Estimation in Rotationally Invariant Generalized Linear Models via Approximate Message Passing

Ramji Venkataramanan [1] **Kevin Kögler** [2] **Marco Mondelli** [2]

Abstract

We consider the problem of signal estimation in generalized linear models defined via rotationally invariant design matrices. Since these matrices can have an arbitrary spectral distribution, this model is well suited for capturing complex correlation structures which often arise in applications. We propose a novel family of approximate message passing (AMP) algorithms for signal estimation, and rigorously characterize their performance in the high-dimensional limit via a state evolution recursion. Our rotationally invariant AMP has complexity of the same order as the existing AMP derived under the restrictive assumption of a Gaussian design; our algorithm also recovers this existing AMP as a special case. Numerical results showcase a performance close to Vector AMP (which is conjectured to be Bayes-optimal in some settings), but obtained with a much lower complexity, as the proposed algorithm does not require a computationally expensive singular value decomposition.

1. Introduction

We consider the problem of estimating a $d-$dimensional signal $\boldsymbol{x}^* \in \mathbb{R}^d$ from an observation $\boldsymbol{y} \in \mathbb{R}^n$ obtained via a generalized linear model (GLM) (McCullagh & Nedler, 1989). Specifically, given a design matrix $\boldsymbol{A} \in \mathbb{R}^{n\times d}$ with rows $\boldsymbol{a}_1, \dots, \boldsymbol{a}_n \in \mathbb{R}^d$, the observation $\boldsymbol{y} \equiv (y_1, \dots, y_n)$ is generated as

$$y_i = q(\langle \boldsymbol{a}_i, \boldsymbol{x}^* \rangle, \varepsilon_i), \quad \text{for } i = 1, \dots, n, \tag{1}$$

where $\langle \boldsymbol{a}_i, \boldsymbol{x}^* \rangle = \boldsymbol{a}_i^\mathsf{T} \boldsymbol{x}^*$ denotes the Euclidean inner product, $\boldsymbol{\varepsilon} \equiv (\varepsilon_1, \dots, \varepsilon_n)$ is a noise vector and $q : \mathbb{R}^2 \to \mathbb{R}$ is a known function. The model (1) covers many widely studied problems in statistical estimation and signal processing: examples include linear regression (Donoho, 2006; Eldar & Kutyniok, 2012) ($y_i = \langle \boldsymbol{a}_i, \boldsymbol{x}^* \rangle + \varepsilon_i$), phase retrieval (Shechtman et al., 2015; Fannjiang & Strohmer, 2020) ($y_i = |\langle \boldsymbol{a}_i, \boldsymbol{x}^* \rangle|^2 + \varepsilon_i$), and 1-bit compressed sensing (Boufounos & Baraniuk, 2008) ($y_i = \text{sign}(\langle \boldsymbol{a}_i, \boldsymbol{x}^* \rangle + \varepsilon_i)$).

A range of estimators based on convex relaxations, spectral methods, and non-convex methods have been proposed for specific instances of GLMs, such as sparse linear regression (Tibshirani, 1996; Candés & Tao, 2007; Hastie et al., 2019), phase retrieval (Netrapalli et al., 2013; Candès et al., 2013; 2015; Mondelli & Montanari, 2019; Luo et al., 2019; Lu & Li, 2020) and one-bit compressed sensing (Plan & Vershynin, 2012; 2013; Jacques et al., 2013). Most of these techniques are generic and can incorporate certain constraints like sparsity, but they are not well-equipped to exploit specific information about $\boldsymbol{x}^*$, e.g., a known signal prior.

Approximate message passing (AMP) is a family of iterative algorithms that can be tailored to take advantage of structural information known about the signal. AMP algorithms were first proposed for estimation in linear models (Kabashima, 2003; Bayati & Montanari, 2012; 2011; Donoho et al., 2009; Krzakala et al., 2012; Maleki et al., 2013), but have since been applied to a range of statistical estimation problems, including generalized linear models (Barbier et al., 2019; Ma et al., 2019; Maillard et al., 2020; Mondelli & Venkataramanan, 2021a; Rangan, 2011; Schniter & Rangan, 2014; Sur & Candès, 2019) and low-rank matrix estimation (Deshpande & Montanari, 2014; Fletcher & Rangan, 2018; Kabashima et al., 2016; Lesieur et al., 2017; Montanari & Venkataramanan, 2021; Barbier et al., 2020). An attractive feature of AMP is that under suitable model assumptions, its performance in the high-dimensional limit is precisely characterized by a succinct deterministic recursion called *state evolution* (Bayati & Montanari, 2011; Bolthausen, 2014; Javanmard & Montanari, 2013). Using the state evolution analysis, it has been proved that AMP achieves Bayes-optimal performance for some models (Deshpande & Montanari, 2014; Donoho et al., 2013; Montanari & Venkataramanan, 2021; Barbier et al., 2019),

[1]University of Cambridge, United Kingdom [2]ISTA, Austria. Correspondence to: Ramji Venkataramanan <rv285@cam.ac.uk>, Marco Mondelli <marco.mondelli@ist.ac.at>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

and a conjecture from statistical physics posits that AMP is optimal among polynomial-time algorithms. The above works, including the original GAMP algorithm (Rangan, 2011) for estimation in GLMs, all assume that the matrix $\boldsymbol{A}$ defining the model is i.i.d. Gaussian. While some of these results can be generalized to the broader class of i.i.d. sub-Gaussian matrices via universality arguments (Bayati et al., 2015; Chen & Lam, 2021), the i.i.d. assumption limits the applicability of AMP in practice. In this paper, we present an AMP algorithm for generalized linear models defined via a rotationally invariant design matrix $\boldsymbol{A}$. The class of rotational invariant matrices includes Gaussian matrices, but is much bigger. Rotational invariance only imposes that the orthogonal matrices in the singular value decomposition of $\boldsymbol{A}$ are uniformly random, and allows for arbitrary singular values. Hence, $\boldsymbol{A}$ is able to capture a complex correlation structure, which is typical in applications.

Main contributions. We propose an AMP algorithm for GLMs with rotationally invariant design matrices. The algorithm, which we call RI-GAMP, uses a pair of multivariate 'denoising' functions to produce an updated signal estimate in each iteration. The iterates depend on the free cumulants of the spectral distribution of the design matrix. Assuming that these free cumulants are known (e.g., via the spectral distribution), the complexity of RI-GAMP is of the same order as that of the standard GAMP algorithm (Rangan, 2011). Moreover, when the design matrix is i.i.d. Gaussian, RI-GAMP reduces to standard GAMP. Our main technical contribution is a state evolution result for RI-GAMP (Theorem 3.1), which gives a rigorous characterization of its performance in the high-dimensional limit as $n, d \to \infty$ with a fixed ratio $\delta = n/d$, for a constant $\delta > 0$. We also present numerical simulation results for linear regression and 1-bit compressed sensing, showcasing the performance of RI-GAMP on both synthetic data and images. The performance of RI-GAMP closely matches that of Vector AMP (Rangan et al., 2019; Pandit et al., 2020) (which is conjectured to be Bayes-optimal in some settings), but obtained with much lower complexity, as RI-GAMP does not require computing the singular value decomposition of $\boldsymbol{A}$.

RI-GAMP offers a flexible framework to analyze other estimators for GLMs, e.g., spectral methods. Standard GAMP has been used as a proof technique to study the distributional properties of linear and spectral estimators under Gaussian model assumptions (Mondelli et al., 2021; Mondelli & Venkataramanan, 2021a). An exciting research direction is to use RI-GAMP to analyze spectral estimators for rotationally invariant GLMs in the high-dimensional limit.

Proof idea. The key idea in establishing the state evolution result is to design an auxiliary AMP algorithm whose iterates are close to those of RI-GAMP. The auxiliary AMP is an instance of the abstract AMP iteration for rotationally invariant matrices analyzed in (Fan, 2021; Zhong et al., 2021), hence a state evolution result can be obtained for it. We then show that each iterate of RI-GAMP is close to one of the auxiliary AMP, and use this fact to translate the state evolution result for the latter to the RI-GAMP. We emphasize that the auxiliary AMP only serves as a proof technique. Indeed, it is initialized using the unknown signal $\boldsymbol{x}^*$, and therefore cannot be used for estimation.

RI-GAMP vs. Vector AMP. Vector AMP (VAMP) is an iterative algorithm (based on Expectation Propagation) for estimation in rotationally invariant linear (Rangan et al., 2019; Takeuchi, 2020; 2021b) and generalized linear models (Schniter et al., 2016; Pandit et al., 2020). Like RI-GAMP, VAMP can be tailored to take advantage of prior information about the signal and its performance can be characterized by a state evolution recursion. It is also shown in Rangan et al. (2019); Pandit et al. (2020) that the asymptotic estimation error of VAMP (with optimal denoising functions) coincides with the replica prediction for the Bayes-optimal error whenever the state evolution recursion has a unique fixed point.

The RI-GAMP algorithm proposed here is distinct from VAMP, and the associated state evolution recursions are also different. Let us highlight some attractive features of RI-GAMP. First, RI-GAMP does not require the computationally expensive ($O(d^3)$) singular value decomposition used in VAMP. Instead, it uses the free cumulants of the design matrix which can be estimated in $O(d^2)$ time (details on p.4). We confirm via numerical simulations that the complexity advantage of RI-GAMP over VAMP is significant and increases with the problem dimension (see Figure 4).

The state evolution result (Theorem 3.1) for RI-GAMP holds under mild assumptions on the denoising functions (see **(A1)** on p.5), while the analysis for VAMP requires slightly stronger conditions, e.g., the denoising functions and their derivatives need to be uniformly Lipschitz continuous. The numerical results in Section 4 show that the performance of RI-GAMP matches that of VAMP, except near parameter values corresponding to a phase transition in the estimation error.

Other related work. Orthogonal AMP (Ma & Ping, 2017) is an algorithm equivalent to VAMP for estimation in rotationally invariant linear models. Çakmak et al. (2016) proposed a variant of Expectation Propagation (an algorithm closely related to VAMP) for rotationally invariant GLMs. Ma et al. (2021) recently studied the performance of Expectation Propagation for rotationally invariant GLMs, and analyzed the impact of the spectrum on the estimation performance. VAMP has also been used to obtain the asymptotic risk of convex penalized estimators for rotationally invariant

GLMs (Gerbelot et al., 2020a;b). A few lower complexity alternatives to VAMP have been proposed recently, including convolutional AMP (Takeuchi, 2021a), Memory AMP for linear models (Liu et al., 2020), and Generalized Memory AMP for GLMs (Tian et al., 2021). The phase retrieval problem (which is a special case of a GLM) has been studied for design matrices with orthogonal columns, a model distinct from the rotationally invariant one considered here (Dudeja et al., 2020a;b). Finally, we mention that AMP has also been applied to low-rank matrix estimation with rotationally invariant noise (Opper et al., 2016; Çakmak & Opper, 2019; Fan, 2021; Zhong et al., 2021; Mondelli & Venkataramanan, 2021b).

2. Preliminaries

Notation and definitions. For $n \in \mathbb{N}$, we use the shorthand $[n]$ to denote the set $\{1, \dots, n\}$. Given a vector $\boldsymbol{x}$, we write $\|\boldsymbol{x}\|$ for its ℓ_2 norm. All vectors are treated as column vectors. Given $\boldsymbol{x} = (x_1, \dots, x_d)$, we denote by $\langle \boldsymbol{x} \rangle$ its empirical average $\frac{1}{d}\sum_{i=1}^{d} x_i$. The empirical distribution of $\boldsymbol{x}$ is given by $\frac{1}{d}\sum_{i=1}^{d} \delta_{x_i}$, where δ_{x_i} denotes a Dirac delta mass on x_i. Similarly, the joint empirical distribution of the rows of a matrix $(\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^t) \in \mathbb{R}^{d \times t}$ is $\frac{1}{d}\sum_{i=1}^{d} \delta_{(x_i^1, \dots, x_i^t)}$. Given a matrix $\boldsymbol{A}$, we denote by $(\boldsymbol{A})_{i,j}$ its (i,j)-th element. For a square matrix $\boldsymbol{M}$, we follow the convention that $\boldsymbol{M}^0$ is the identity matrix.

W_2 convergence. We write $(\boldsymbol{x}^1, \dots, \boldsymbol{x}^k) \xrightarrow{W_2} (X_1, \dots, X_k)$ for the convergence in Wasserstein-2 distance of the joint empirical distribution of the rows of $(\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^k) \in \mathbb{R}^{d \times k}$ to the law of the random vector $(X_1, \dots, X_k)$. Equivalently (Feng et al., 2021)[Corollary 7.4], for any $L > 0$ and function $\psi : \mathbb{R}^k \to \mathbb{R}$ that satisfies

$$|\psi(\boldsymbol{u}) - \psi(\boldsymbol{v})| \le L\|\boldsymbol{u} - \boldsymbol{v}\| \left(1 + \|\boldsymbol{u}\| + \|\boldsymbol{v}\|\right) \tag{2}$$

for all $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^k$, we have

$$\lim_{d \to \infty} \frac{1}{d} \sum_{i=1}^{d} \psi(x_i^1, \dots, x_i^k) = \mathbb{E}\{\psi(X_1, \dots, X_k)\}. \tag{3}$$

A function satisfying (2) for some fixed $L > 0$ is called a pseudo-Lipschitz function of order 2.

Rotationally invariant generalized linear models. The $n \times d$ design matrix $\boldsymbol{A}$ is formed by stacking the sensing vectors $\boldsymbol{a}_1, \dots, \boldsymbol{a}_n$, i.e., $\boldsymbol{A} = [\boldsymbol{a}_1, \dots, \boldsymbol{a}_n]^\mathsf{T}$. We assume that $\boldsymbol{A}$ is bi-rotationally invariant in law, i.e., $\boldsymbol{A} = \boldsymbol{O}^\mathsf{T} \quad \boldsymbol{Q}$, where $\quad = \mathrm{diag}(\boldsymbol{\lambda})$ is an $n \times d$ diagonal matrix containing the singular values of $\boldsymbol{A}$, and $\boldsymbol{O}$, $\boldsymbol{Q}$ are Haar orthogonal matrices independent of one another and also of $\quad$. As $d \to \infty$, we assume that $n/d = \delta$, for some constant $\delta > 0$. The matrix $\boldsymbol{A}$ is independent of the signal $\boldsymbol{x}^* \in \mathbb{R}^d$, and the noise vector $\boldsymbol{\varepsilon} \in \mathbb{R}^n$. The observation $\boldsymbol{y} \in \mathbb{R}^n$ is generated according to (1). We assume that there exist random variables X_*, ε with finite second moments such that $\boldsymbol{x}^* \xrightarrow{W_2} X_*$, and $\boldsymbol{\varepsilon} \xrightarrow{W_2} \varepsilon$ as $n \to \infty$. Furthermore, we assume that the empirical distribution of $\boldsymbol{\lambda}$ converges weakly almost surely to a compactly supported random variable Λ. We denote by $\{\kappa_{2k}\}_{k \ge 1}$ the rectangular free cumulants associated with the moments $\{m_{2k}\}_{k \ge 1}$, where m_{2k} is the k-th moment of the empirical eigenvalue distribution of $\boldsymbol{A}\boldsymbol{A}^\mathsf{T}$ (for details, see (34)-(35) in Appendix A). For $\delta > 1$, let $\tilde{\Lambda}$ be a mixture of Λ (w.p. $1/\delta$) and a point mass at 0 (w.p. $1 - 1/\delta$); for $\delta \le 1$, we set $\tilde{\Lambda} = \Lambda$. Then, the assumptions above imply that as $n, d \to \infty$, $m_{2k} \to \bar{m}_{2k} = \mathbb{E}\{\tilde{\Lambda}^{2k}\}$ and $\kappa_{2k} \to \bar{\kappa}_{2k}$ almost surely, where $\{\bar{m}_{2k}\}_{k \ge 1}$ and $\{\bar{\kappa}_{2k}\}_{k \ge 1}$ are the even moments and rectangular free cumulants of $\tilde{\Lambda}$.

3. Rotationally Invariant Generalized AMP

Algorithm. We propose the following rotationally invariant generalized AMP (RI-GAMP) to estimate $\boldsymbol{x}^* \in \mathbb{R}^d$ from the observation $\boldsymbol{y} \in \mathbb{R}^n$ and the design matrix $\boldsymbol{A} \in \mathbb{R}^{n \times d}$. For $t \ge 1$, compute:

$$\boldsymbol{x}^t = \boldsymbol{A}^\mathsf{T} \boldsymbol{s}^t - \sum_{i=1}^{t} \beta_{ti}\, \hat{\boldsymbol{x}}^i, \qquad \hat{\boldsymbol{x}}^t = f_t(\boldsymbol{x}^1, \dots, \boldsymbol{x}^t), \tag{4}$$

$$\boldsymbol{r}^t = \boldsymbol{A}\hat{\boldsymbol{x}}^t - \sum_{i=1}^{t} \alpha_{ti}\, \boldsymbol{s}^i, \qquad \boldsymbol{s}^{t+1} = h_{t+1}(\boldsymbol{r}^1, \dots, \boldsymbol{r}^t, \boldsymbol{y}). \tag{5}$$

The iteration is initialized with $\boldsymbol{s}^1 = h_1(\boldsymbol{y})$ and $\boldsymbol{x}^1 = \boldsymbol{A}^\mathsf{T}\boldsymbol{s}^1$. For $t \ge 1$, the functions $f_t : \mathbb{R}^t \to \mathbb{R}$ and $h_{t+1} : \mathbb{R}^{t+1} \to \mathbb{R}$ are applied row-wise to vectors and matrices. The scalars $\{\alpha_{ti}\}_{i=1}^t$ and $\{\beta_{ti}\}_{i=1}^{t-1}$ are obtained in terms of two lower-triangular matrices $\quad_{t+1}, \quad_{t+1} \in \mathbb{R}^{(t+1)\times(t+1)}$. These matrices are defined as

$$\quad_{t+1} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & \langle \partial_1 \hat{\boldsymbol{x}}^1 \rangle & 0 & \dots & 0 & \\ 0 & \langle \partial_1 \hat{\boldsymbol{x}}^2 \rangle & \langle \partial_2 \hat{\boldsymbol{x}}^2 \rangle & \dots & 0 & \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & \langle \partial_1 \hat{\boldsymbol{x}}^t \rangle & \langle \partial_2 \hat{\boldsymbol{x}}^t \rangle & \dots & \langle \partial_t \hat{\boldsymbol{x}}^t \rangle & \end{pmatrix},$$

$$\quad_{t+1} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ \langle \partial_g \boldsymbol{s}^1 \rangle & 0 & 0 & \dots & 0 & \\ \langle \partial_g \boldsymbol{s}^2 \rangle & \langle \partial_1 \boldsymbol{s}^2 \rangle & 0 & \dots & 0 & \\ \vdots & \vdots & \ddots & & \vdots & \\ \langle \partial_g \boldsymbol{s}^t \rangle & \langle \partial_1 \boldsymbol{s}^t \rangle & \dots & \langle \partial_{t-1} \boldsymbol{s}^t \rangle & 0 & \end{pmatrix}, \tag{6}$$

where for $k \in [t]$, the vector $\partial_k \hat{\boldsymbol{x}}^t \in \mathbb{R}^d$ denotes the partial derivative $\partial_{x_k} f_t(x_1, \dots, x_t)$ applied row-wise to $\hat{\boldsymbol{x}}^t = f_t(\boldsymbol{x}^1, \dots, \boldsymbol{x}^t)$. Similarly, the vector $\partial_k \boldsymbol{s}^t \in \mathbb{R}^n$ denotes the partial derivative $\partial_{r_k} h_t(r_1, \dots, r_{t-1}, y)$ applied

row-wise to $s^t = h_t(r^1; \dots; r^{t-1}; y)$. Recalling that $y = q(g; \varepsilon)$, we can view $h_t(r_1; \dots; r_{t-1}; q(g; \varepsilon))$ as a function of $(t+1)$ variables, with $\partial_g h_t(r_1; \dots; r_{t-1}; q(g; \varepsilon))$ being the partial derivative with respect to g. For $t > 1$, the vector $\partial_g s^t$ denotes this partial derivative applied row-wise to $s^t = h_t(r^1; \dots; r^{t-1}; q(g; \varepsilon))$, and the vector $\partial_g s^1$ is defined via the partial derivative $\partial_g h_1(q(g; \varepsilon))$. Next, recalling that $\{\kappa_{2k}\}_{k \geq 1}$ denote the rectangular free cumulants, define matrices $M_{t+1}, \tilde{M}_{t+1} \in \mathbb{R}^{(t+1) \times (t+1)}$ as:

$$M_{t+1} = \sum_{j=0}^{k+1} \kappa_{2(j+1)} \, \tilde{\Delta}_{t+1} \, \tilde{\Delta}_{t+1} \, \tilde{\Delta}_{t+1}^{\,j}; \tag{7}$$

$$\tilde{M}_{t+1} = \sum_{j=0}^{t} \kappa_{2(j+1)} \, \tilde{\Delta}_{t+1} \, \tilde{\Delta}_{t+1} \, \tilde{\Delta}_{t+1}^{\,j}. \tag{8}$$

Then, the coefficients $\{\gamma_{ti}\}_{i=1}^{t}$ and $\{\tilde{\gamma}_{ti}\}_{i=1}^{t-1}$ in (4)-(5) are obtained from the last row of M_{t+1} and $\tilde{M}_{t+1}$ as:

$$(\gamma_{t1}, \dots, \gamma_{tt}) = ((M_{t+1})_{t+1;2}, \dots, (M_{t+1})_{t+1;t+1}); \tag{9}$$

$$(\tilde{\gamma}_{t1}, \dots, \tilde{\gamma}_{t,t-1}) = ((\tilde{M}_{t+1})_{t+1;2}, \dots, (\tilde{M}_{t+1})_{t+1;t}). \tag{10}$$

Estimating the free cumulants. From the definitions of $\{\gamma_{ti}\}_{i=1}^{t}$ and $\{\tilde{\gamma}_{ti}\}_{i=1}^{t-1}$ above, it follows that we need the free cumulants $\{\kappa_{2(j+1)}\}_{j=0}^{t+1}$ to compute the first t iterates of RI-GAMP in (5). These free cumulants can be recursively computed from the moments $\{m_{2(j+1)}\}_{j=0}^{t+1}$ of the spectral distribution of AA^T, using the formula (35) in Appendix A. The moments $\{m_{2(j+1)}\}_{j=0}^{t+1}$ can be estimated in $O(d^2)$ time via the following simple algorithm proposed by Liu et al. (2020). Given $A \in \mathbb{R}^{n \times d}$, pick an independent standard Gaussian vector $s^0 \sim N(0; I_n)$, and for $k \geq 1$, compute $s^k = A^T s^{k-1}$ for odd k and $s^k = A s^{k-1}$ for even k. Then, $\frac{\|s^k\|^2}{d}$ is a consistent estimate of the kth moment of the spectral distribution of AA^T. Thus, the complexity of estimating the free cumulants is of the same order as one iteration of RI-GAMP.

State evolution. The coefficients $\{\gamma_{ti}\}_{i=1}^{t}$ and $\{\tilde{\gamma}_{ti}\}_{i=1}^{t-1}$ play a crucial role in debiasing the AMP iterates, ensuring that their limiting empirical distributions are accurately captured by state evolution. Indeed, Theorem 3.1 shows that the joint empirical distribution of $(g; r^1; \dots; r^t)$ converges to a $(t+1)$-dimensional Gaussian distribution $N(0; \Sigma_{t+1})$. Similarly, the joint empirical distribution of $(x^1 - \Delta_1 x; \dots; x^t - \Delta_t x)$ converges to a t-dimensional Gaussian $N(0; \Gamma_t)$. We define the covariance matrices $\Sigma_t, \Gamma_t \in \mathbb{R}^{t \times t}$ and the vector $\Delta_t = (\Delta_1; \dots; \Delta_t)$ recursively for $t \geq 1$, starting with

$$\begin{aligned} \Sigma_1 &= \kappa_2 \mathbb{E}\{X^2\}; \quad \Delta_1 = \kappa_2 \mathbb{E}\{\partial_g h_1(q(G; \varepsilon))\}; \\ \Gamma_1 &= \kappa_2 \mathbb{E}\{h_1(q(G; \varepsilon))^2\} \\ &\quad + \kappa_4 \mathbb{E}\{X^2\} (\mathbb{E}\{\partial_g h_1(q(G; \varepsilon))\})^2; \end{aligned} \tag{11}$$

where $G \sim N(0; \kappa_2 \mathbb{E}\{X^2\})$ is independent of ε. Here, X is the law of the limiting empirical distribution of the signal, as defined in Section 2. For $t \geq 1$, given $\Sigma_t, \Gamma_t, \Delta_t$, let

$$\begin{aligned} &(G; R_1; \dots; R_{t-1}) \sim N(0; \Sigma_t); \\ &S_t = h_t(R_1; \dots; R_{t-1}; Y); \quad \text{where } Y = q(G; \varepsilon); \\ &(X_1; \dots; X_t) = \Delta_t X + (W_1; \dots; W_t); \\ &\text{where } (W_1; \dots; W_t) \sim N(0; \Gamma_t) \text{ is independent of } X; \\ &\hat{X}_t = f_t(X_1; \dots; X_t). \end{aligned} \tag{12, 13}$$

Let $\Sigma_{t+1}, \Gamma_{t+1} \in \mathbb{R}^{(t+1) \times (t+1)}$ be symmetric matrices with entries given by

$$\begin{aligned} &(\Sigma_{t+1})_{1;i} = (\Sigma_{t+1})_{i;1} = 0; \\ &(\Sigma_{t+1})_{i+1;j+1} = \mathbb{E}\{S_i S_j\}; \quad i, j \in [t]; \\ &(\Gamma_{t+1})_{1;1} = \mathbb{E}\{X^2\}; \\ &(\Gamma_{t+1})_{1;i+1} = (\Gamma_{t+1})_{i+1;1} = \mathbb{E}\{X \hat{X}_i\}; \\ &(\Gamma_{t+1})_{i+1;j+1} = \mathbb{E}\{\hat{X}_i \hat{X}_j\}; \quad i, j \in [t]. \end{aligned} \tag{14, 15}$$

Furthermore, let $\hat{\Sigma}_{t+1}, \hat{\Gamma}_{t+1}$ denote the deterministic versions of the matrices $\Sigma_{t+1}, \Gamma_{t+1}$ in (6), obtained by replacing the empirical averages by expectations. Specifically, to obtain $\hat{\Sigma}_{t+1}, \hat{\Gamma}_{t+1}$ we replace the entries as follows:

$$\begin{aligned} &\langle \partial_x x^t \rangle \to \mathbb{E}\{\partial_x \hat{X}_t\} = \mathbb{E}\{\partial_{x_k} f_t(X_1; \dots; X_t)\}; \\ &\langle \partial_x s^t \rangle \to \mathbb{E}\{\partial_x S_t\} = \mathbb{E}\{\partial_{R_k} h_t(R_1; \dots; R_{t-1}; q(G; \varepsilon))\}; \\ &\langle \partial_g s^t \rangle \to \mathbb{E}\{\partial_g S_t\} = \mathbb{E}\{\partial_g h_t(R_1; \dots; R_{t-1}; q(g; \varepsilon))|_{g=G}\}. \end{aligned} \tag{16}$$

We now describe how $\Sigma_{t+1}, \Delta_{t+1}, \Gamma_{t+1}$ are computed from $\Sigma_t, \Delta_t, \Gamma_t$. Given $\Sigma_t, \Delta_t, \Gamma_t$, we can evaluate the matrices $\hat{\Sigma}_{t+1}, \hat{\Delta}_{t+1}, \hat{\Gamma}_{t+1}, \hat{\Gamma}_{t+1}$. From these, we compute $\Delta_{t+1} \in \mathbb{R}^{(t+1) \times (t+1)}$ as

$$\Delta_{t+1} = \sum_{j=0}^{2k+1} \kappa_{2(j+1)} \, \Delta_{t+1}^{(j)}; \tag{17}$$

where $\Delta_{t+1}^{(0)} = \Delta_{t+1}$, and for $j \geq 1$:

$$\begin{aligned} \Delta_{t+1}^{(j)} &= \sum_{i=0}^{j} (\hat{\Sigma}_{t+1} \hat{\Delta}_{t+1})^i \, \hat{\Gamma}_{t+1} \, (\hat{\Sigma}_{t+1} \hat{\Delta}_{t+1})^{T\,(j-i)} \\ &+ \sum_{i=0}^{j-1} (\hat{\Sigma}_{t+1} \hat{\Delta}_{t+1})^i \, \hat{\Sigma}_{t+1} \hat{\Gamma}_{t+1}^T (\hat{\Sigma}_{t+1} \hat{\Delta}_{t+1})^{T\,(j-i-1)}. \end{aligned} \tag{18}$$

Recalling that $_{t+1}$ is the covariance of $(G; R_1; \dots; R_t)$, we can now compute $_{t+2}$; $_{t+2} \in R^{(t+2)} \ (t+2)$. Using these, we define a symmetric $(t+2) \ (t+2)$ matrix $^0_{t+2}$, whose first row and column equal zero and whose lower-right $(t+1) \ (t+1)$ submatrix equals $_{t+1}$. Specifically,

$$^0_{t+2} = \sum_{j=0}^{2(t+1)} \ 2(j+1) \ ^{(j)}_{t+2}; \tag{19}$$

where $^{(0)}_{t+2} = \ _{t+2}$, and for $j \ 1$:

$$^{(j)}_{t+2} = \sum_{i=0}^{j} (\ _{t+2} \ _{t+2})^i \ _{t+2} (\ _{t+2} \ _{t+2})^{T \ j \ i} + \sum_{i=0}^{j \ 1} (\ _{t+2} \ _{t+2})^i \ _{t+2} \ _{t+2}^T \ _{t+2} (\ _{t+2} \ _{t+2})^{T \ j \ i \ 1}. \tag{20}$$

Then, the entries of the covariance matrix $_{t+1} \in R^{(t+1)} \ (t+1)$ are given by:

$$(\ _{t+1})_{ij} = (\ ^0_{t+2})_{i+1;j+1}; \quad i; j \in [t+1]. \tag{21}$$

Finally, we compute the mean parameter $_{t+1} = M_{t+2} \ _{t+2;1}$, where

$$M_{t+2} = \sum_{j=0}^{t+1} \ 2(j+1) \ _{t+2} \ _{t+2} \ _{t+2}^j. \tag{22}$$

Though the formulas for $^{(j)}_{t+2}$ in (20) and M_{t+2} in (22) contain $_{t+2}$ and $_{t+2}$, the last rows and columns of these matrices are zeroed out in the computation (due to the form of $_{t+2}$ and $_{t+2}$). Therefore the formulas depend only on the top left submatrices of $_{t+2}$ and $_{t+2}$, namely, $_{t+1}$ and $_{t+1}$. We also note that the matrices $_t$ and $_t$ are the top left submatrices of $_{t+1}$ and $_{t+1}$, respectively. Similarly, the mean vector $_{t+1}$ is obtained by appending $_{t+1}$ to $_t$.

Main result. Having defined the state evolution recursion to compute $_t$; $_t$; $_t$ (which specify the joint distributions in (12)-(13)), we are ready to state our main result. We make the following assumption on the functions f_t, h_t used in the AMP (4)-(5), for $t \ 1$:

(A1) The functions $f_t(X_1; \dots; X_t)$ and $h_t(R_1; \dots; R_t \ _1; q(G; ''))$ are Lipschitz in each of their arguments. The partial derivatives $@_k f_t((X_1; \dots; X_t))$, $@_k h_t(R_1; \dots; R_t \ _1; q(G; ''))$, and $@_k h_t(R_1; \dots; R_t \ _1; q(G; ''))$ are all continuous on sets of probability 1, under the laws of $(X_1; \dots; X_t)$ and $(G; R_1; \dots; R_t \ _1)$ given in (12)-(13).

Theorem 3.1. Consider a rotationally invariant generalized linear model with the assumptions in Section 2 and the AMP (4)-(5) with the assumption (A1) above. Let $: R^{2t+1} \ ! \ R$ and $: R^{2t+2} \ ! \ R$ be any pseudo-Lipschitz functions of order 2. Then for each $t \ 1$, we almost surely have

$$\lim_{n!1} \frac{1}{d} \sum_{i=1}^{d} (x_i^1; \dots; x_i^t; \hat{x}_i^1; \dots; \hat{x}_i^t; x_i) = E\{ (X_1; \dots; X_t; \hat{X}_1; \dots; \hat{X}_t; X)\}, \tag{23}$$

$$\lim_{n!1} \frac{1}{n} \sum_{i=1}^{n} (r_i^1; \dots; r_i^t; s_i^1; \dots; s_i^{t+1}; y_i) = E\{ (R_1; \dots; R_t; S_1; \dots; S_{t+1}; Y)\}, \tag{24}$$

where the random variables on the right are defined in (12)-(13). Equivalently, as $n \ ! \ 1$, the joint empirical distributions of $(x^1; \dots; x^t; \hat{x}^1; \dots; \hat{x}^t; x)$ and $(r^1; \dots; r^t; s^1; \dots; s^{t+1}; y)$ converge almost surely in Wasserstein-2 distance to $(X_1; \dots; X_t; \hat{X}_1; \dots; \hat{X}_t; X)$ and $(R_1; \dots; R_t; S_1; \dots; S_{t+1}; Y)$, respectively.

The proof of the theorem is given in Appendix D, and we provide a proof sketch in Section 5. When the design matrix A has i.i.d. $N(0; 1=n)$ entries, we have $_2 = \ $ and $_{2k} = 0$ for $k \ 2$. In this case, the RI-GAMP (4)-(5), with denoising functions of the form $h_t(x^t)$ and $h_{t+1}(r^t; y)$, reduces to the existing GAMP algorithm (Rangan, 2011). The state evolution recursion also reduces to that of GAMP (see, e.g., Section 4 of (Feng et al., 2021)). This opens up an exciting research direction on using RI-GAMP to generalize results where GAMP has been used as a proof technique under Gaussian model assumptions. One example is to determine the distributional properties of spectral estimators for rotationally invariant GLMs.

MSE and correlation. The result (23) readily leads to the evaluation of the usual quantities of interest, such as the mean squared error (MSE) and the normalized squared correlation. Indeed, by taking $(\hat{x}_i^t; x_i) = (\hat{x}_i^t \ \ x_i)^2$, we have that $\frac{1}{d}\|\hat{x}^t \ \ x\|_2^2 \ ! \ E\{(\hat{X}_t \ \ X)^2\}$ for each $t \ 1$. Furthermore, by taking $(\hat{x}_i^t; x_i) = \hat{x}_i^t \ x_i$, $(\hat{x}_i^t) = (\hat{x}_i^t)^2$ and $(x_i) = (x_i)^2$, we have that $|\langle \hat{x}^t; x \rangle|^2 = (\|\hat{x}^t\|^2 \|x \ \|^2)$ tends to $(E\{\hat{X}_t X \ \ \})^2 = (E\{\hat{X}_t^2\} E\{X^2\})$.

Empirical state evolution parameters. We can define empirical versions of the state evolution parameters, denoted by $_{t+1}$; $_{t+1}$; $_{t+1}$, by replacing $_k$; $_k$; $_k$; $_k$ $(k \ 2 \ \{t+1; t+2\})$ with $_k$; $_k$; $_k$; $_k$ in (18), (20), and (22). The latter matrices are computed using empirical averages instead of expectations. $_k$; $_k$ are defined in (6) and for $_k$; $_k$, we replace the expectations $E\{S_i S_j\}$ and $E\{\hat{X}_i \hat{X}_j\}$ in (14)-(15) by $\langle s^i; s^j \rangle = n$ and $\langle \hat{x}^i; \hat{x}^j \rangle = d$. The expectations $E\{X \ \hat{X}_i\}$ in $_k$ can be estimated for the case

of posterior mean denoisers using the identity (57). Theorem 3.1 gives that the empirical versions of these matrices converge to the deterministic ones almost surely, and therefore, $_{t+1} \to _{t+1}$, $_{t+1} \to _{t+1}$, and $_{t+1} \to _{t+1}$. For the simulations in Section 4, RI-GAMP is implemented with state evolution parameters empirically estimated, as this choice leads to more stable numerical results.

Initialization. Note that the algorithm is initialized with $x^1 = A^T h_1(y)$. If this initialization is not effective, in the sense that $\frac{1}{d}\langle x^1, x \rangle \to 0$, then state evolution remains stuck at a trivial fixed point, i.e., $_t = 0$ for all t, and all the iterates produced by RI-GAMP are not correlated with the signal[1]. To address this issue, we can assume to be given an initialization x^1, which is correlated with x, i.e., $\frac{1}{d}\langle x^1, x \rangle \to > 0$, and independent of A. Theorem 3.1 still holds for such an initialization, with the only change being in the initialization of the state evolution recursion. Suppose that $x^1 \xrightarrow{W_2} X_1$ for a random variable X_1 satisfying $\mathbb{E}[X_1 X] = $. Then, in the state evolution initialization (11), we set $_1 = \mathbb{E}[X_1 X] = $ and $_1 = \mathbb{E}[(X_1 - {}_1 X)^2] $ (the parameter $_1$ is unchanged). This ensures that state evolution is not stuck at a trivial fixed point. A practical alternative to assuming an informative initialization is to initialize AMP with a spectral estimator. Analyzing RI-GAMP with spectral initialization is an interesting direction for future research.

Choice of denoisers. The performance of RI-GAMP is determined by the functions $\{f_t, h_{t+1}\}_{t \geq 1}$. A key question is how to choose these functions to optimize the estimation performance. Given any choice of $\{f_t, h_{t+1}\}$ satisfying Assumption (A1), Theorem 3.1 implies

$$r^{t-1} \xrightarrow{W_2} R_{t-1} \quad \frac{(\ _t)_{t;1}}{(\ _t)_{1;1}} G + W_{t-1}^0;$$
$$W_{t-1}^0 \sim \mathcal{N}\left(0; (\ _t)_{t;t} \quad \frac{((\ _t)_{t;1})^2}{(\ _t)_{1;1}}\right); \quad t \geq 2; \quad (25)$$
$$x^t \xrightarrow{W_2} X_t \quad {}_t X + W_t;$$
$$W_t \sim \mathcal{N}(0; (\ _t)_{t;t}) \text{ independent of } X; \quad t \geq 1; \quad (26)$$

where the RHS of (25) and (26) follow from the joint distributions specified in (12)-(13). From (26), we see that the quality of the estimate in each iteration is governed by the ratio $^2_t/(\ _t)_{t;t}$. Thus, having fixed $\{f_k, h_k\}_{k \leq t-1}$, the Bayes-optimal choice for $h_t(r_1, \ldots, r_{t-1}, y)$ maximizes $^2_t/(\ _t)_{t;t}$. Similarly, given $\{f_k\}_{k \leq t-1}$ and $\{h_k\}_{k \leq t}$, the Bayes-optimal choice for $f_t(x_1, \ldots, x_t)$ maximizes the normalized squared correlation between f_t and G, which is proportional to $\frac{(\ _{t+1})^2_{t+1;1}}{(\ _{t+1})_{t+1;t+1}}$. We remark that, even when the signal prior (i.e., the law of X) is known, finding these optimal denoisers is challenging due to the complicated nature of the state evolution recursion (12)-(22). However, for the special case of an i.i.d. Gaussian design, the state evolution is considerably simpler and the Bayes-optimal choices are (cf. Section 4.2 of (Feng et al., 2021)):

$$f_t(x_t) = c_1 \mathbb{E}[X \mid X_t = x_t]; \quad (27)$$
$$h_{t+1}(r_t, y) = c_2(\mathbb{E}[G \mid R_t = r_t, Y = y] \quad \mathbb{E}[G \mid R_t = r_t]);$$

where c_1, c_2 are arbitrary non-zero constants. Here, for a general rotationally invariant A, we propose the following denoisers:

$$f_t(x_1, \ldots, x_t) = \mathbb{E}[X \mid X_1 = x_1, \ldots, X_t = x_t]; \quad (28)$$
$$h_{t+1}(r_1, \ldots, r_t, y) = \mathbb{E}[G \mid R_1 = r_1, \ldots, R_t = r_t, Y = y]$$
$$\mathbb{E}[G \mid R_1 = r_1, \ldots, R_t = r_t]. \quad (29)$$

For an i.i.d. Gaussian A, (28)-(29) reduce to (27), which is provably Bayes-optimal. When A is not Gaussian, using denoisers that depend on all the preceding iterates (instead of only the most recent one) can have a remarkable impact on the performance of RI-GAMP. In fact, in the setting of Section 4 where the eigenvalues of A follow a Beta distribution, taking (27) does not improve much over the performance of the existing GAMP algorithm that assumes A to be Gaussian (green curves in Figures 2-3). In contrast, taking (28)-(29) leads to a performance close to VAMP (blue curves). Though we do not expect the choice (28)-(29) to be optimal iteration-by-iteration, based on the simulation results we conjecture that they achieve the same fixed point as the Bayes-optimal denoisers.

(a) Linear regression (b) 1-bit compressed sensing

Figure 1: Normalized squared correlation between iterate x^t and signal x, as a function of the number of iterations t. Solid lines represent state evolution predictions, and the markers represent the empirical performance of RI-GAMP.

[1] For both linear regression and 1-bit compressed sensing, the standard initialization $x^1 = A^T h_1(y)$ is effective. For a characterization of the GLMs for which this initialization is not effective for a Gaussian design matrix A, see (3.13) in (Mondelli & Venkataramanan, 2021a) and the discussion therein. One important example is phase retrieval ($y_i = |\langle a_i, x \rangle|^2 + \varepsilon_i$).

(a) $= 1$ (b) $= 0.1$

Figure 2: Linear regression with a Rademacher prior: normalized squared correlation vs. noise level (on the left) and vs. aspect ratio (on the right).

(a) Rademacher prior (b) Gaussian prior

Figure 3: Noiseless 1-bit compressed sensing: normalized squared correlation vs. aspect ratio, for two priors.

4. Numerical Simulations

For synthetic data, we consider two models: (i) linear regression, i.e., $y_i = \langle a_i, x_i \rangle + \varepsilon_i$, with $\varepsilon_i \sim N(0, \sigma^2)$; (ii) noiseless 1-bit compressed sensing, i.e., $y_i = \mathrm{sign}(\langle a_i, x_i \rangle)$. The design matrix A is rotationally invariant in law, i.e., $A = O^T \Phi Q$, where O, Q are Haar orthogonal matrices, and Φ has i.i.d. $\sqrt{6}\,\mathrm{Beta}(1, 2)$ diagonal entries. (The normalization of the $\mathrm{Beta}(1, 2)$ distribution is chosen to ensure a unit second moment.) In the simulations, the free cumulants Φ_{2k} are replaced by their limits Φ_{2k}, which can be obtained in closed-form (see Appendix A). We set $n = 8000$, repeat each experiment for 10 independent runs, and report the average and error bars at 1 standard deviation.

We implement the RI-GAMP given in (4)-(5), with initialization $s^1 = y$ and $x^1 = A^T s^1$. The denoisers f_t and h_{t+1}, for $t \geq 1$, are given by (28)-(29). The expressions for these denoisers and the associated calculations are given in Appendix B. The denoisers f_t, h_{t+1} and their derivatives depend on the state evolution parameters, which can be estimated consistently from the data. The implementation details are described at the end of Appendix B.

Figure 1 shows that the state evolution predictions closely match the performance of RI-GAMP for practical values of d and n, validating the result of Theorem 3.1. We plot the normalized squared correlation $\langle x^t, x_i \rangle^2 / (\|x^t\|^2 \|x_i\|^2)$ as a function of the iteration number. In (a), we consider linear regression with a Rademacher prior, $\sigma = 1$ and $\delta \in \{0.1, 0.4, 0.7\}$; and in (b), noiseless 1-bit compressed sensing with a Rademacher prior and $\delta \in \{0.8, 1.6, 2.4\}$. In all cases, the agreement between RI-GAMP and its SE is excellent. The next two figures show that the performance of RI-GAMP closely matches that of VAMP in a variety of settings. The results for linear regression with a Rademacher prior are shown in Figure 2: on the left, we plot the normalized squared correlation as a function of σ, for $\delta = 1$; on the right, we plot the same metric as a function of δ, for $\sigma = 0.1$. Additional results for a different choice of σ and δ are reported in Figure 6 in Appendix C. Figure 3 shows the performance for noiseless 1-bit compressed sensing: we plot the normalized squared correlation as a function of δ, for two signal priors (Rademacher in (a), and Gaussian in (b)). The red curve in each plot corresponds to RI-GAMP, together with the related SE. The blue curve corresponds to VAMP, together with the related SE. The implementation details for VAMP are given at the end of Appendix B. The green curve corresponds to the standard GAMP algorithm which is derived based on the (incorrect) assumption that A is i.i.d. Gaussian. The denoisers f_t and h_{t+1} are given by (27), which would be Bayes-optimal were the design matrix A Gaussian. The implementation of GAMP is a special case of our proposed RI-GAMP (obtained by setting all the rectangular free cumulants except Φ_2 to 0). The GAMP state evolution predictions (not shown in the plots) do not match the performance of the algorithm, since A is not Gaussian. Finally, the black curve corresponds to (i) the linear minimum mean squared error (LMMSE) estimator $\hat{x} = A^T(AA^T + \sigma I)^{-1} y$ for linear regression, and (ii) a subgradient method for 1-bit compressed sensing. This last method minimizes $\|[y \odot \mathrm{sign}(Ax)]_-\|_1$ via subgradient descent (here, $\odot$ denotes the Hadamard product and $[a]_- = \max\{-a, 0\}$ is applied component-wise). The algorithm was proposed in (Jacques et al., 2013) for the recovery of sparse signals, and the original version includes a sparsity enforcing step. For our setup (with no sparsity), we run it without the sparsity enforcing step, and the method reduces to subgradient descent.

Performance of RI-GAMP vs. VAMP. Taking the results of Figures 2-3 together, we highlight that RI-GAMP exhibits a performance close to VAMP. Recall that the fixed points of the VAMP state evolution satisfy the replica equation, whose solution is conjectured to give the Bayes-

(a) $d = 2000$ (b) $d = 4000$ (c) $d = 8000$

Figure 4: Running time of RI-GAMP vs. VAMP. The green curve shows the time for computing the free cumulants in RI-GAMP, and the blue curve the time for the SVD in the initial step of VAMP.

optimal mean squared error; see Theorem 3 of Rangan et al. (2019) for linear regression, and Theorem 4 of Pandit et al. (2020) for the general case. Thus, this conjecture implies that VAMP is Bayes-optimal when its state evolution has a unique fixed point, and our numerical simulations suggest that RI-GAMP is also near-optimal. We remark that, close to the phase transition for exact recovery (see $\delta = 0.6$ in Figure 2b, and $\delta = 1.8$ in Figure 3a), there is a (small) performance gap between RI-GAMP and VAMP. This is due to the fact that, for both RI-GAMP and VAMP, the number of iterations needed for convergence grows when the algorithm operates close to this phase transition. We note that VAMP shows a larger error bar for $\delta = 1.8$ in Figure 3a. For RI-GAMP, the issue is that the expressions of s_t and h_{t+1} depend on covariance matrices whose dimension grows with t. For large t, these covariance matrices become ill-conditioned and RI-GAMP is unstable. However, the stability displayed by VAMP comes at the cost of requiring the computationally expensive SVD of A. As shown next, RI-GAMP is significantly faster than VAMP, and is therefore an appealing alternative in many practical settings.

Complexity of RI-GAMP vs. VAMP. The computational complexity of VAMP is dominated by the initial SVD which has $O(d^3)$ running time. In contrast, the free cumulants required for RI-GAMP can be estimated in $O(d^2)$ time, as described on p.4. Each iteration of RI-GAMP also takes $O(d^2)$ time, and the algorithm typically converges in a few tens of iterations.

Figure 4 shows the running times for VAMP (including the initial SVD) and RI-GAMP (including the estimation of the free cumulants from the data), for noiseless 1-bit compressed sensing. The running time of VAMP is dominated by the SVD, and the computational advantage of RI-GAMP increases quickly with the problem dimension. For $\delta := \frac{n}{d} \in [0.5, 4]$, RI-GAMP is 20-60 faster than VAMP at $d = 2000$, 40-120 faster at $d = 4000$, and 80-240 faster at $d = 8000$.

Impact of eigenvalue distribution and prior. RI-GAMP exploits the spectral distribution of A, which gives a large performance improvement over the GAMP designed for a Gaussian A. Furthermore, RI-GAMP also takes advantage of the signal prior, which cannot be exploited by either the LMMSE estimator (for linear regression) or the subgradient method (for 1-bit compressed sensing). In fact, the subgradient method has roughly the same correlation for the two choices of the prior (cf. the black curves in Figure 3a and 3b), and is outperformed by RI-GAMP in both settings.

1-bit compressed sensing on a sparse image. In Figure 5, we consider noiseless 1-bit compressed sensing with the input x being the sparse grayscale image considered in (Schniter & Rangan, 2014), with $d = 225^2 = 50625$ and a sparsity (fraction of non-black pixels) of $0.645 \cdot 50625$. The design matrix A is $A = Q_n \Pi_n \Delta_d Q_d$, where Q_n, Q_d are orthonormal Discrete Cosine Transform (DCT) matrices in n, d dimensions, Π_n, Δ_d are random permutation matrices, and Δ has i.i.d. $\sqrt{6}\,\mathrm{Beta}(1, 2)$ diagonal entries. This choice of A significantly speeds up matrix multiplications, as in (Tian et al., 2021). We report the average and error bars at 1 standard deviation for 100 independent trials. For RI-GAMP, we use a non-negative Bernoulli-Gaussian prior (cf. (Schniter & Rangan, 2014) and (Vila & Schniter, 2014)); the expression for the corresponding denoiser f_t is in Appendix B. As shown in Figure 5a, RI-GAMP improves on the subgradient method in (Jacques et al., 2013) up until $\delta = 1.5$ (this improvement is clearly visibile in the reconstructions for $\delta = 0.8$, see Figures 5c and 5d). For larger δ, the performance of RI-GAMP does not improve further, due to the aforementioned numerical instabilities. Additional experiments on RGB images when the input x is obtained via a wavelet transform are reported in Appendix C.

5. Proof Sketch of Theorem 3.1

The proof is based on an auxiliary AMP algorithm whose iterates mimic the true AMP in (4)-(5). The iterates of the

(a) Correlation vs. (b) Original image (c) RI-GAMP, $= 0.8$ (d) Subgradient, $= 0.8$

Figure 5: RI-GAMP versus subgradient method for the recovery of a sparse image from 1-bit measurements.

auxiliary AMP, denoted by $z^t, v^t \in \mathbb{R}^d$ and $v^t, u^{t+1} \in \mathbb{R}^n$, are computed as follows, for $t \geq 1$:

$$z^t = A^T u^t - \sum_{i=1}^{t-1} b_{ti} v^i; \quad v^t = \tilde{f}_t(z^1, \ldots, z^t; x); \quad (30)$$

$$m^t = A v^t - \sum_{i=1}^{t} a_{ti} u^i; \quad u^{t+1} = \tilde{h}_{t+1}(m^1, \ldots, m^t; \varepsilon): \quad (31)$$

The iteration is initialized with $u^1 = 0$, $z^1 = 0$. The function $\tilde{f}_1 : \mathbb{R}^2 \to \mathbb{R}$ is defined as $\tilde{f}_1(z_1; x) = x$, which yields $v^1 = x$ and $m^1 = Ax = g$. For $t \geq 1$, the functions $\tilde{f}_{t+1} : \mathbb{R}^{t+2} \to \mathbb{R}$ and $\tilde{h}_{t+1} : \mathbb{R}^{t+1} \to \mathbb{R}$, which act row-wise on matrices, are defined as

$$\begin{aligned} &\tilde{f}_{t+1}(z_1, \ldots, z_{t+1}; x) \\ &\quad = f_t(z_2 + \gamma_1 x, z_3 + \gamma_2 x, \ldots, z_{t+1} + \gamma_t x); \quad (32) \\ &\tilde{h}_{t+1}(m_1, \ldots, m_t; \varepsilon) = h_t(m_2, \ldots, m_t; q(m_1; \varepsilon)): \end{aligned}$$

Here, f_t, h_t are the functions from the original AMP, and $(\gamma_1, \ldots, \gamma_t)$ are the state evolution parameters computed according to (22). The debiasing coefficients $\{a_{ti}\}_{i=1}^t$ and $\{b_{ti}\}_{i=1}^{t-1}$ for the auxiliary AMP are given in Appendix D.1.

Idea of the proof. The auxiliary AMP (30)-(31) is an instance of the abstract AMP recursion for non-symmetric rotationally invariant matrices, which was analyzed in (Zhong et al., 2021). The state evolution result in Theorem 2.6 of (Zhong et al., 2021) implies that the joint empirical distribution of $(m^1, \ldots, m^t)$ converges to a t-dimensional Gaussian $N(0, \Sigma_t)$. Similarly, the joint empirical distribution of $(z^1, \ldots, z^t)$ also converges to a t-dimensional Gaussian $N(0, \Phi_t)$. The covariance matrices Σ_t, Φ_t are recursively defined via the state evolution for the auxiliary AMP (for details, see Section D.2).

The proof of Theorem 3.1 consists of two steps. First, we show that the state evolution parameters of the auxiliary AMP match those of the true AMP. In particular, Lemma D.2 proves that $\Phi_t = \Gamma_t$ and $\Sigma_{t+1} = \Gamma^0_{t+1}$, where the matrices on the right are defined in (17) and (19). Next, we show in Lemma D.3 that the true AMP iterates (4)-(5) are close to the auxiliary AMP iterates (30)-(31) in the following sense. For $t \geq 1$:

$$\frac{\|x^t - (z^{t+1} + \gamma_t x)\|^2}{d} \to 0; \quad \frac{\|x^t - v^{t+1}\|^2}{d} \to 0;$$
$$\frac{\|r^t - m^{t+1}\|^2}{n} \to 0; \quad \frac{\|s^t - u^{t+1}\|^2}{n} \to 0: \quad (33)$$

Lemma D.3 actually proves a more general convergence statement which implies (33). It shows that the empirical joint distribution of the iterates of the true AMP converges to that of the auxiliary AMP. The result of Theorem 3.1 then follows from Lemmas D.2 and D.3.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments. KK and MM were partially supported by the 2019 Lopez-Loreta Prize.

References

Barbier, J., Krzakala, F., Macris, N., Miolane, L., and Zdeborová, L. Optimal errors and phase transitions in high-dimensional generalized linear models. Proceedings of the National Academy of Sciences, 116(12):5451–5460, 2019.

Barbier, J., Macris, N., and Rush, C. All-or-nothing statistical and computational phase transitions in sparse spiked matrix estimation. In Neural Information Processing Systems (NeurIPS), 2020.

Bayati, M. and Montanari, A. The dynamics of message passing on dense graphs, with applications to compressed sensing. IEEE Transactions on Information Theory, 57: 764–785, 2011.

Bayati, M. and Montanari, A. The LASSO risk for Gaussian matrices. IEEE Transactions on Information Theory, 58: 1997–2017, 2012.

Bayati, M., Lelarge, M., Montanari, A., et al. Universality in polytope phase transitions and message passing algorithms. The Annals of Applied Probability, 25(2): 753–822, 2015.

Benaych-Georges, F. Rectangular random matrices, related convolution. Probability Theory and Related Fields, 144 (3-4):471–515, 2009.

Bolthausen, E. An iterative construction of solutions of the TAP equations for the Sherrington–Kirkpatrick model. Communications in Mathematical Physics, 325(1):333–366, 2014.

Boufounos, P. T. and Baraniuk, R. G. 1-bit compressive sensing. In Conference on Information Sciences and Systems (CISS), pp. 16–21, 2008.

Çakmak, B. and Opper, M. Memory-free dynamics for the Thouless-Anderson-Palmer equations of Ising models with arbitrary rotation-invariant ensembles of random coupling matrices. Physical Review E, 99(6):062140, 2019.

Çakmak, B., Opper, M., Fleury, B. H., and Winther, O. Self-averaging expectation propagation. arXiv:1608.06602, 2016.

Candes, E. and Tao, T. The Dantzig selector: statistical estimation when p is much larger than n. Annals of Statistics, 35:2313–2351, 2007.

Candes, E. J., Strohmer, T., and Voroninski, V. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. Communications on Pure and Applied Mathematics, 66(8):1241–1274, 2013.

Candes, E. J., Li, X., and Soltanolkotabi, M. Phase retrieval via Wirtinger flow: Theory and algorithms. IEEE Transactions on Information Theory, 61(4):1985–2007, 2015.

Chen, W.-K. and Lam, W.-K. Universality of approximate message passing algorithms. Electronic Journal of Probability, 26:1 – 44, 2021.

Deshpande, Y. and Montanari, A. Information-theoretically optimal sparse PCA. In IEEE International Symposium on Information Theory (ISIT), pp. 2197–2201, 2014.

Donoho, D. L. Compressed sensing. IEEE Transactions on Information Theory, 52:489–509, April 2006.

Donoho, D. L., Maleki, A., and Montanari, A. Message Passing Algorithms for Compressed Sensing. Proceedings of the National Academy of Sciences, 106:18914–18919, 2009.

Donoho, D. L., Javanmard, A., and Montanari, A. Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing. IEEE Transactions on Information Theory, 59(11):7434–7464, Nov. 2013.

Dudeja, R., Bakhshizadeh, M., Ma, J., and Maleki, A. Analysis of spectral methods for phase retrieval with random orthogonal matrices. IEEE Transactions on Information Theory, 66(8):5182–5203, 2020a.

Dudeja, R., Ma, J., and Maleki, A. Information theoretic limits for phase retrieval with subsampled haar sensing matrices. IEEE Transactions on Information Theory, 66: 8002 – 8045, Decemeber 2020b.

Eldar, Y. C. and Kutyniok, G. Compressed sensing: Theory and applications. Cambridge University Press, 2012.

Fan, Z. Approximate message passing algorithms for rotationally invariant matrices. Annals of Statistics, 2021. to appear.

Fannjiang, A. and Strohmer, T. The numerics of phase retrieval. Acta Numerica, 29:125–228, 2020.

Feng, O. Y., Venkataramanan, R., Rush, C., and Samworth, R. J. A unifying tutorial on Approximate Message Passing. arXiv:2105.02180, 2021.

Fletcher, A. K. and Rangan, S. Iterative reconstruction of rank-one matrices in noise. Information and Inference: A Journal of the IMA, 7(3):531–562, 2018.

Gerbelot, C., Abbara, A., and Krzakala, F. Asymptotic errors for high-dimensional convex penalized linear regression beyond Gaussian matrices. In Conference on Learning Theory (COLT), pp. 1682–1713, 2020a.

Gerbelot, C., Abbara, A., and Krzakala, F. Asymptotic errors for teacher-student convex generalized linear models (or: How to prove Kabashima's replica formula). arXiv:2006.06581, 2020b.

Hastie, T., Tibshirani, R., and Wainwright, M. Statistical learning with sparsity: the Lasso and generalizations. Chapman and Hall/CRC, 2019.

Jacques, L., Laska, J. N., Boufounos, P. T., and Baraniuk, R. G. Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. IEEE Transactions on Information Theory, 59(4):2082–2102, 2013.

Javanmard, A. and Montanari, A. State evolution for general approximate message passing algorithms, with applications to spatial coupling. Information and Inference, pp. 115–144, 2013.

Kabashima, Y. A CDMA multiuser detection algorithm on the basis of belief propagation. Journal of Physics A: Mathematical and General, 36(43):11111–11121, Oct 2003.

Kabashima, Y., Krzakala, F., Mézard, M., Sakata, A., and Zdeborová, L. Phase transitions and sample complexity in Bayes-optimal matrix factorization. IEEE Transactions on Information Theory, 62(7):4228–4265, 2016.

Krzakala, F., Mézard, M., Sausset, F., Sun, Y., and Zdeborová, L. Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices. Journal of Statistical Mechanics: Theory and Experiment, 2012(08):P08009, 2012.

Lee, G. R., Gommers, R., Waselewski, F., Wohlfahrt, K., and O'Leary, A. Pywavelets: A python package for wavelet analysis. Journal of Open Source Software, 4(36): 1237, 2019. doi: 10.21105/joss.01237. URL https://doi.org/10.21105/joss.01237.

Lesieur, T., Krzakala, F., and Zdeborová, L. Constrained low-rank matrix estimation: Phase transitions, approximate message passing and applications. Journal of Statistical Mechanics: Theory and Experiment, 2017(7): 073403, 2017.

Liu, L., Huang, S., and Kurkoski, B. M. Memory approximate message passing. arXiv:2012.10861, 2020.

Lu, Y. M. and Li, G. Phase transitions of spectral initialization for high-dimensional non-convex estimation. Information and Inference: A Journal of the IMA, 9(3): 507–541, 2020.

Luo, W., Alghamdi, W., and Lu, Y. M. Optimal spectral initialization for signal recovery with applications to phase retrieval. IEEE Transactions on Signal Processing, 67(9): 2347–2356, 2019.

Ma, J. and Ping, L. Orthogonal AMP. IEEE Access, 5: 2020–2033, 2017.

Ma, J., Xu, J., and Maleki, A. Optimization-based AMP for phase retrieval: The impact of initialization and ℓ_2 regularization. IEEE Transactions on Information Theory, 65(6):3600–3629, 2019.

Ma, J., Xu, J., and Maleki, A. Impact of sensing spectrum for signal recovery under a generalized linear model. In Neural Information Processing Systems (NeurIPS), 2021.

Maillard, A., Loureiro, B., Krzakala, F., and Zdeborová, L. Phase retrieval in high dimensions: Statistical and computational phase transitions. In Neural Information Processing Systems (NeurIPS), 2020.

Maleki, A., Anitori, L., Yang, Z., and Baraniuk, R. G. Asymptotic analysis of complex lasso via complex approximate message passing (CAMP). IEEE Transactions on Information Theory, 59(7):4290–4308, 2013.

McCullagh, P. and Nedler, J. Generalized linear models. Chapman and Hall/CRC, 1989.

Mondelli, M. and Montanari, A. Fundamental limits of weak recovery with applications to phase retrieval. Foundations of Computational Mathematics, 19:703–773, 2019.

Mondelli, M. and Venkataramanan, R. Approximate message passing with spectral initialization for generalized linear models. In International Conference on Artificial Intelligence and Statistics (AISTATS), pp. 397–405. PMLR, 2021a.

Mondelli, M. and Venkataramanan, R. PCA initialization for approximate message passing in rotationally invariant models. In Neural Information Processing Systems (NeurIPS), 2021b.

Mondelli, M., Thrampoulidis, C., and Venkataramanan, R. Optimal combination of linear and spectral estimators for generalized linear models. Foundations of Computational Mathematics, pp. 1–54, 2021.

Montanari, A. and Venkataramanan, R. Estimation of low-rank matrices via approximate message passing. Annals of Statistics, 45(1):321–345, 2021.

Netrapalli, P., Jain, P., and Sanghavi, S. Phase retrieval using alternating minimization. In Neural Information Processing Systems (NIPS), pp. 2796–2804, 2013.

Opper, M., Cakmak, B., and Winther, O. A theory of solving TAP equations for Ising models with general invariant random matrices. Journal of Physics A: Mathematical and Theoretical, 49(11):114002, 2016.

Pandit, P., Sahraee-Ardakan, M., Rangan, S., Schniter, P., and Fletcher, A. K. Inference with deep generative priors in high dimensions. IEEE Journal on Selected Areas in Information Theory, 1(1):336–347, 2020.

Plan, Y. and Vershynin, R. Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach. IEEE Transactions on Information Theory, 59 (1):482–494, 2012.

Plan, Y. and Vershynin, R. One-bit compressed sensing by linear programming. Communications on Pure and Applied Mathematics, 66(8):1275–1297, 2013.

Rangan, S. Generalized Approximate Message Passing for Estimation with Random Linear Mixing. In IEEE International Symposium on Information Theory (ISIT), 2011.

Rangan, S., Schniter, P., and Fletcher, A. K. Vector approximate message passing. IEEE Transactions on Information Theory, 65(10):6664–6684, 2019.

Schniter, P. and Rangan, S. Compressive phase retrieval via generalized approximate message passing. IEEE Transactions on Signal Processing, 63(4):1043–1055, 2014.

Schniter, P., Rangan, S., and Fletcher, A. K. Vector approximate message passing for the generalized linear model. In 50th Asilomar Conference on Signals, Systems and Computers, pp. 1525–1529. IEEE, 2016.

Shechtman, Y., Eldar, Y. C., Cohen, O., Chapman, H. N., Miao, J., and Segev, M. Phase retrieval with application to optical imaging: a contemporary overview. IEEE Signal Processing Magazine, 32(3):87–109, 2015.

Sur, P. and Candès, E. J. A modern maximum-likelihood theory for high-dimensional logistic regression. Proceedings of the National Academy of Sciences, 116(29):14516–14525, 2019.

Takeuchi, K. Rigorous dynamics of expectation-propagation-based signal recovery from unitarily invariant measurements. IEEE Transactions on Information Theory, 66(1):368–386, 2020.

Takeuchi, K. Bayes-optimal convolutional AMP. IEEE Transactions on Information Theory, 67(7):4405–4428, 2021a.

Takeuchi, K. On the convergence of Orthogonal/Vector AMP: Long-memory message-passing strategy. arXiv:2111.05522, 2021b.

Tian, F., Liu, L., and Chen, X. Generalized memory approximate message passing. arXiv:2110.06069, 2021.

Tibshirani, R. Regression shrinkage and selection with the Lasso. J. Royal. Statist. Soc, B58:267–288, 1996.

Vila, J. P. and Schniter, P. An empirical-Bayes approach to recovering linearly constrained non-negative sparse signals. IEEE Transactions on Signal Processing, 62(18): 4689–4703, 2014.

Zhong, X., Su, C., and Fan, Z. Approximate Message Passing for orthogonally invariant ensembles: Multivariate non-linearities and spectral initialization. arXiv:2110.02318, 2021.

A. Background on Rectangular Free Cumulants

Let X be a random variable of finite moments of all orders, and denote its even moments by $m_{2k} = \mathbb{E}\{X^{2k}\}$. In this paper, X^2 represents either the empirical eigenvalue distribution of $AA^T \in \mathbb{R}^{n \times n}$, or its limit law μ^2 (in the latter case, the moments and rectangular free cumulants are denoted by $\{m_{2k}\}_{k \geq 1}$ and $\{\kappa_{2k}\}_{k \geq 1}$, respectively). The rectangular free cumulants $\{\kappa_{2k}\}_{k \geq 1}$ of X are defined recursively by the moment-cumulant relations (cf. Section 3 of (Benaych-Georges, 2009))

$$m_{2k} = \sum_{\substack{\mathcal{S} \in NC^0(2k) \\ \min \mathcal{S} \text{ is odd}}} \prod_{|S| \in \mathcal{S}} \prod_{\substack{S \in \mathcal{S} \\ \min S \text{ is even}}} \kappa_{|S|}; \tag{34}$$

where $NC^0(2k)$ is the set of non-crossing partitions of $\{1, \ldots, 2k\}$ such that each set $S \in \mathcal{S}$ has even cardinality. Furthermore, by exploiting the connection between the formal power series with coefficients $\{m_{2k}\}_{k \geq 1}$ and $\{\kappa_{2k}\}_{k \geq 1}$, each rectangular free cumulant κ_{2k} can be computed from $m_2, \ldots, m_{2k}$ and $\kappa_2, \ldots, \kappa_{2(k-1)}$ as (cf. Lemma 3.4 of (Benaych-Georges, 2009))

$$\kappa_{2k} = m_{2k} - [z^k] \sum_{j=1}^{k-1} \kappa_{2j} \left(z(M(z)+1)(M(z)+1)\right)^j; \tag{35}$$

where $M(z) = \sum_{k=1}^{\infty} m_{2k} z^k$ and $[z^k](q(z))$ denotes the coefficient of z^k in the polynomial $q(z)$.

In the numerical simulations of Section 4, the singular values of A are i.i.d. $\sqrt{6}\,\mathrm{Beta}(1, 2)$. Hence, for $\delta \in (0, 1)$, X has distribution $\sqrt{6}\,\mathrm{Beta}(1, 2)$ and consequently $m_{2k} = \frac{6^k}{(k+1)(2k+1)}$; for $\delta > 1$, X has distribution $\sqrt{6}\,\mathrm{Beta}(1, 2)$ w.p. $1/\delta$ and it is equal to 0 w.p. $1 - 1/\delta$, and consequently $m_{2k} = \frac{1}{\delta} \frac{6^k}{(k+1)(2k+1)}$. Then, given the moments $\{m_{2k}\}_{k \geq 1}$, the rectangular free cumulants $\{\kappa_{2k}\}_{k \geq 1}$ are computed recursively using (35).

B. Computation of Denoisers, and Implementation Details

Computation of f_t for Rademacher prior. Here, $P(X = 1) = P(X = -1) = 1/2$. Hence, (28) can be specialized as:

$$f_t(x_1, \ldots, x_t) = \mathbb{E}\{X \mid X_1 = x_1, \ldots, X_t = x_t\} = 2\, P(X = 1 \mid X_1 = x_1, \ldots, X_t = x_t) - 1. \tag{36}$$

From (13), we have that $(X_1, \ldots, X_t) = \mu_t X + (W_1, \ldots, W_t)$, with $(W_1, \ldots, W_t) \sim \mathcal{N}(0, \Sigma_t)$. Thus,

$$P(X = 1 \mid X_1 = x_1, \ldots, X_t = x_t) = \frac{\exp\left(-\frac{(x - \mu_t)^T (\Sigma_t)^{-1} (x - \mu_t)}{2}\right)}{\exp\left(-\frac{(x - \mu_t)^T (\Sigma_t)^{-1} (x - \mu_t)}{2}\right) + \exp\left(-\frac{(x + \mu_t)^T (\Sigma_t)^{-1} (x + \mu_t)}{2}\right)}; \tag{37}$$

where $x = (x_1, \ldots, x_t)^T$. (All vectors in this section, including x and μ_t, are treated as column vectors, unless otherwise mentioned.) Combining (36) and (37), we obtain

$$f_t(x_1, \ldots, x_t) = \tanh\left(\mu_t^T (\Sigma_t)^{-1} x\right). \tag{38}$$

Furthermore, the partial derivatives of f_t can be expressed in the following compact form:

$$\partial_{x_i} f_t(x_1, \ldots, x_t) = \left(1 - \tanh^2\left(\mu_t^T (\Sigma_t)^{-1} x\right)\right) \mu_t^T (\Sigma_t)^{-1} e_i; \quad \text{for } i \in [t]; \tag{39}$$

where e_i is the vector corresponding to the i-th element of the canonical basis of $\mathbb{R}^t$.

Computation of f_t for Gaussian prior. Here, $X \sim \mathcal{N}(0, 1)$. By evaluating explicitly the conditional expectation, one readily obtains that

$$f_t(x_1, \ldots, x_t) = \mathbb{E}\{X \mid X_1 = x_1, \ldots, X_t = x_t\} = \frac{\mu_t^T (\Sigma_t)^{-1} x}{1 + \mu_t^T (\Sigma_t)^{-1} \mu_t}; \tag{40}$$

which leads to the following expressions for the partial derivatives:

$$\partial_{x_i} f_t(x_1, \ldots, x_t) = \frac{\mu_t^T (\Sigma_t)^{-1} e_i}{1 + \mu_t^T (\Sigma_t)^{-1} \mu_t}; \quad \text{for } i \in [t]; \tag{41}$$

Computation of f_t for non-negative Bernoulli-Gaussian prior. Here, X is equal to 0 with probability 1 and it is distributed according to the modulus of a Gaussian with mean and variance ^{2} with probability , i.e., X $(1 \quad) \ _0 + \quad N_+(0; \ ^2)$. The parameter is taken to be $1=6$, which is close to the actual sparsity of the image given by $8645=50625$; the parameter ^{2} is taken to be $1= \ $, which gives $E f X^2 g = 1$, and the image is normalized to have unit second moment. Now, we can write

$$f_t(x_1;\dots;x_t) = E f X \ \ j \ X_1 = x_1;\dots;X_t = x_t g = \frac{E_X \ \ f X \ \ P(X_1 = x_1;\dots;X_t = x_t \ j \ X \ \ )g}{E_X \ \ f P(X_1 = x_1;\dots;X_t = x_t \ j \ X \ \ )g}; \tag{42}$$

where E_X denotes the expected value with respect X to Using that $(X_1;\dots;X_t) = \ _t X \ + (W_1;\dots;W_t)$ with $(W_1;\dots;W_t) \ \ N \ \ (0; \ _t)$, it is straightforward to compute the expectations on the RHS, which yields

$$f_t(x_1;\dots;x_t) = \frac{\left[\sqrt{2 \ ^2}\ \sqrt{\frac{b}{2a^3}} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right) + \frac{2}{a}\right]}{1 \ \ + \sqrt{a \ ^2} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right)}; \tag{43}$$

where $a = 1= \ ^2 + \ _t^T(\ _t)^{\ 1} \ _t$, $b = 2 \ _t^T(\ _t)^{\ 1}x$, $x = (x_1;\dots;x_t)$, and Erf is the error function.
To compute the derivative, we write $@_{x_i} f_t(x_1;\dots;x_t) = @_b f_t(x_1;\dots;x_t) @_{x_i} b$. Since $@_{x_i} b = 2 \ _t^T(\ _t)^{\ 1}e_i$, after some manipulations, one obtains

$$\begin{aligned} @_{x_i} f_t(x_1;\dots;x_t) = {} & \frac{2\sqrt{2 \ ^2}\left[\frac{b}{2a^2} + \sqrt{\frac{b}{2a^3}}\left(1 + \frac{b^2}{4a}\right)\exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right)\right]}{1 \ \ + \sqrt{a \ ^2} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right)} \ _t^T(\ _t)^{\ 1}e_i \\ & - \frac{\sqrt{2 \ ^2}\ \sqrt{\frac{b}{2a^3}} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right) + \frac{2}{a}}{\left(1 \ \ + \sqrt{a \ ^2} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right)\right)^2} \\ & \cdot 2\left(\frac{\sqrt{2 \ ^2}}{a} + \frac{\sqrt{b}}{4\sqrt{a^3 \ ^2}} \exp\left(\frac{b^2}{8a}\right)\left(1 + \mathrm{Erf}\left(\sqrt{\frac{b}{8a}}\right)\right)\right) \ _t^T(\ _t)^{\ 1}e_i : \end{aligned} \tag{44}$$

Computation of h_{t+1} for linear regression. In this case, we have $Y = G + W$, where $W \ \ N \ \ (0; \ ^2)$. For $t = 0$, we set $h_1(y) = y$ and consequently

$$@_y h_1(y) = 1 : \tag{45}$$

For $t > 0$, h_{t+1} is de ned as in (29). From (12), we have that $(G; R_1;\dots;R_t) \ \ N \ \ (0; \ _{t+1})$. Thus, the second conditional expectation in (29) can be expressed as

$$E f G \ j \ R_1 = r_1;\dots;R_t = r_t g = (\ _{t+1})_{[1;2:t+1]} \ \ (\ _{t+1})_{[2:t+1;2:t+1]}^{\ \ 1} r ; \tag{46}$$

where $r = (r_1;\dots;r_t)^T$. Note that $(G; R_1;\dots;R_t; Y) \ \ N \ \ (0; S_{t+2})$, where

$$S_{t+2} := \begin{pmatrix} \ _{t+1} & (\ _{t+1})_{[1;t+1;1]} \\ (\ _{t+1})_{[1;1:t+1]} & E f Y^2 g \end{pmatrix} :$$

Here, we denote by $(A)_{[i_1:i_2;j_1:j_2]}$ the submatrix obtained by taking the rows of A from i_1 to i_2 and the columns of A from j_1 to j_2 (if $i_1 = i_2$ or $j_1 = j_2$, the second index is omitted). Thus, the rst conditional expectation (29) can be expressed as

$$E f G \ j \ R_1 = r_1;\dots;R_t = r_t; Y = y g = (S_{t+2})_{[1;2:t+2]} \ \ (S_{t+2})_{[2:t+2;2:t+2]}^{\ \ 1} \begin{pmatrix} r \\ y \end{pmatrix} : \tag{47}$$

By combining (46) and (47), we obtain

$$\begin{aligned} h_{t+1}(r_1;\dots;r_t;y) &= E f G \ j \ R_1 = r_1;\dots;R_t = r_t; Y = y g \ \ E f G \ j \ R_1 = r_1;\dots;R_t = r_t g \\ &= (S_{t+2})_{[1;2:t+2]} \ \ (S_{t+2})_{[2:t+2;2:t+2]}^{\ \ 1} \begin{pmatrix} r \\ y \end{pmatrix} \ \ (\ _{t+1})_{[1;2:t+1]} \ \ (\ _{t+1})_{[2:t+1;2:t+1]}^{\ \ 1} r : \end{aligned} \tag{48}$$

Furthermore, the partial derivatives of h_{t+1} can be expressed in the following compact form:

$$\begin{aligned}
\partial_{r_i} h_{t+1}(r_1;\dots;r_t;y) &= (S_{t+2})_{[1;2:t+2]} \quad (S_{t+2})_{[2:t+2;2:t+2]}^{-1} \begin{pmatrix} e_i \\ 0 \end{pmatrix} \\
&\qquad - (\quad_{t+1})_{[1;2:t+1]} \quad (\quad_{t+1})_{[2:t+1;2:t+1]}^{-1} e_i; \quad \text{for } i \in [t]; \\
\partial_g h_{t+1}(r_1;\dots;r_t;y) &= (S_{t+2})_{[1;2:t+2]} \quad (S_{t+2})_{[2:t+2;2:t+2]}^{-1} e_{t+1}:
\end{aligned} \tag{49}$$

Computation of h_{t+1} for noiseless 1-bit compressed sensing. In this case, we have that $Y_t = \mathrm{sign}(G)$, which implies that

$$P(Y = 1 \mid G = g) = \frac{1+\mathrm{sign}(g)}{2}; \qquad P(Y = \quad 1 \mid G = g) = \frac{1 \quad \mathrm{sign}(g)}{2}: \tag{50}$$

For $t = 0$, we set $h_1(y) = y$ and consequently

$$\mathbb{E}[\partial_g h_1(Y)] = \mathbb{E}[\partial_g \mathrm{sign}(G)] = \frac{\mathbb{E}[G\,\mathrm{sign}(G)]}{\mathbb{E}[G^2]} = \sqrt{\frac{2}{\mathbb{E}[G^2]}}; \tag{51}$$

where the first equality follows from the definition of h_1 and y, and the second equality is obtained by recalling that G is Gaussian with zero mean and by applying Stein's lemma. For $t > 0$, h_{t+1} is defined as in (29). Since $(G; R_1;\dots;R_t) \sim \mathcal{N}(0;\quad_{t+1})$, the conditional distribution of G given $(R_1 = r_1;\dots;R_t = r_t)$ is $\mathcal{N}(\hat{r}_t;\hat{\tau}_t^2)$ where

$$\begin{aligned}
\hat{r}_t &= \mathbb{E}[G \mid R_1 = r_1;\dots;R_t = r_t] = (\quad_{t+1})_{[1;2:t+1]} \quad (\quad_{t+1})_{[2:t+1;2:t+1]}^{-1} r; \\
\hat{\tau}_t^2 &= \mathbb{E}[G^2] \quad (\quad_{t+1})_{[1;2:t+1]} \quad (\quad_{t+1})_{[2:t+1;2:t+1]}^{-1} (\quad_{t+1})_{[2:t+1;1]}:
\end{aligned} \tag{52}$$

We therefore have

$$\mathbb{E}[G \mid R_1 = r_1;\dots;R_t = r_t; Y = y] = \mathbb{E}[G \mid \hat{R}_t = \hat{r}_t; Y = y] = \frac{\mathbb{E}_Z \quad (\hat{r}_t + \hat{\tau}_t Z)\dfrac{1+\mathrm{sign}(y(\hat{r}_t+\hat{\tau}_t Z))}{2}}{\mathbb{E}_Z \quad \dfrac{1+\mathrm{sign}(y(\hat{r}_t+\hat{\tau}_t Z))}{2}}; \tag{53}$$

where $Z \sim \mathcal{N}(0;1)$, $\mathbb{E}_Z$ indicates that the expectation is taken over Z, and in the second line we used (50). As $\mathbb{E}[G \mid R_1 = r_1;\dots;R_t = r_t] = \hat{r}_t$, (53) readily implies that

$$h_{t+1}(r_1;\dots;r_t;y) = \frac{\hat{\tau}_t \mathbb{E}_Z \quad Z\dfrac{1+\mathrm{sign}(y(\hat{r}_t+\hat{\tau}_t Z))}{2}}{\mathbb{E}_Z \quad \dfrac{1+\mathrm{sign}(y(\hat{r}_t+\hat{\tau}_t Z))}{2}} = \frac{\hat{\tau}_t \quad \dfrac{\hat{r}_t}{\hat{\tau}_t}}{\dfrac{y+1}{2} \quad \dfrac{\hat{r}_t}{\hat{\tau}_t}}; \tag{54}$$

where $\phi(x) = \frac{1}{\sqrt{2\pi}}\exp(\quad x^2/2)$, and $\Phi(x) = \int_{-1}^{x} \quad \phi(t)\,dt$. The second equality in (54) is obtained by computing the expectations and using the fact that $y \in \{\quad 1; 1\}$.

For the partial derivatives of h_{t+1}, we note that $\partial_{r_i} h_{t+1} = \frac{\partial h_{t+1}}{\partial \hat{r}_t}\frac{\partial \hat{r}_t}{\partial r_i}$, and

$$\frac{\partial \hat{r}_t}{\partial r_i} = (\quad_{t+1})_{[1;2:t+1]} \quad (\quad_{t+1})_{[2:t+1;2:t+1]}^{-1} e_i; \quad i \in [t]:$$

Thus, by using (54) to compute $\frac{\partial h_{t+1}}{\partial \hat{r}_t}$, after some manipulations, we have that

$$\partial_{r_i} h_{t+1}(r_1;\dots;r_t;y) = \frac{\dfrac{\hat{r}_t}{\hat{\tau}_t}^2 \quad \dfrac{\hat{r}_t}{\hat{\tau}_t}\dfrac{\hat{r}_t}{\hat{\tau}_t}\dfrac{y+1}{2} \quad \dfrac{\hat{r}_t}{\hat{\tau}_t}}{\dfrac{y+1}{2} \quad \dfrac{\hat{r}_t}{\hat{\tau}_t}^2} \quad (\quad_{t+1})_{[1;2:t+1]} \quad (\quad_{t+1})_{[2:t+1;2:t+1]}^{-1} e_i: \tag{55}$$

(a) $= 0.7$ (b) $= 0.4$

Figure 6: Additional numerical results for linear regression with a Rademacher prior: normalized squared correlation vs. noise level (on the left) and vs. aspect ratio (on the right).

Figure 7: Comparison between the normalized squared correlation of RI-GAMP and of a subgradient method for the recovery of an image from 1-bit measurements of its wavelet transform.

Finally, for the partial derivative with respect to g, we have

$$\begin{aligned} \mathbb{E}\{\partial_g h_{t+1}(R_1,\dots,R_t;Y)\} &= \mathbb{E}\left[h_{t+1}(R_1,\dots,R_t;Y)\,\frac{\mathbb{E}\{G \mid R_1,\dots,R_t;Y\} - \mathbb{E}\{G \mid R_1,\dots,R_t\}}{\mathrm{Var}(G \mid R_1,\dots,R_t)}\right] \\ &= \frac{1}{\hat{\sigma}_t^2}\mathbb{E}\left[h_{t+1}(R_1,\dots,R_t;Y)^2\right]; \end{aligned} \tag{56}$$

where the first equality follows from Stein's lemma (see e.g. (A.8) of (Mondelli & Venkataramanan, 2021a)), and in the second equality we use the definition (29) of h_{t+1} and that $\mathrm{Var}(G \mid R_1,\dots,R_t) = \hat{\sigma}_t^2$ (see (52)).

Implementation details. RI-GAMP. We use consistent empirical estimates for the state evolution parameters required for the posterior mean denoisers and their partial derivatives. These estimates are computed as described on p.6. To estimate the first row and column of Σ_{t+1}, we use the definition (28) and the tower property of conditional expectation:

$$(\Sigma_{t+1})_{1;i+1} = \mathbb{E}\{X f_i(X_1,\dots,X_i)\} = \mathbb{E}\left[f_i(X_1,\dots,X_i)^2\right]; \quad i \in [t]. \tag{57}$$

Therefore, one can consistently estimate $(\Sigma_{t+1})_{1;i+1}$ via $\|f_i(x_1,\dots,x_i)\|^2/d$. The partial derivatives for the matrix Σ_{t+1} (defined in (6)) are computed using (39) (for Rademacher prior) and (41) (for Gaussian prior). The partial derivatives for the matrix Φ_{t+1} (again, defined in (6)) are computed using (49) (for linear regression) and (55)-(56) (for 1-bit compressed sensing). For the quantity $\mathbb{E}\{\partial_g h_1(y)\}$, we use the deterministic limit $\mathbb{E}\{\partial_g h_1(Y)\}$ which is given in the two settings by (45) and (51), respectively.

VAMP. Our implementation of VAMP is based on Algorithm 2 in (Schniter et al., 2016) and the corresponding state evolution is derived from (Pandit et al., 2020).[2] To ensure numerical stability, we clipped the γ_i and η_i in Algorithm 2 to lie in $[\mathrm{tol}, 1-\mathrm{tol}]$, where $\mathrm{tol} = 10^{-11}$.

C. Additional Numerical Results

In Figure 6, we provide additional numerical results for the model of linear regression with a Rademacher signal prior: on the left, we plot the normalized squared correlation as a function of δ, for $= 0.7$; and on the right, we plot the same metric as function of δ, for $= 0.4$. The results showcase a similar qualitative behavior as discussed in Section 4: the performance of RI-GAMP is close to that of VAMP, except when approaching the phase transition for exact recovery ($\delta \approx 0.2$ in Figure 6a); and RI-GAMP significantly improves upon algorithms that do not take into account the signal prior (LMMSE) or the spectrum of the noise (Gauss AMP).

[2] See also the code available at https://sourceforge.net/projects/gampmatlab/ .