Recent Advances in Approximate Message Passing

Phil Schniter

The Ohio State University

Supported in part by NSF grant CCF-1716388.

July 5, 2019

Overview

1. Linear Regression
2. Approximate Message Passing (AMP)
3. Vector AMP (VAMP)
4. Unfolding AMP and VAMP into Deep Neural Networks
5. Extensions: GLMs, Parameter Learning, Bilinear Problems

Outline

1. Linear Regression
2. Approximate Message Passing (AMP)
3. Vector AMP (VAMP)
4. Unfolding AMP and VAMP into Deep Neural Networks
5. Extensions: GLMs, Parameter Learning, Bilinear Problems

The Linear Regression Problem

Consider the following linear regression problem:

Recover $\boldsymbol{x}_o$ from

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \boldsymbol{w} \quad \text{with} \quad \begin{cases} \boldsymbol{x}_o \in \mathbb{R}^n & \text{unknown signal} \\ \boldsymbol{A} \in \mathbb{R}^{m \times n} & \text{known linear operator} \\ \boldsymbol{w} \in \mathbb{R}^m & \text{white Gaussian noise.} \end{cases}$$

Typical methodologies:

1. Optimization (or MAP estimation):

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \frac{1}{2} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + R(\boldsymbol{x}) \right\}$$

2. Approximate MMSE:

$$\widehat{\boldsymbol{x}} \approx \mathbb{E}\{\boldsymbol{x}|\boldsymbol{y}\} \text{ for } \boldsymbol{x} \sim p(\boldsymbol{x}), \quad \boldsymbol{y}|\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \nu_w \boldsymbol{I})$$

3. Plug-and-play:[1] iteratively apply a denoising algorithm like BM3D

4. Train a deep network to recover $\boldsymbol{x}_o$ from $\boldsymbol{y}$.

[1] Venkatakrishnan,Bouman,Wohlberg'13

Outline

1 Linear Regression

2 Approximate Message Passing (AMP)

3 Vector AMP (VAMP)

4 Unfolding AMP and VAMP into Deep Neural Networks

5 Extensions: GLMs, Parameter Learning, Bilinear Problems

The AMP Methodology

- All of the aforementioned methodologies can be addressed using the Approximate Message Passing (AMP) framework.

- AMP tackles these problems via iterative denoising.
  - We will write the iteration-t denoiser as $\boldsymbol{\eta}^t(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$.

- Each method defines the denoiser $\boldsymbol{\eta}^t(\cdot)$ differently:
  - Optimization: $\boldsymbol{\eta}^t(\boldsymbol{r}) = \arg\min_{\boldsymbol{x}} \{R(\boldsymbol{x}) + \frac{1}{2\nu^t}\|\boldsymbol{x} - \boldsymbol{r}\|_2^2\} \triangleq$ "$\text{prox}_{R\nu^t}(\boldsymbol{r})$"
  - MMSE: $\boldsymbol{\eta}^t(\boldsymbol{r}) = \mathbb{E}\left\{\boldsymbol{x} \,\middle|\, \boldsymbol{r} = \boldsymbol{x} + \mathcal{N}(\boldsymbol{0}, \nu^t)\right\}$
  - Plug-and-play: $\boldsymbol{\eta}^t(\boldsymbol{r}) = \text{BM3D}(\boldsymbol{r}, \nu^t)$
  - Deep network: $\boldsymbol{\eta}^t(\boldsymbol{r})$ is learned from training data.

The AMP Algorithm

initialize $\widehat{\boldsymbol{x}}^0 = \mathbf{0}$, $\boldsymbol{v}^{-1} = \mathbf{0}$

for $t = 0, 1, 2, \ldots$

$\boldsymbol{v}^t = \boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}^t + \frac{N}{M}\boldsymbol{v}^{t-1}\,\mathrm{div}\big(\boldsymbol{\eta}^{t-1}(\widehat{\boldsymbol{x}}^{t-1} + \boldsymbol{A}^\mathsf{T}\widehat{\boldsymbol{v}}^{t-1})\big)$ corrected residual

$\widehat{\boldsymbol{x}}^{t+1} = \boldsymbol{\eta}^t(\widehat{\boldsymbol{x}}^t + \boldsymbol{A}^\mathsf{T}\boldsymbol{v}^t)$ denoising

where

$$\mathrm{div}\big(\boldsymbol{\eta}^t(\boldsymbol{r})\big) \triangleq \frac{1}{n}\mathrm{tr}\left(\frac{\partial \boldsymbol{\eta}^t(\boldsymbol{r})}{\partial \boldsymbol{r}}\right) \quad \text{"divergence."}$$

Note:

- Original version proposed by Donoho, Maleki, and Montanari in 2009.
  - They considered "scalar" denoisers, such that $[\boldsymbol{\eta}^t(\boldsymbol{r})]_j = \eta^t(r_j)\ \forall j$
  - For scalar denoisers, $\mathrm{div}\big(\boldsymbol{\eta}^t(\boldsymbol{r})\big) = \frac{1}{n}\sum_{j=1}^n \eta^{t\prime}(r_j)$
- Can be recognized as iterative shrinkage/thresholding[2] plus "Onsager correction."
- Can be derived using Gaussian & Taylor-series approximations of loopy belief-propagation (hence "AMP").

[2] Chambolle,DeVore,Lee,Lucier'98

AMP's Denoising Property

Original AMP Assumptions

- $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is drawn i.i.d. Gaussian
- $m, n \to \infty$ s.t. $\frac{m}{n} \to \delta \in (0, \infty)$ ... "large-system limit"
- $[\boldsymbol{\eta}^t(\boldsymbol{r})]_j = \eta^t(r_j)$ with Lipschitz $\eta(\cdot)$ ... "scalar denoising"

Under these assumptions, the denoiser's input $\boldsymbol{r}^t \triangleq \widehat{\boldsymbol{x}}^t + \boldsymbol{A}^\mathsf{T}\boldsymbol{v}^t$ obeys[3]

$$\boxed{r_j^t = x_{o,j} + \mathcal{N}(0, \nu_r^t)}$$

- That is, $\boldsymbol{r}^t$ is a Gaussian-noise corrupted version of the true signal $\boldsymbol{x}_o$.
- It should now be clear why we think of $\boldsymbol{\eta}^t(\cdot)$ as a "denoiser."

Furthermore, the effective noise variance can be consistently estimated:

$$\widehat{\nu}_r^t \triangleq \frac{1}{m}\|\boldsymbol{v}^t\|^2 \quad \longrightarrow \quad \nu_r^t.$$

[3] Bayati,Montanari'11

AMP's State Evolution

- Assume that the measurements $\boldsymbol{y}$ were generated via

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \nu_w \boldsymbol{I})$$

where $\boldsymbol{x}_o$ empirically converges to some random variable X_o as $n \to \infty$.

- Define the iteration-t mean-squared error (MSE)

$$\mathcal{E}^t \triangleq \tfrac{1}{n}\|\widehat{\boldsymbol{x}}^t - \boldsymbol{x}_o\|^2.$$

- Under above assumptions, AMP obeys the following state evolution (SE):[4]

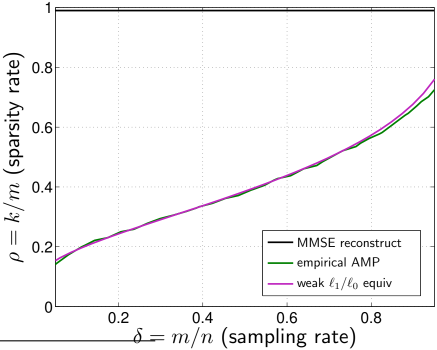
$$\begin{aligned}
&\text{for } t = 0, 1, 2, \ldots \\
&\quad \nu_r^t = \nu_w + \tfrac{n}{m}\mathcal{E}^t \\
&\quad \mathcal{E}^{t+1} = \mathbb{E}\left\{\left[\eta^t\left(X_o + \mathcal{N}(0, \nu_r^t)\right) - X_o\right]^2\right\}
\end{aligned}$$

[4] Bayati, Montanari'11

Achievability Analysis via the AMP SE

- AMP's SE can be applied to analyze achievability in various problems.
- E.g., it yields a closed-form expression[5] for the sparsity/sampling region where ℓ_1-penalized regression is equivalent to ℓ_0-penalized regression:

$$\rho(\delta) = \max_{c>0} \frac{1 - 2\delta^{-1}[(1+c^2)\Phi(-c) - c\phi(c)]}{1 + c^2 - 2[(1+c^2)\Phi(-c) - c\phi(c)]},$$



[5] Donoho, Maleki, Montanari'09

MMSE Optimality of AMP

- Now suppose that the AMP Assumptions hold, and that

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \nu_w \boldsymbol{I}),$$

where the elements of $\boldsymbol{x}_o$ are i.i.d. draws of some random variable X_o.

- Suppose also that $\eta^t(\cdot)$ is the MMSE denoiser, i.e.,

$$\eta^t(R) = \mathbb{E}\left\{ X_o \,\middle|\, R = X_o + \mathcal{N}(0, \nu_r^t) \right\}$$

- Then, if the state evolution has a unique fixed point, the MSE of $\widehat{\boldsymbol{x}}^t$ converges[6] to the replica prediction of the MMSE as $t \to \infty$.

- Under the AMP Assumptions, the replica prediction of the MMSE was shown to be correct.[7][8]

---

[6] Bayati,Montanari'11, [7] Reeves,Pfister'16, [8] Barbier,Dia,Macris,Krzakala'16

Universality of AMP State Evolution

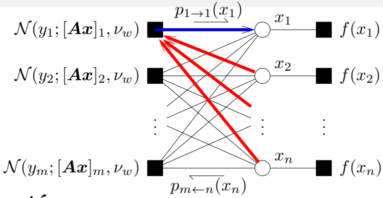
- Until now, it was assumed that $\boldsymbol{A}$ is drawn i.i.d. Gaussian.
- The state evolution also holds when $\boldsymbol{A}$ is drawn from i.i.d. A_{ij} such that

$$\begin{aligned}\mathbb{E}\{A_{ij}\} &= 0\\ \mathbb{E}\{A_{ij}^2\} &= 1/m\\ \mathbb{E}\{A_{ij}^6\} &= C/m \text{ for some fixed } C > 0.\end{aligned}$$

  often abbreviated as "sub-Gaussian A_{ij}."
- The proof[9] assumes polynomial scalar denoising $\eta^t(\cdot)$ of bounded order.

[9] Bayati,Lelarge,Montanari'15

Deriving AMP via Loopy BP (e.g., sum-product alg)



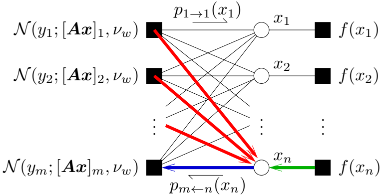
1. Message from y_i node to x_j node:

$$p_{i\to j}(x_j) \propto \int_{\{x_l\}_{l\neq j}} \mathcal{N}\big(y_i; \overbrace{\textstyle\sum_l a_{il}x_l}^{\approx \mathcal{N} \text{ via CLT}}, \nu_w\big) \textstyle\prod_{l\neq j} p_{i\leftarrow l}(x_l)$$

$$\approx \int_{z_i} \mathcal{N}(y_i; z_i, \nu_w)\, \mathcal{N}\big(z_i; \widehat{z}_i(x_j), \nu_i^z(x_j)\big) \sim \mathcal{N}$$

To compute $\widehat{z}_i(x_j), \nu_i^z(x_j)$, the means and variances of $\{p_{i\leftarrow l}\}_{l\neq j}$ suffice, implying Gaussian message passing, similar to expectation-propagation. Remaining problem: we have $2mn$ messages to compute (too many!).

2. Exploiting similarity among the messages $\{p_{i\leftarrow j}\}_{i=1}^m$, AMP employs a Taylor-series approximation of their difference whose error vanishes as $m\to\infty$ for dense $\boldsymbol{A}$ (and similar for $\{p_{i\leftarrow j}\}_{j=1}^n$ as $n\to\infty$). Finally, need to compute only $O(m+n)$ messages!



Understanding AMP

- The belief-propagation derivation of AMP provides very little insight!
  - Loopy BP is suboptimal, even if implemented exactly
  - The i.i.d. property of $\boldsymbol{A}$ is never used in the derivation
- And the rigorous proofs of AMP's state evolution are very technical!
- As a middle ground, we suggest an alternate derivation that gives insight into how and why AMP works.
  - Based on the idea of "first-order cancellation"
  - We will assume equiprobable Bernoulli $a_{ij} \in \pm 1/\sqrt{m}$ and polynomial $\eta(\cdot)$

AMP as First-Order Cancellation

Recall the AMP recursion:

$$\begin{aligned} \boldsymbol{v}^t &= \boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}^t + \tfrac{n}{m}\boldsymbol{v}^{t-1}\,\mathrm{div}\big(\eta(\boldsymbol{r}^{t-1})\big) \\ \widehat{\boldsymbol{x}}^{t+1} &= \eta(\underbrace{\widehat{\boldsymbol{x}}^t + \boldsymbol{A}^\mathsf{T}\boldsymbol{v}^t}_{\triangleq\, \boldsymbol{r}^t}) \end{aligned}$$

Notice that

$$\begin{aligned} [\boldsymbol{A}\widehat{\boldsymbol{x}}^t]_i &= \boldsymbol{a}_i^\mathsf{T}\eta\big(\widehat{\boldsymbol{x}}^{t-1} + \textstyle\sum_l \boldsymbol{a}_l v_k^{t-1}\big) \quad \text{where } \boldsymbol{a}_i^\mathsf{T} \text{ is the } i\text{th row of } \boldsymbol{A} \\ &= \boldsymbol{a}_i^\mathsf{T}\eta\big(\underbrace{\widehat{\boldsymbol{x}}^{t-1} + \textstyle\sum_{l\neq i} \boldsymbol{a}_l v_l^{t-1}}_{\triangleq\, \boldsymbol{r}_i^{t-1}\text{, which removes the direct contribution of } \boldsymbol{a}_i \text{ from } \boldsymbol{r}^{t-1}} + \boldsymbol{a}_i v_i^{t-1}\big) \\ &= \boldsymbol{a}_i^\mathsf{T}\Big[\eta(\boldsymbol{r}_i^{t-1}) + \frac{\partial\eta}{\partial\boldsymbol{r}}(\boldsymbol{r}_i^{t-1})\boldsymbol{a}_i v_i^{t-1} + O(1/m)\Big] \quad \text{using a Taylor expansion} \\ &= \boldsymbol{a}_i^\mathsf{T}\eta(\boldsymbol{r}_i^{t-1}) + v_i^{t-1}\textstyle\sum_j a_{ij}^2 \eta'(r_{ij}^{t-1}) + O(1/\sqrt{m}) \\ &= \boldsymbol{a}_i^\mathsf{T}\eta(\boldsymbol{r}_i^{t-1}) + \tfrac{n}{m}v_i^{t-1}\underbrace{\tfrac{1}{n}\textstyle\sum_j \eta'(r_{ij}^{t-1})}_{\mathrm{div}(\eta(\boldsymbol{r}_i^{t-1}))} + O(1/\sqrt{m}) \quad \text{since } a_{ij}^2 = 1/m \ \forall ij \end{aligned}$$

which uncovers the Onsager correction.

AMP as First-Order Cancellation (cont.)

Now use $[\boldsymbol{A}\widehat{\boldsymbol{x}}^t]_i$ to study jth component of denoiser input error $\boldsymbol{e}^t \triangleq \boldsymbol{r}^t - \boldsymbol{x}_o$:

$$e_j^t = \sum_i a_{ij} \sum_{l \neq j} a_{il} \left[x_{o,l} - \eta(r_{il}^{t-1})\right] + \sum_i a_{ij} w_i + \sum_i a_{ij} \left[\tfrac{n}{m} v_i^{t-1} \operatorname{div}\big(\eta(\boldsymbol{r}^{t-1})\big) - \tfrac{n}{m} v_i^{t-1} \operatorname{div}\big(\eta(\boldsymbol{r}_i^{t-1})\big)\right] + O(1/\sqrt{m})$$

where the divergence difference can be absorbed into the $O(1/\sqrt{m})$ term...

$$= \underbrace{\sum_i a_{ij} \sum_{l \neq j} a_{il} \underbrace{\left[x_{o,l} - \eta(r_{il}^{t-1})\right]}_{\triangleq\, \epsilon_{il}}}_{\sim \mathcal{N}\left(0, \frac{1}{m^2} \sum_i \sum_{l \neq j} (\epsilon_{il}^t)^2\right)} + \underbrace{\sum_i a_{ij} w_i}_{\sim \mathcal{N}\left(0, \frac{1}{m} \sum_i w_i^2\right)} + O(1/\sqrt{m})$$

using the CLT and assuming independence of $\{a_{il}\}_{l=1}^n$ and $\{r_{il}^{t-1}\}_{l=1}^n$

$$\sim \mathcal{N}\left(0, \tfrac{n}{m} \mathcal{E}^{(t)} + \nu_w\right) + O(1/\sqrt{m})$$ ...the AMP state evolution

where $\mathcal{E}^{(t)} \triangleq \frac{1}{n} \sum_{j=1}^n \left[x_{o,j} - \widehat{x}_j^{(t)}\right]^2$ and $\nu_w \triangleq \frac{1}{m} \sum_{i=1}^m w_i^2$

AMP with Non-Separable Denoisers

- Until now, we have focused on separable denoisers, i.e., $[\boldsymbol{\eta}^t(\boldsymbol{r})]_j = \eta^t(r_j)\ \forall j$
- Can we use sophisticated non-separable $\boldsymbol{\eta}(\cdot)$ with AMP?
- Yes! Many examples. . .
  - Markov chain,[10] Markov field,[12] Markov tree,[12] denoisers in 2010
  - Blockwise & TV denoising considered by Donoho, Johnstone, Montanari in 2011
  - BM3D denoising considered by Metzler, Maleki, Baraniuk in 2015
- Rigorous state-evolution proven by Berthier, Montanari, Nguyen in 2017.
  - Assumes $\boldsymbol{A}$ drawn i.i.d. Gaussian
  - Assumes $\boldsymbol{\eta}$ is Lipschitz and "convergent under Gaussian inputs"

[10] S'10, [11] Som,S'11, [12] Som,S'12

AMP at Large but Finite Dimensions

- Until now, we have focused on the large-system limit $m, n \to \infty$ with $m/n \to \delta \in (0, \infty)$
- The non-asymptotic case was analyzed by Rush and Venkataramanan.[13]
- They showed that probability of ϵ-deviation between the finite and limiting SE falls exponentially in m, as long as the number of iterations $t < o(\frac{\log n}{\log \log n})$

[13] Rush,Venkataramanan'18

AMP Summary: The good, the bad, and the ugly

The good:

- With large i.i.d. sub-Gaussian $\boldsymbol{A}$, AMP is rigorously characterized by a scalar state-evolution whose fixed points, when unique, are MMSE optimal under proper choice of denoiser.
- Empirically, AMP behaves well with many other "sufficiently random" $\boldsymbol{A}$ (e.g., randomly sub-sampled Fourier $\boldsymbol{A}$ & i.i.d. sparse $\boldsymbol{x}$).

The bad:

- With general $\boldsymbol{A}$, AMP gives no guarantees.

The ugly:

- With some $\boldsymbol{A}$, AMP may fail to converge! (e.g., ill-conditioned or non-zero-mean $\boldsymbol{A}$)

Outline

1. Linear Regression
2. Approximate Message Passing (AMP)
3. Vector AMP (VAMP)
4. Unfolding AMP and VAMP into Deep Neural Networks
5. Extensions: GLMs, Parameter Learning, Bilinear Problems

Vector AMP (VAMP)

- Recall goal is linear regression: Recover $\boldsymbol{x}_o$ from $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_w)$.
  - Now it will be easier to work with inverse variances, i.e., precisions

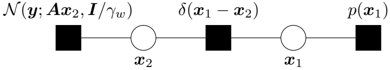
- VAMP is like AMP in many ways, but supports a larger class of random matrices.

- VAMP yields a precise analysis for right-orthogonally invariant $\boldsymbol{A}$:

$$\text{svd}(\boldsymbol{A}) = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{\mathsf{T}} \quad \text{for} \quad \begin{cases} \boldsymbol{U}\text{: deterministic orthogonal} \\ \boldsymbol{S}\text{: deterministic diagonal} \\ \boldsymbol{V}\text{: “Haar;” uniform on set of orthogonal matrices} \end{cases}$$

  of which i.i.d. Gaussian is a special case.

- Can be derived as a form of message passing on a vector-valued factor graph.



VAMP: The Algorithm

With SVD $\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})\boldsymbol{V}^\mathsf{T}$, damping $\zeta \in (0,1]$, and Lipschitz $\boldsymbol{\eta}_1^t(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$.

Initialize $\boldsymbol{r}_1, \gamma_1$.

For $t = 1, 2, 3, \ldots$

$\widehat{\boldsymbol{x}}_1 \leftarrow \boldsymbol{\eta}_1^t(\boldsymbol{r}_1)$ — denoising of $\boldsymbol{r}_1 = \boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_1)$

$\xi_1 \leftarrow \gamma_1 / \operatorname{div}\big(\boldsymbol{\eta}_1^t(\boldsymbol{r}_1)\big)$

$\boldsymbol{r}_2 \leftarrow (\xi_1 \widehat{\boldsymbol{x}}_1 - \gamma_1 \boldsymbol{r}_1)/(\xi_1 - \gamma_1)$ — Onsager correction

$\gamma_2 \leftarrow \xi_1 - \gamma_1$

$\widehat{\boldsymbol{x}}_2 \leftarrow \boldsymbol{\eta}_2(\boldsymbol{r}_2; \gamma_2)$ — LMMSE estimate of $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{r}_2, \boldsymbol{I}/\gamma_2)$ from $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_w)$

$\xi_2 \leftarrow \gamma_2 / \operatorname{div}\big(\boldsymbol{\eta}_2(\boldsymbol{r}_2; \gamma_2)\big)$

$\boldsymbol{r}_1 \leftarrow \zeta(\xi_2 \widehat{\boldsymbol{x}}_2 - \gamma_2 \boldsymbol{r}_2)/(\xi_2 - \gamma_2) + (1-\zeta)\boldsymbol{r}_1$ — Onsager correction

$\gamma_1 \leftarrow \zeta(\xi_2 - \gamma_2) + (1-\zeta)\gamma_1$ — damping

where

$$\begin{aligned}
\boldsymbol{\eta}_2(\boldsymbol{r}_2; \gamma_2) &= (\gamma_w \boldsymbol{A}^\mathsf{T}\boldsymbol{A} + \gamma_2 \boldsymbol{I})^{-1}(\gamma_w \boldsymbol{A}^\mathsf{T}\boldsymbol{y} + \gamma_2 \boldsymbol{r}_2) \\
&= \boldsymbol{V}\left(\gamma_w \,\mathrm{Diag}(\boldsymbol{s})^2 + \gamma_2 \boldsymbol{I}\right)^{-1}\left(\gamma_w \,\mathrm{Diag}(\boldsymbol{s})\boldsymbol{U}^\mathsf{T}\boldsymbol{y} + \gamma_2 \boldsymbol{V}^\mathsf{T}\boldsymbol{r}_2\right) \\
\xi_2 &= \left[\tfrac{1}{n}\textstyle\sum_{j=1}^n (\gamma_w s_j^2 + \gamma_2)^{-1}\right]^{-1}
\end{aligned}$$

two mat-vec mults per iteration!

VAMP's Denoising Property

Original VAMP Assumptions

- $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is right-orthogonally invariant
- $m,n \to \infty$ s.t. $m/n \to \delta \in (0,\infty)$ ... "large-system limit"
- $[\boldsymbol{\eta}_1^t(\boldsymbol{r})]_j = \eta_1^t(r_j)$ with Lipschitz $\eta_1^t(\cdot)$ ... "separable denoising"

Under Assumption 2, the elements of the denoiser's input $\boldsymbol{r}_1^t$ obey[14]

$$\boxed{r_{1,j}^t = x_{o,j} + \mathcal{N}(0,\nu_1^t)}$$

- That is, $\boldsymbol{r}_1^t$ is a Gaussian-noise corrupted version of the true signal $\boldsymbol{x}_o$.
- As with AMP, we can interpret $\boldsymbol{\eta}_1(\cdot)$ as a "denoiser."

[14] Rangan,S,Fletcher'16

VAMP's State Evolution

Assume empirical convergence of $\{s_j\} \to S$ and $\{(r^0_{1,j}, x_{o,j})\} \to (R^0_1, X_o)$, and define

$$\mathcal{E}_i^t \triangleq \tfrac{1}{n}\|\widehat{\boldsymbol{x}}_i^t - \boldsymbol{x}_o\|^2 \text{ for } i = 1, 2.$$

Then under the VAMP Assumptions, VAMP obeys the following state-evolution:

for $t = 0, 1, 2, \ldots$

$$\mathcal{E}_1^t = \mathbb{E}\left\{\left[\eta_1^t\big(X_o + \mathcal{N}(0, \nu_1^t)\big) - X_o\right]^2\right\} \qquad \text{MSE}$$

$$\alpha_1^t = \mathbb{E}\left\{\eta_1^{t\prime}(X_o + \mathcal{N}(0, \nu_1^t))\right\} \qquad \text{divergence}$$

$$\gamma_2^t = \gamma_1^t \tfrac{1-\alpha_1^t}{\alpha_1^t}, \quad \nu_2^t = \tfrac{1}{(1-\alpha_1^t)^2}\left[\mathcal{E}_1^t - \left(\alpha_1^t\right)^2 \nu_1^t\right]$$

$$\mathcal{E}_2^t = \mathbb{E}\left\{\left[\gamma_w S^2 + \gamma_2^t\right]^{-1}\right\} \qquad \text{MSE}$$

$$\alpha_2^t = \gamma_2^t\,\mathbb{E}\left\{\left[\gamma_w S^2 + \gamma_2^t\right]^{-1}\right\} \qquad \text{divergence}$$

$$\gamma_1^{t+1} = \gamma_2^t \tfrac{1-\alpha_2^t}{\alpha_2^t}, \quad \nu_1^{t+1} = \tfrac{1}{(1-\alpha_2^t)^2}\left[\mathcal{E}_2^t - \left(\alpha_2^t\right)^2 \nu_2^t\right]$$

Note: Above equations assume $\boldsymbol{\eta}_2(\cdot)$ uses true noise precision γ_w.
If not, there are more complicated expressions for $\mathcal{E}_2^t$ and α_2^t.

MMSE Optimality of VAMP

- Now suppose that the VAMP Assumptions hold, and that

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_w),$$

  where the elements of $\boldsymbol{x}_o$ are i.i.d. draws of some random variable X_o.

- Suppose also that $\eta_1^t(\cdot)$ is the MMSE denoiser, i.e.,

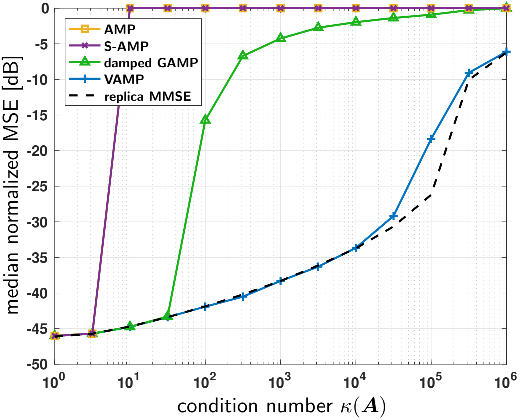
$$\eta_1^t(R_1) = \mathbb{E}\left\{X_o \,\middle|\, R_1 = X_o + \mathcal{N}(0, \nu_1^t)\right\}$$

- Then, if the state evolution has a unique fixed point, the MSE of $\widehat{\boldsymbol{x}}_1^t$ converges[15] to the replica prediction[16] of the MMSE as $t \to \infty$.

[15] Rangan,S,Fletcher'16, [16] Tulino,Caire,Verdu,Shamai'13

Experiment with MMSE Denoising

Comparison of several algorithms[17] with MMSE denoising.



$n = 1024$
$m/n = 0.5$

$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})\boldsymbol{V}^\mathsf{T}$
$\boldsymbol{U}, \boldsymbol{V} \sim$ Haar
$s_j/s_{j-1} = \phi \;\forall j$
ϕ determines $\kappa(\boldsymbol{A})$

$X_o \sim$Bernoulli-Gaussian
$\Pr\{X_0 \neq 0\} = 0.1$

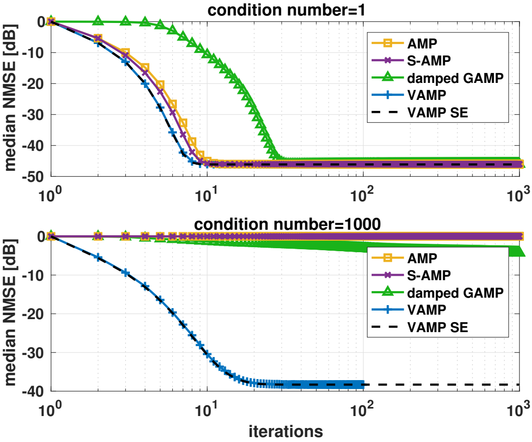
SNR$= 40$dB

VAMP achieves the replica MMSE over a wide range of condition numbers.

[17] S-AMP: Cakmak,Fleury,Winther'14, damped GAMP: Vila,S,Rangan,Krzakala,Zdeborová'15

Experiment with MMSE Denoising (cont.)

Comparison of several algorithms with priors matched to data.



$n = 1024$
$m/n = 0.5$

$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})\boldsymbol{V}^\mathsf{T}$
$\boldsymbol{U}, \boldsymbol{V} \sim$ Haar
$s_j/s_{j-1} = \phi \ \forall j$
ϕ determines $\kappa(\boldsymbol{A})$

$X_o \sim$Bernoulli-Gaussian
$\Pr\{X_0 \neq 0\} = 0.1$

SNR$=$ 40dB

VAMP is relative fast even when $\boldsymbol{A}$ is ill-conditioned.

VAMP for Optimization

- Consider the optimization problem

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \tfrac{1}{2} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|^2 + R(\boldsymbol{x}) \right\}$$

where $R(\cdot)$ is strictly convex and $\boldsymbol{A}$ is arbitrary (e.g., not necessarily RRI).

- If we choose the denoiser

$$\boldsymbol{\eta}_1^t(\boldsymbol{r}) = \arg\min_{\boldsymbol{x}} \left\{ R(\boldsymbol{x}) + \frac{\gamma_1^t}{2} \|\boldsymbol{x} - \boldsymbol{r}\|^2 \right\} = \mathsf{prox}_{R/\gamma_1^t}(\boldsymbol{r})$$

and the damping parameter

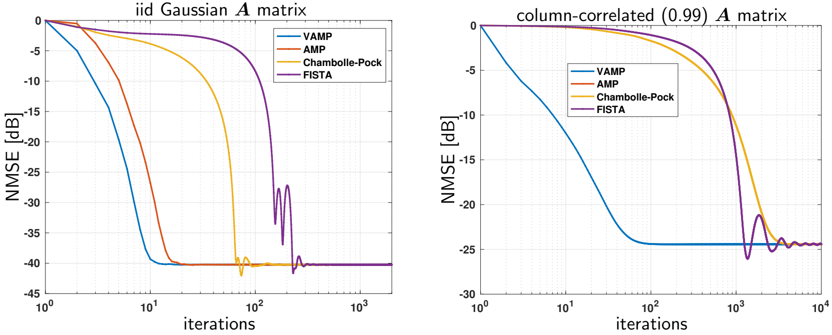
$$\zeta \le \frac{2\min\{\gamma_1, \gamma_2\}}{\gamma_1 + \gamma_2},$$

then a double-loop version of VAMP converges[18] to $\widehat{\boldsymbol{x}}$ from above.

- Furthermore, if the γ_1 and γ_2 variables are fixed over the iterations, then VAMP reduces to the Peaceman-Rachford variant of ADMM.

[18] Fletcher,Sahraee,Rangan,S'16

Example of AMP & VAMP on the LASSO Problem



Solving LASSO to reconstruct 40-sparse $\boldsymbol{x} \in \mathbb{R}^{1000}$ from noisy $\boldsymbol{y} \in \mathbb{R}^{400}$.

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_1 \right\}.$$

Deriving VAMP from EC

- Ideally, we would like to compute the exact posterior density

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{x})\ell(\boldsymbol{x};\boldsymbol{y})}{Z(\boldsymbol{y})} \quad \text{for} \quad Z(\boldsymbol{y}) \triangleq \int p(\boldsymbol{x})\ell(\boldsymbol{x};\boldsymbol{y})\,\mathrm{d}\boldsymbol{x},$$

  but the high-dimensional integral in $Z(\boldsymbol{y})$ is difficult to compute.

- We might try to circumvent $Z(\boldsymbol{y})$ through variational optimization:

$$\begin{aligned}
p(\boldsymbol{x}|\boldsymbol{y}) &= \arg\min_b D\big(b(\boldsymbol{x})\big\|p(\boldsymbol{x}|\boldsymbol{y})\big) \text{ where } D(\cdot\|\cdot) \text{ is KL divergence} \\
&= \arg\min_b \underbrace{D\big(b(\boldsymbol{x})\big\|p(\boldsymbol{x})\big) + D\big(b(\boldsymbol{x})\big\|\ell(\boldsymbol{x};\boldsymbol{y})\big) + H\big(b(\boldsymbol{x})\big)}_{\text{Gibbs free energy}} \\
&= \arg\min_{b_1,b_2,q} \underbrace{D\big(b_1(\boldsymbol{x})\big\|p(\boldsymbol{x})\big) + D\big(b_2(\boldsymbol{x})\big\|\ell(\boldsymbol{x};\boldsymbol{y})\big) + H\big(q(\boldsymbol{x})\big)}_{\triangleq J_{\text{Gibbs}}(b_1,b_2,q)} \\
&\quad \text{s.t. } b_1 = b_2 = q,
\end{aligned}$$

  but the density constraint keeps the problem difficult.

Deriving VAMP from EC (cont.)

- In expectation-consistent approximation (EC)[19], the density constraint is relaxed to moment-matching constraints:

$$p(\boldsymbol{x}|\boldsymbol{y}) \approx \underset{b_1,b_2,q}{\arg\min}\, J_{\mathsf{Gibbs}}(b_1,b_2,q)$$
$$\text{s.t. } \begin{cases} \mathbb{E}\{\boldsymbol{x}|b_1\} = \mathbb{E}\{\boldsymbol{x}|b_2\} = \mathbb{E}\{\boldsymbol{x}|q\} \\ \mathrm{tr}(\mathrm{Cov}\{\boldsymbol{x}|b_1\}) = \mathrm{tr}(\mathrm{Cov}\{\boldsymbol{x}|b_2\}) = \mathrm{tr}(\mathrm{Cov}\{\boldsymbol{x}|q\}). \end{cases}$$

- The stationary points of EC are the densities

$$\begin{aligned} b_1(\boldsymbol{x}) &\propto p(\boldsymbol{x})\mathcal{N}(\boldsymbol{x};\boldsymbol{r}_1,\boldsymbol{I}/\gamma_1) \\ b_2(\boldsymbol{x}) &\propto \ell(\boldsymbol{x};\boldsymbol{y})\mathcal{N}(\boldsymbol{x};\boldsymbol{r}_2,\boldsymbol{I}/\gamma_2) \\ q(\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{x};\widehat{\boldsymbol{x}},\boldsymbol{I}/\xi) \end{aligned} \quad \text{s.t. } \begin{cases} \mathbb{E}\{\boldsymbol{x}|b_1\} = \mathbb{E}\{\boldsymbol{x}|b_2\} = \widehat{\boldsymbol{x}} \\ \frac{1}{n}\mathrm{tr}(\mathrm{Cov}\{\boldsymbol{x}|b_1\}) = \frac{1}{n}\mathrm{tr}(\mathrm{Cov}\{\boldsymbol{x}|b_2\}) = \frac{1}{\xi} \end{cases}$$

- VAMP iteratively solves for the quantities $\boldsymbol{r}_1, \gamma_1, \boldsymbol{r}_2, \gamma_2, \widehat{\boldsymbol{x}}, \xi$ above.
  - Leads to $\boldsymbol{\eta}_1^t(\cdot)$ being the MMSE denoiser of $\boldsymbol{r}_1 = \boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_1^t)$
  - In this setting, VAMP is simply an instance of expectation propagation (EP)[20].
  - But VAMP is more general than EP, in that it allows non-MMSE denoisers $\boldsymbol{\eta}_1$.

[19] Opper,Winther'04, [20] Minka'01

Plug-and-play VAMP

- Recall the scalar denoising step of VAMP (or AMP):

$$\widehat{\boldsymbol{x}}_1 = \eta_1^t(\boldsymbol{r}_1) \quad \text{where } \boldsymbol{r}_1 = \boldsymbol{x}_o + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\gamma_1^t)$$

- For many signal classes (e.g., images), very sophisticated *non-separable* denoisers $\boldsymbol{\eta}_1(\cdot)$ have been developed (e.g., BM3D, DnCNN).

- These non-separable denoisers can be "plugged into" VAMP!

- Their divergence can be approximated via Monte Carlo[21]

$$\operatorname{div}\big(\boldsymbol{\eta}_1^t(\boldsymbol{r})\big) \approx \frac{1}{K}\sum_{k=1}^{K} \frac{\boldsymbol{p}_k^{\mathsf{T}}\big[\boldsymbol{\eta}_1^t(\boldsymbol{r}+\epsilon\boldsymbol{p}_k) - \boldsymbol{\eta}_1^t(\boldsymbol{r})\big]}{n\epsilon}$$

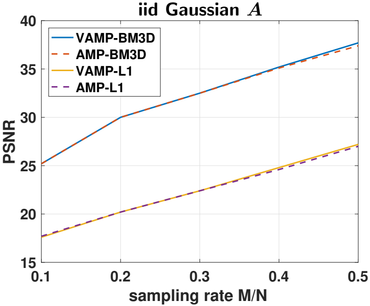
  with random vectors $\boldsymbol{p}_k \in \{\pm 1\}^n$ and small $\epsilon > 0$. Empirically, $K=1$ suffices.

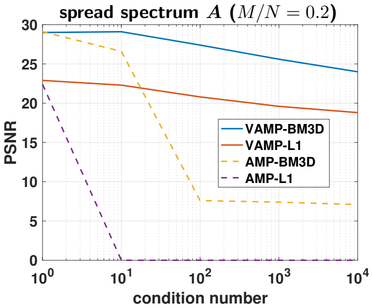
- A rigorous state-evolution has been established for plug-and-play VAMP.[22]

[21] Ramani,Blu,Unser'08, [22] Fletcher,Rangan,Sarkar,S'18

Experiment: Compressive Image Recovery with BM3D

Plug-and-play versions of VAMP and AMP behave similarly with i.i.d. Gaussian $\boldsymbol{A}$ is i.i.d., but VAMP can handle a larger class of random matrices $\boldsymbol{A}$.





Results above are averaged over 128×128 versions of

lena, barbara, boat, fingerprint, house, peppers

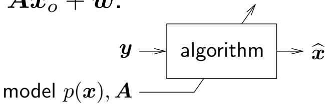
and 10 random realizations of $\boldsymbol{A}, \boldsymbol{w}$.

Outline

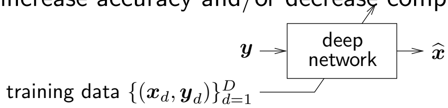
1. Linear Regression
2. Approximate Message Passing (AMP)
3. Vector AMP (VAMP)
4. Unfolding AMP and VAMP into Deep Neural Networks
5. Extensions: GLMs, Parameter Learning, Bilinear Problems

Deep learning for sparse reconstruction

- Until now we've focused on designing algorithms to recover $\boldsymbol{x}_o \sim p(\boldsymbol{x})$ from measurements $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \boldsymbol{w}$.



- What about training deep networks to predict $\boldsymbol{x}_o$ from $\boldsymbol{y}$?
  Can we increase accuracy and/or decrease computation?



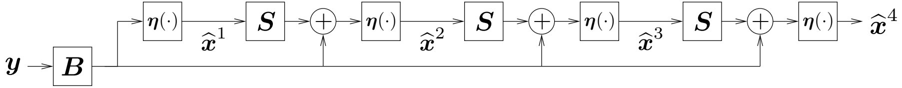
- Are there connections between these approaches?

Unfolding Algorithms into Networks

Consider, e.g., the classical sparse-reconstruction algorithm, ISTA.[23]

$$\boxed{\begin{aligned} \boldsymbol{v}^t &= \boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}^t \\ \widehat{\boldsymbol{x}}^{t+1} &= \boldsymbol{\eta}(\widehat{\boldsymbol{x}}^t + \boldsymbol{A}^\mathsf{T}\boldsymbol{v}^t) \end{aligned}} \quad \Leftrightarrow \quad \boxed{\widehat{\boldsymbol{x}}^{t+1} = \boldsymbol{\eta}(\boldsymbol{S}\widehat{\boldsymbol{x}}^t + \boldsymbol{B}\boldsymbol{y}) \text{ with } \begin{array}{l} \boldsymbol{S} \triangleq \boldsymbol{I} - \boldsymbol{A}^\mathsf{T}\boldsymbol{A} \\ \boldsymbol{B} \triangleq \boldsymbol{A}^\mathsf{T} \end{array}}$$

Gregor & LeCun[24] proposed to "unfold" it into a deep net and "learn" improved parameters using training data, yielding "learned ISTA" (LISTA):

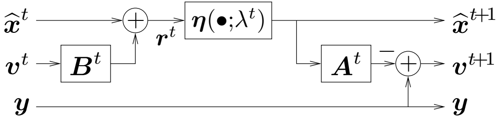


The same "unfolding & learning" idea can be used to improve AMP, yielding "learned AMP" (LAMP).[25]

[23] Chambolle,DeVore,Lee,Lucier'98. [24] Gregor,LeCun'10. [25] Borgerding,S'16.

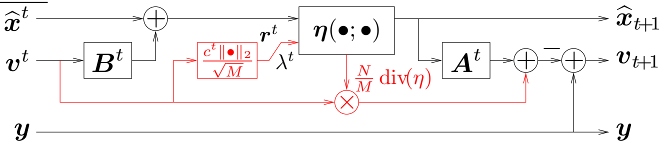
Onsager-Corrected Deep Networks

t^{th} LISTA layer:



to exploit low-rank $\boldsymbol{B}^t\boldsymbol{A}^t$ in linear stage $\boldsymbol{S}^t = \boldsymbol{I} - \boldsymbol{B}^t\boldsymbol{A}^t$.

t^{th} LAMP layer:

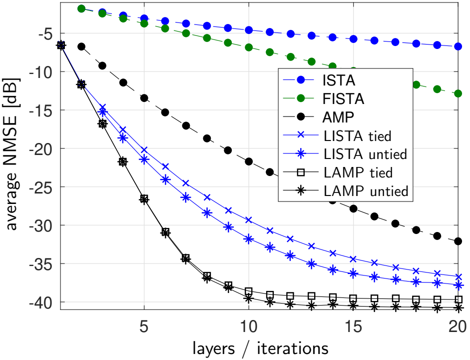


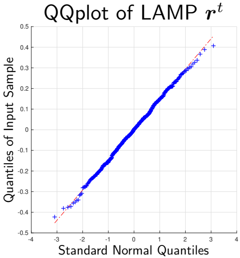
Onsager correction now aims to decouple errors across layers.

LAMP performance with soft-threshold denoising

LISTA beats AMP,FISTA,ISTA
LAMP beats LISTA

in convergence speed and asymptotic MSE.

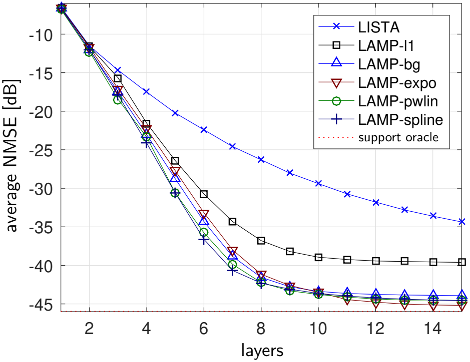




LAMP beyond soft-thresholding

So far, we used soft-thresholding to isolate the effects of Onsager correction.

What happens with more sophisticated (learned) denoisers?



Here we learned the parameters of these denoiser families:

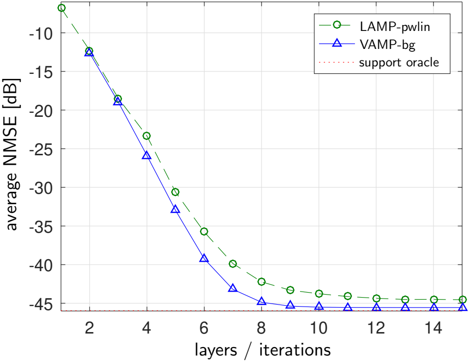
- scaled soft-thresholding
- conditional mean under BG
- Exponential kernel[26]
- Piecewise Linear[26]
- Spline[27]

Big improvement!

[26] Guo,Davies'15. [27] Kamilov,Mansour'16.

LAMP versus VAMP

How does our best Learned AMP compare to MMSE VAMP?

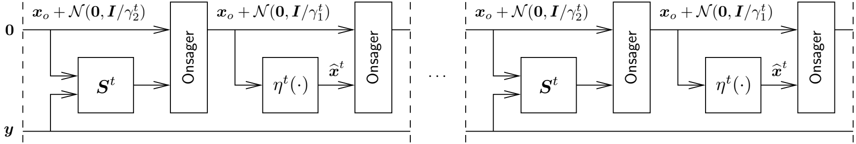


VAMP wins!

So what about "learned VAMP"?

Learned VAMP

- Suppose we unfold VAMP and learn (via backprop) the parameters $\{\boldsymbol{S}^t, \eta^t\}_{t=1}^T$ that minimize the training MSE.



- Remarkably, backpropagation learns the parameters prescribed by VAMP!

  Theory explains the deep network!

- Onsager correction decouples the design of $\{\boldsymbol{S}^t, \eta^t(\cdot)\}_{t=1}^T$:

  Layer-wise optimal $\boldsymbol{S}^t, \eta^t(\cdot)$ $\Rightarrow$ Network optimal $\{\boldsymbol{S}^t, \eta^t(\cdot)\}_{t=1}^T$

Outline

1 Linear Regression

2 Approximate Message Passing (AMP)

3 Vector AMP (VAMP)

4 Unfolding AMP and VAMP into Deep Neural Networks

5 Extensions: GLMs, Parameter Learning, Bilinear Problems

Generalized linear models

- Until now we have considered the standard linear model: $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_o + \boldsymbol{w}$.
- One may also consider the generalized linear model (GLM), where

$$\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{z}) \text{ with hidden } \boldsymbol{z} = \boldsymbol{A}\boldsymbol{x}_o$$

  which supports, e.g.,
  - $y_i = z_i + w_i$: additive, possibly non-Gaussian noise
  - $y_i = Q(z_i + w_i)$: quantization
  - $y_i = \mathrm{sgn}(z_i + w_i)$: binary classification
  - $y_i = |z_i + w_i|$: phase retrieval
  - Poisson y_i: photon-limited imaging
- For this, there is a Generalized AMP[29] with a rigorous state evolution.[30]
- There is also a Generalized VAMP[31] with a rigorous state evolution.[32]

[29] Rangan'11, [30] Javanmard,Montanari'12, [31] S,Fletcher,Rangan'16. [32] Fletcher,Rangan,S'18.

Parameter learning

- Consider inference under prior $p(\boldsymbol{x};\boldsymbol{\theta}_1)$ and likelihood $\ell(\boldsymbol{x};\boldsymbol{y},\boldsymbol{\theta}_2)$, where the hyperparameters $\boldsymbol{\theta} \triangleq [\boldsymbol{\theta}_1,\boldsymbol{\theta}_2]$ are unknown.
  - $\boldsymbol{\theta}_1$ might specify sparsity rate, or all parameters of a GMM
  - $\boldsymbol{\theta}_2$ might specify the measurement noise variance, or forward model $\boldsymbol{A}$
- EM-inspired extensions of (G)AMP and (G)VAMP that simultaneously estimate $\boldsymbol{x}$ and learn $\boldsymbol{\theta}$ from $\boldsymbol{y}$ have been developed.
  - Have rigorous state evolutions[33][34]
  - "Adaptive VAMP" yields asymptotically consistent[34] estimates of $\boldsymbol{\theta}$
- SURE-based auto-tuning AMP algorithms have also been proposed
  - for LASSO by Mousavi, Maleki, and Baraniuk
  - for parametric separable denoisers by Guo and Davies

[33] Kamilov,Rangan,Fletcher,Unser'12, [34] Fletcher,Sahraee,Rangan,S'17

Bilinear problems

- So far we have considered (generalized) linear models.
- AMP has also been applied to (generalized) *bilinear* models.
- The typical problem is to recover $\boldsymbol{B} \in \mathbb{R}^{m\times k}$ and $\boldsymbol{C} \in \mathbb{R}^{k\times n}$ from ...
  - $\begin{cases} \boldsymbol{Y} = \boldsymbol{BC} + \boldsymbol{W} \text{ (standard bilinear model)} \\ \boldsymbol{Y} \sim p(\boldsymbol{Y}|\boldsymbol{Z}) \text{ for } \boldsymbol{Z} = \boldsymbol{BC} \text{ (generalized bilinear model)} \end{cases}$
  - The case where $m, n \to \infty$ for fixed k is well understood.[35] (See Jean's talk)
  - With $m, n, k \to \infty$, algorithms work (e.g., BiGAMP[36]) but are not well understood.
- A more general bilinear problem is to recover $\boldsymbol{b} \in \mathbb{R}^k$ and $\boldsymbol{c} \in \mathbb{R}^n$ from
  - $\begin{cases} y_i = \boldsymbol{b}^\mathsf{T}\boldsymbol{A}_i\boldsymbol{c} + w_i,\ i = 1\dots m \\ y_i \sim p(y_i|z_i) \text{ for } z_i = \boldsymbol{b}^\mathsf{T}\boldsymbol{A}_i\boldsymbol{c},\ i = 1\dots m \end{cases}$ where $\{\boldsymbol{A}_i\}$ are known matrices
  - Algorithms[37] and replica analyses[38] (for $m, n, k \to \infty$ and i.i.d. $\boldsymbol{A}_i$) exist.

[35] Montanari,Venkataramanan'17, [36] Parker,S,Cevher'14, [37] Parker,S'16, [38] Schulke,S,Zdeborova'16

Conclusions

- AMP and VAMP are a computationally efficient algorithms for (generalized) linear regression.
- With large random $\boldsymbol{A}$, the ensemble behaviors of AMP and VAMP obey rigorous state evolutions whose fixed-points, when unique, agree with the replica predictions of the MMSE.
- AMP and VAMP support nonseparable (i.e., "plug-in") denoisers, also with rigorous state evolutions.
- For convex optimization problems, VAMP is provably convergent for any $\boldsymbol{A}$.
- Extensions of AMP and VAMP cover ...
  - unfolded deep networks
  - the learning of unknown prior/likelihood parameters
  - bilinear problems
- Not discussed: multilayer versions of AMP & VAMP.

References I

S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal Info. Process.*, pp. 945–948, 2013.

D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, pp. 18914–18919, Nov. 2009.

A. Chambolle, R. A. DeVore, N. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, pp. 319–335, Mar. 1998.

M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inform. Theory*, vol. 57, pp. 764–785, Feb. 2011.

G. Reeves and H. D. Pfister, "The replica-symmetric prediction for compressed sensing with Gaussian matrices is exact," in *Proc. IEEE Int. Symp. Inform. Thy.*, 2016.

J. Barbier, M. Dia, N. Macris, and F. Krzakala, "The mutual information in random linear estimation," in *Proc. Allerton Conf. Commun. Control Comput.*, pp. 625–632, 2016.

M. Bayati, M. Lelarge, and A. Montanari, "Universality in polytope phase transitions and message passing algorithms," *Ann. App. Prob.*, vol. 25, no. 2, pp. 753–822, 2015.

References II

- P. Schniter, “Turbo reconstruction of structured sparse signals,” in *Proc. Conf. Inform. Science & Syst.*, (Princeton, NJ), pp. 1–6, Mar. 2010.

- S. Som and P. Schniter, “Approximate message passing for recovery of sparse signals with Markov-random-field support structure.” *Internat. Conf. Mach. Learning—Workshop on Structured Sparsity: Learning and Inference*, (Bellevue, WA), July 2011.

- S. Som and P. Schniter, “Compressive imaging using approximate message passing and a Markov-tree prior,” *IEEE Trans. Signal Process.*, vol. 60, pp. 3439–3448, July 2012.

- D. L. Donoho, I. M. Johnstone, and A. Montanari, “Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising,” *IEEE Trans. Inform. Theory*, vol. 59, June 2013.

- C. A. Metlzer, A. Maleki, and R. G. Baraniuk, “BM3D-AMP: A new image recovery algorithm based on BM3D denoising,” in *Proc. IEEE Int. Conf. Image Process.*, pp. 3116–3120, 2015.

- R. Berthier, A. Montanari, and P.-M. Nguyen, “State evolution for approximate message passing with non-separable functions,” *Inform. Inference*, 2019.

References III

- C. Rush and R. Venkataramanan, "Finite-sample analysis of approximate message passing algorithms," *IEEE Trans. Inform. Theory*, vol. 64, no. 11, pp. 7264–7286, 2018.
- S. Rangan, P. Schniter, and A. K. Fletcher, "Vector approximate message passing," *IEEE Trans. Inform. Theory*, to appear (see also arXiv:1610.03082).
- A. M. Tulino, G. Caire, S. Verdú, and S. Shamai (Shitz), "Support recovery with sparsely sampled free random matrices," *IEEE Trans. Inform. Theory*, vol. 59, pp. 4243–4271, July 2013.
- B. Çakmak, O. Winther, and B. H. Fleury, "S-AMP: Approximate message passing for general matrix ensembles," in *Proc. Inform. Theory Workshop*, pp. 192–196, 2014.
- J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborová, "Adaptive damping and mean removal for the generalized approximate message passing algorithm," in *Proc. IEEE Int. Conf. Acoust. Speech & Signal Process.*, pp. 2021–2025, 2015.
- A. K. Fletcher, M. Sahraee-Ardakan, S. Rangan, and P. Schniter, "Expectation consistent approximate inference: Generalizations and convergence," in *Proc. IEEE Int. Symp. Inform. Thy.*, pp. 190–194, 2016.
- M. Opper and O. Winther, "Expectation consistent approximate inference," *J. Mach. Learn. Res.*, vol. 1, pp. 2177–2204, 2005.

References IV

- T. Minka, *A Family of Approximate Algorithms for Bayesian Inference.* PhD thesis, Dept. Comp. Sci. Eng., MIT, Cambridge, MA, Jan. 2001.
- S. Ramani, T. Blu, and M. Unser, "Monte-Carlo SURE: A black-box optimization of regularization parameters for general denoising algorithms," *IEEE Trans. Image Process.*, vol. 17, no. 9, pp. 1540–1554, 2008.
- A. K. Fletcher, P. Pandit, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," in *Proc. Neural Inform. Process. Syst. Conf.*, pp. 7440–7449, 2018.
- M. Borgerding, P. Schniter, and S. Rangan, "AMP-inspired deep networks for sparse linear inverse problems," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4293–4308, 2017.
- C. Guo and M. E. Davies, "Near optimal compressed sensing without priors: Parametric SURE approximate message passing," *IEEE Trans. Signal Process.*, vol. 63, pp. 2130–2141, 2015.
- U. Kamilov and H. Mansour, "Learning optimal nonlinearities for iterative thresholding algorithms," *IEEE Signal Process. Lett.*, vol. 23, pp. 747–751, May 2016.

References V

S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inform. Thy.*, pp. 2168–2172, Aug. 2011.
(full version at *arXiv:1010.5141*).

A. Javanmard and A. Montanari, "State evolution for general approximate message passing algorithms, with applications to spatial coupling," *Inform. Inference*, vol. 2, no. 2, pp. 115–144, 2013.

P. Schniter, S. Rangan, and A. K. Fletcher, "Vector approximate message passing for the generalized linear model," in *Proc. Asilomar Conf. Signals Syst. Comput.*, pp. 1525–1529, 2016.

A. K. Fletcher, S. Rangan, and P. Schniter, "Inference in deep networks in high dimensions," in *Proc. IEEE Int. Symp. Inform. Thy.*, 2018.

U. S. Kamilov, S. Rangan, A. K. Fletcher, and M. Unser, "Approximate message passing with consistent parameter estimation and applications to sparse learning," *IEEE Trans. Inform. Theory*, vol. 60, pp. 2969–2985, May 2014.

A. K. Fletcher, M. Sahraee-Ardakan, S. Rangan, and P. Schniter, "Rigorous dynamics and consistent estimation in arbitrarily conditioned linear systems," in *Proc. Neural Inform. Process. Syst. Conf.*, pp. 2542–2551, 2017.

References VI

- A. Mousavi, A. Maleki, and R. G. Baraniuk, "Consistent parameter estimation for LASSO and approximate message passing," *Ann. Statist.*, vol. 45, no. 6, pp. 2427–2454, 2017.
- A. Montanari and R. Venkataramanan, "Estimation of low-rank matrices via approximate message passing," *arXiv:1711.01682*, 2017.
- J. T. Parker, P. Schniter, and V. Cevher, "Bilinear generalized approximate message passing—Part I: Derivation," *IEEE Trans. Signal Process.*, vol. 62, pp. 5839–5853, Nov. 2014.
- J. T. Parker, P. Schniter, and V. Cevher, "Bilinear generalized approximate message passing—Part II: Applications," *IEEE Trans. Signal Process.*, vol. 62, pp. 5854–5867, Nov. 2014.
- J. T. Parker and P. Schniter, "Parametric bilinear generalized approximate message passing," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 795–808, 2016.
- C. Schülke, P. Schniter, and L. Zdeborová, "Phase diagram of matrix compressed sensing," *Physical Rev. E*, vol. 94, pp. 062136(1–16), Dec. 2016.