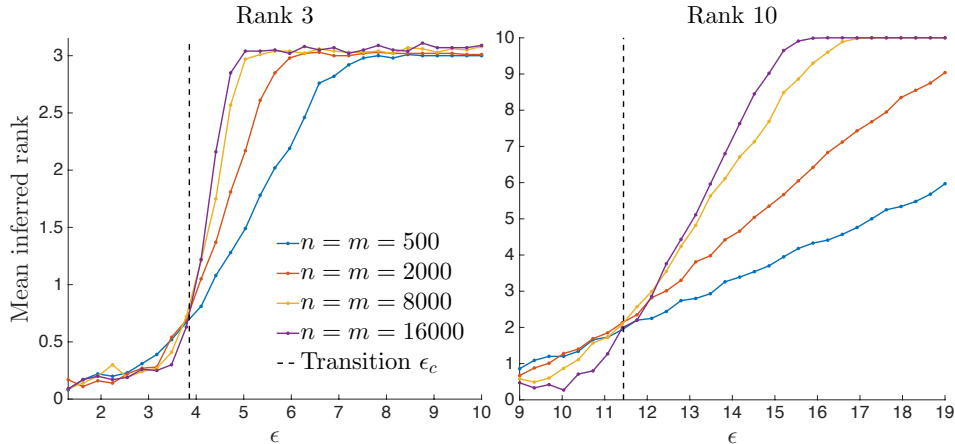
so that we reach the simple asymptotic expression, in the large ϵ, r limit, that $\epsilon_c = r$, or equivalently $C(r) = 1$. Interestingly, this result was obtained as the detectability threshold in completion of rank $r = O(n)$ matrices from $O(n^2)$ entries in the Bayes optimal setting in [20]. Notice, however, that *exact* completion in the setting of [20] is only possible for $\epsilon > r(m+n)/\sqrt{nm}$: clearly detection and exact completion are different phenomena. The previous analysis can be extended beyond the random setting assumption, as long as the empirical distribution of the entries is well defined, and the lines of X (resp. Y) are approximately orthogonal and centered. This condition is related to the standard incoherence property [1, 3].

3.2 Computation of the spectral density

In this section, we show how the spectral density of the Bethe Hessian can be computed analytically on tree-like graphs such as those generated by picking uniformly at random the observed entries of the matrix XY^{T}. This further motivates our algorithm and in particular our choice of $\beta = \hat{\beta}_{\mathrm{SG}}$, independently of section 3. The spectral density is defined as

$$\nu(\lambda) = \lim_{n,m \to \infty} \frac{1}{n+m} \sum_{i=1}^{n+m} \delta(\lambda - \lambda_i), \quad (15)$$

where the λ_i's are the eigenvalues of the Bethe Hessian. Using again the cavity method, it can be shown [21] that the spectral density (in which potential delta peaks have been removed) is given by

$$\nu(\lambda) = \lim_{n,m \to \infty} \frac{1}{\pi(n+m)} \sum_{i=1}^{n+m} \mathrm{Im}\Delta_i(\lambda), \quad (16)$$

where the Δ_i are complex variables living on the vertices of the graph $\mathcal{G}$, which are given by:

$$\Delta_i = \left(-\lambda + 1 + \sum_{k \in \partial i} \sinh^2 \beta A_{ik} - \sum_{l \in \partial i} \frac{1}{4}\sinh^2(2\beta A_{il})\Delta_{l \to i} \right)^{-1}, \quad (17)$$

where ∂i is the set of neighbors of i. The $\Delta_{i \to j}$ are the (linearly stable) solution of the following belief propagation recursion:

$$\Delta_{i \to j} = \left(-\lambda + 1 + \sum_{k \in \partial i} \sinh^2 \beta A_{ik} - \sum_{l \in \partial i \setminus j} \frac{1}{4}\sinh^2(2\beta A_{il})\Delta_{l \to i} \right)^{-1}. \quad (18)$$

6

Figure 3: Mean inferred rank as a function of ϵ, for different sizes, averaged over 100 samples of $n \times m \ XY^{\mathsf{T}}$ matrices. The entries of X, Y are drawn from a Gaussian distribution of mean 0 and variance 1. The theoretical transition is computed with a population dynamics algorithm (see section 3.1). The finite size effects are considerable but consistent with the asymptotic prediction.

This formula can be derived by turning the computation of the spectral density into a marginalization problem for a graphical model on the graph $\mathcal{G}$ and then solving it using loopy belief propagation. Quite remarkably, this approach leads to an asymptotically exact (and *rigorous* [22]) description of the spectral density on Erdős-Rényi random graphs. Solving equation (18) numerically we obtain the results shown on Fig. 1: the bulk of the spectrum, in particular, is always positive.

We now demonstrate that for any value of $\beta < \beta_{\mathrm{SG}}$, there exists an open set around $\lambda = 0$ where the spectral density vanishes. This justifies independently or choice for the parameter β. The proof follows [5] and begins by noticing that $\Delta_{i \to j} = \cosh^{-2}(\beta A_{ij})$ is a fixed point of the recursion (18) for $\lambda = 0$. Since this fixed point is real, the corresponding spectral density is 0. Now consider a small perturbation δ_{ij} of this solution such that $\Delta_{i \to j} = \cosh^{-2}(\beta A_{ij})(1 + \cosh^{-2}(\beta A_{ij})\delta_{ij})$. The linearized version of (18) writes $\delta_{i \to j} = \sum_{l \in \partial i \backslash j} \tanh^2(\beta A_{il})\delta_{i \to l}$. The linear operator thus defined is a weighted version of the non-backtracking matrix of [8]. Its spectral radius is given by $\rho = \epsilon \lambda(\beta)$, where λ is defined in 10. In particular, for $\beta < \beta_{\mathrm{SG}}$, $\rho < 1$, so that a straightforward application [5] of the implicit function theorem allows to show that there exists a neighborhood U of 0 such that for any $\lambda \in U$, there exists a real, linearly stable fixed point of (18), yielding a spectral density equal to 0. At $\beta = \hat{\beta}_{\mathrm{SG}}$, the informative eigenvalues (those outside of the bulk), are therefore exactly the negative ones, which motivates independently our algorithm.

4 Numerical tests

Figure 3 illustrates the ability of the Bethe Hessian to infer the rank above the critical value ϵ_c in the limit of large size n, m (see section 3.1). In Figure 4, we demonstrate the suitability of the eigenvectors of the Bethe Hessian as starting point for the minimization of the cost function (2). We compare the final RMSE achieved on the reconstructed matrix XY^{T} with 4 other initializations of the optimization, including the largest singular vectors of the trimmed matrix $\mathcal{M}$ [3]. MaCBetH systematically outperforms all the other choices of initial conditions, providing a better initial condition for the optimization of (2). Remarkably, the performance achieved by MaCBetH with the inferred rank is essentially the same as the one achieved with an oracle rank. By contrast, estimating the correct rank from the (trimmed) SVD is more challenging. We note that for the choice of parameters we consider, trimming had a negligible effect. Along the same lines, OptSpace [3] uses a different minimization procedure, but from our tests we could not see any difference in performance due to that. When using Alternating Least Squares [23, 24] as optimization method, we also obtained a similar improvement in reconstruction by using the eigenvectors of the Bethe Hessian, instead of the singular vectors of $\mathcal{M}$, as initial condition.
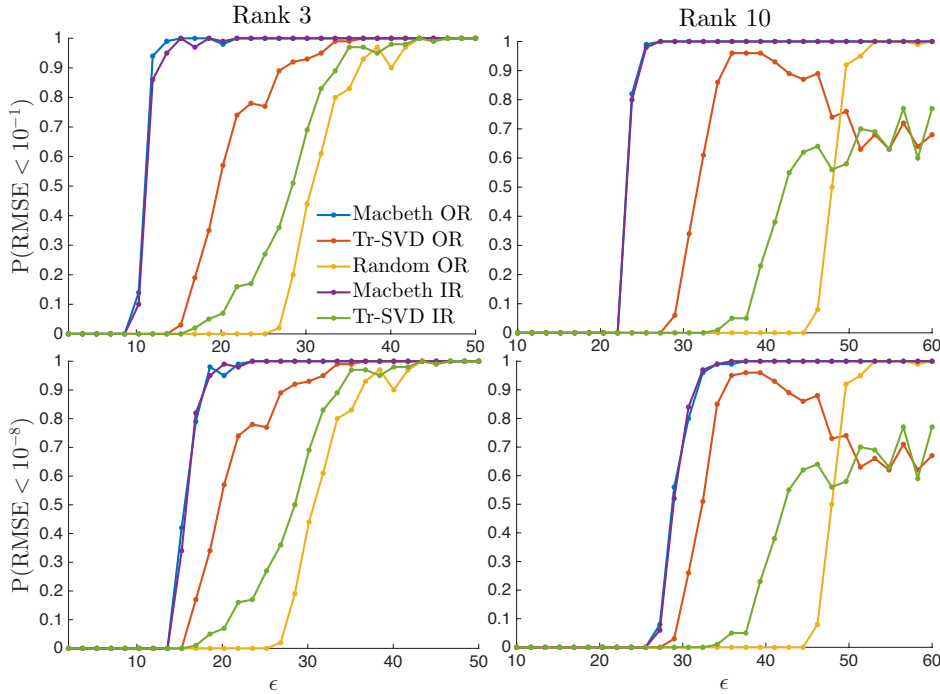
Figure 4: RMSE as a function of the number of revealed entries per row ϵ: comparison between different initializations for the optimization of the cost function (2). The top row shows the probability that the achieved RMSE is smaller than 10^{-1}, while the bottom row shows the probability that the final RMSE is smaller than 10^{-8}. The probabilities were estimated as the frequency of success over 100 samples of matrices XY^T of size 10000×10000, with the entries of X, Y drawn from a Gaussian distribution of mean 0 and variance 1. All methods optimize the cost function (2) using a low storage BFGS algorithm [25] part of NLopt [26], starting from different initial conditions. The maximum number of iterations was set to 1000. The initial conditions compared are MaCBetH with oracle rank (MaCBetH OR) or inferred rank (MaCBetH IR), SVD of the observed matrix $\mathcal{M}$ after trimming, with oracle rank (Tr-SVD OR), or inferred rank (Tr-SVD IR, note that this is equivalent to OptSpace [3] in this regime), and random initial conditions with oracle rank (Random OR). For the Tr-SVD IR method, we inferred the rank from the SVD by looking for an index for which the ratio between two consecutive eigenvalues is minimized, as suggested in [27].

5 Conclusion

In this paper, we have presented MaCBetH, an algorithm for matrix completion that is efficient for two distinct, complementary, tasks: (i) it has the ability to estimate a finite rank r reliably from fewer random entries than other existing approaches, and (ii) it gives lower root-mean-square reconstruction errors than its competitors. The algorithm is built around the Bethe Hessian matrix and leverages both on recent progresses in the construction of efficient spectral methods for clustering of sparse networks [8, 5, 9], and on the OptSpace approach [3] for matrix completion.

The method presented here offers a number of possible future directions, including replacing the minimization of the cost function by a message-passing type algorithm, the use of different neural network models, or a more theoretical direction involving the computation of information theoretically optimal transitions for detectability.

Acknowledgment

References

[1] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[2] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *Information Theory, IEEE Transactions on*, vol. 56, no. 5, pp. 2053–2080, 2010.

[3] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *Information Theory, IEEE Transactions on*, vol. 56, no. 6, pp. 2980–2998, 2010.

[4] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert, "Quantum state tomography via compressed sensing," *Physical review letters*, vol. 105, no. 15, p. 150401, 2010.

[5] A. Saade, F. Krzakala, and L. Zdeborová, "Spectral clustering of graphs with the bethe hessian," in *Advances in Neural Information Processing Systems*, 2014, pp. 406–414.

[6] A. Saade, F. Krzakala, M. Lelarge, and L. Zdeborová, "Spectral detection in the censored block model," *IEEE International Symposium on Information Theory (ISIT2015), to appear*, 2015.

[7] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[8] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang, "Spectral redemption in clustering sparse networks," *Proc. Natl. Acad. Sci.*, vol. 110, no. 52, pp. 20 935–20 940, 2013.

[9] C. Bordenave, M. Lelarge, and L. Massoulié, "Non-backtracking spectrum of random graphs: community detection and non-regular ramanujan graphs," 2015, arXiv:1501.06087.

[10] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci.*, vol. 79, no. 8, pp. 2554–2558, 1982.

[11] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Bethe free energy, kikuchi approximations, and belief propagation algorithms," *Advances in neural information processing systems*, vol. 13, 2001.

[12] M. Mezard and A. Montanari, *Information, Physics, and Computation*. Oxford University Press, 2009.

[13] D. J. Amit, H. Gutfreund, and H. Sompolinsky, "Spin-glass models of neural networks," *Physical Review A*, vol. 32, no. 2, p. 1007, 1985.

[14] B. Wemmenhove and A. Coolen, "Finite connectivity attractor neural networks," *Journal of Physics A: Mathematical and General*, vol. 36, no. 37, p. 9617, 2003.

[15] I. P. Castillo and N. Skantzos, "The little–hopfield model on a sparse random graph," *Journal of Physics A: Mathematical and General*, vol. 37, no. 39, p. 9087, 2004.

[16] P. Zhang, "Nonbacktracking operator for the ising model and its applications in systems with multiple states," *Physical Review E*, vol. 91, no. 4, p. 042120, 2015.

[17] J. M. Mooij and H. J. Kappen, "Validity estimates for loopy belief propagation on binary real-world networks." in *Advances in Neural Information Processing Systems*, 2004, pp. 945–952.

[18] F. Ricci-Tersenghi, "The bethe approximation for solving the inverse ising problem: a comparison with other inference methods," *J. Stat. Mech.: Th. and Exp.*, p. P08015, 2012.

[19] L. Zdeborová, "Statistical physics of hard optimization problems," *acta physica slovaca*, vol. 59, no. 3, pp. 169–303, 2009.

[20] Y. Kabashima, F. Krzakala, M. Mézard, A. Sakata, and L. Zdeborova, "Phase transitions and sample complexity in bayes-optimal matrix factorization," 2014, arXiv:1402.1298.

[21] T. Rogers, I. P. Castillo, R. Kühn, and K. Takeda, "Cavity approach to the spectral density of sparse symmetric random matrices," *Phys. Rev. E*, vol. 78, no. 3, p. 031116, 2008.

[22] C. Bordenave and M. Lelarge, "Resolvent of large random graphs," *Random Structures and Algorithms*, vol. 37, no. 3, pp. 332–352, 2010.

[23] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. ACM, 2013, pp. 665–674.

[24] M. Hardt, "Understanding alternating minimization for matrix completion," in *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*. IEEE, 2014, pp. 651–660.

[25] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1-3, pp. 503–528, 1989.

[26] S. G. Johnson, "The nlopt nonlinear-optimization package," 2014.

[27] R. H. Keshavan, A. Montanari, and S. Oh, "Low-rank matrix completion with noisy observations: a quantitative comparison," in *47th Annual Allerton Conference on Communication, Control, and Computing*, 2009, pp. 1216–1222.