Vector Approximate Message Passing for the Generalized Linear Model

Phil Schniter

THE OHIO STATE UNIVERSITY

Duke UNIVERSITY

iiD

Collaborators: **Sundeep Rangan** (NYU), **Alyson Fletcher** (UCLA)

Supported in part by NSF grant CCF-1527162.

Asilomar Conference — Nov 8, 2016

Signal Recovery

- We consider problems where we want to
  - recover a "structured" signal $\boldsymbol{x} \in \mathbb{C}^N$
  - from "corrupted" measurements $\boldsymbol{y} \in \mathbb{C}^M$
  - of hidden linear-transform outputs $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x} \in \mathbb{C}^M$.
- The measurement corruption mechanism might be
  - additive: $y_i = z_i + w_i$, but possibly non-Gaussian
  - quantized: $y_i = \operatorname{sgn}(z_i + w_i)$, such as in classification & one-bit CS
  - phase-less: $y_i = |z_i + w_i|$, such as in phase retrieval
  - Poisson, such as in photon-limited imaging, etc...
- The signal $\boldsymbol{x}$ might be
  - (approximately) sparse, such as in compressive sensing
  - finite alphabet, such as in communications
  - constant modulus, etc...

Generalized Linear Model (GLM)

We take a statistical approach to signal recovery:

- corruption modeled using a likelihood fxn $p(\boldsymbol{y}|\boldsymbol{z})$ with $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x}$
- signal modeled using a prior distribution $p(\boldsymbol{x})$

The posterior tells all we can learn about $\boldsymbol{x}$, but it's not computable:

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{x})\, p(\boldsymbol{y}|\boldsymbol{A}\boldsymbol{x})}{p(\boldsymbol{y})}.$$

Instead, we usually settle for point estimates of $\boldsymbol{x}$ like the

- MAP estimate: $\widehat{\boldsymbol{x}}_{\mathsf{MAP}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y})$
- MMSE estimate: $\widehat{\boldsymbol{x}}_{\mathsf{MMSE}} = \mathrm{E}\{\boldsymbol{x}|\boldsymbol{y}\} = \int_{\mathbb{C}^N} \boldsymbol{x}\, p(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x}$

and perhaps marginal uncertainty information like $\mathrm{var}\{x_j|\boldsymbol{y}\}$.

Assumptions

In this talk, we assume a

- separable prior: $p(\boldsymbol{x}) = \prod_{j=1}^{N} p(x_j)$
- separable likelihood: $p(\boldsymbol{y}|\boldsymbol{z}) = \prod_{i=1}^{M} p(y_i|z_i)$

Then MAP estimation reduces to a familiar optimization problem:

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{\mathsf{MAP}} &= \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) \\
&= \arg\max_{\boldsymbol{x}} \; \ln p(\boldsymbol{x}|\boldsymbol{y}) \\
&= \arg\max_{\boldsymbol{x}} \underbrace{\textstyle\sum_{i=1}^{M} \ln p\big(y_i \,\big|\, [\boldsymbol{A}\boldsymbol{x}]_i\big)}_{\text{data fidelity}} + \underbrace{\textstyle\sum_{j=1}^{N} \ln p(x_j)}_{\text{regularization}}.
\end{aligned}$$

E.g., AWGN & Laplace $\quad\Rightarrow\quad \widehat{\boldsymbol{x}}_{\mathsf{MAP}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_1.$

But often the prior and/or likelihood are not log-concave!

Existing Methods

1. Convex optimization
   - MAP only ☹
   - need log-concave prior & likelihood ☹
2. Sparse Bayesian Learning (SBL) & Expectation Propagation (EP)
   - posterior must be log-concave ☹
   - additional constraints on prior & likelihood ☹
   - per-iteration matrix inverse (slow) ☹
3. MCMC
   - slow, convergence difficult to assess ☹
4. Generalized Approximate Message Passing (GAMP)
   - any prior & likelihood ☺
   - no matrix inverses (fast) ☺
   - guaranteed only under large, i.i.d. Gaussian $\boldsymbol{A}$ ☹

Proposed Method

We propose to . . .

1. Rewrite $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x}$ as $\mathbf{0} = \begin{bmatrix}\boldsymbol{A} & -\boldsymbol{I}\end{bmatrix} \begin{bmatrix}\boldsymbol{x} \\ \boldsymbol{z}\end{bmatrix} \triangleq \overline{\boldsymbol{A}}\overline{\boldsymbol{x}}$, thereby converting the GLM problem to a standard linear regression problem:

   Recover $\overline{\boldsymbol{x}}$ from $\overline{\boldsymbol{y}} = \overline{\boldsymbol{A}}\overline{\boldsymbol{x}} + \overline{\boldsymbol{w}}$ with $\overline{\boldsymbol{w}} \sim \mathcal{N}(\mathbf{0}, \epsilon\boldsymbol{I})$,
   where now $\overline{\boldsymbol{y}} = \mathbf{0}$ and $\epsilon \to 0$.

2. Apply the recently proposed "Vector AMP" algorithm,[1] tracking separate divergences on $\boldsymbol{x}$ and $\boldsymbol{z}$.

[1] Rangan,Schniter,Fletcher—arXiv:1610.03082

Vector AMP for Standard Linear Regression

To recover $\boldsymbol{x}$ from $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$ with $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \sigma_w^2 \boldsymbol{I})$ and i.i.d. $x_j \sim p(x_j) \ldots$

Initialize $\widetilde{\boldsymbol{r}}_1 = \boldsymbol{0}$ and $\widetilde{\sigma}_1 = \infty$.

Repeat for $t = 1, 2, 3, \ldots$

$\widetilde{\boldsymbol{x}}_t = \widetilde{\boldsymbol{\eta}}(\widetilde{\boldsymbol{r}}_t; \widetilde{\sigma}_t)$	LMMSE estimation	(1a)
$\widetilde{\tau}_t = \langle \widetilde{\boldsymbol{\eta}}'(\widetilde{\boldsymbol{r}}_t; \widetilde{\sigma}_t) \rangle$	divergence	(1b)
$\boldsymbol{r}_t = (\widetilde{\boldsymbol{x}}_t - \widetilde{\tau}_t \widetilde{\boldsymbol{r}}_t)/(1 - \widetilde{\tau}_t)$	Onsager correction	(1c)
$\sigma_t^2 = \widetilde{\sigma}_t^2 \widetilde{\tau}_t/(1 - \widetilde{\tau}_t)$	variance update	(1d)
$\widehat{\boldsymbol{x}}_t = \boldsymbol{\eta}(\boldsymbol{r}_t; \sigma_t)$	denoising	(2a)
$\tau_t = \langle \boldsymbol{\eta}'(\boldsymbol{r}_t; \sigma_t) \rangle$	divergence	(2b)
$\widetilde{\boldsymbol{r}}_{t+1} = (\widehat{\boldsymbol{x}}_t - \tau_t \boldsymbol{r}_t)/(1 - \tau_t)$	Onsager correction	(2c)
$\widetilde{\sigma}_{t+1}^2 = \sigma_t^2 \tau_t/(1 - \tau_t)$	variance update	(2d)

where

$$[\boldsymbol{\eta}(\boldsymbol{r}_t; \sigma_t)]_j = \begin{cases} \int x_j p(x_j | r_{tj}) \mathrm{d}x_j & \text{MMSE} \\ \arg\max_{x_j} p(x_j | r_{tj}) & \text{MAP} \end{cases} \quad \text{with} \quad p(x_j | r_{tj}) \propto p(x_j) \mathcal{N}(x_j; r_{tj}, \sigma_t^2)$$

$$\widetilde{\boldsymbol{\eta}}(\widetilde{\boldsymbol{r}}_t; \widetilde{\sigma}_t) = \boldsymbol{V} \left(\mathrm{Diag}(\boldsymbol{s})^2 + \frac{\sigma_w^2}{\widetilde{\sigma}_t^2} \boldsymbol{I}_R \right)^{-1} \left(\mathrm{Diag}(\boldsymbol{s}) \boldsymbol{U}^\mathsf{T} \boldsymbol{y} + \frac{\sigma_w^2}{\widetilde{\sigma}_t^2} \boldsymbol{V}^\mathsf{T} \widetilde{\boldsymbol{r}}_t \right)$$

with SVD $\boldsymbol{A} = \boldsymbol{U}\, \mathrm{Diag}(\boldsymbol{s}) \boldsymbol{V}^\mathsf{H}$

Why call this "Vector AMP"?

1) Can be derived using an approximation of message passing on a factor graph, now with vector-valued variable nodes.

2) Performance rigorously characterized by a scalar state-evolution[2] under certain large random $\boldsymbol{A}$:

SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{\mathsf{T}}$

- $\boldsymbol{U}$ *is deterministic*
- $\boldsymbol{S}$ *is deterministic*
- $\boldsymbol{V}$ *is uniformly distributed on the group of orthogonal matrices*

"$\boldsymbol{A}$ *is right rotationally invariant.*"

Thus the VAMP state evolution holds for "almost any $\boldsymbol{A}$."

[2] Rangan,Fletcher,Schniter–16

Connections to the Replica Prediction

- The replica method from statistical physics is often used to characterize the average behavior of large disordered systems.
- Although not fully rigorous, replica predictions are usually correct.
- For estimation of i.i.d. $\boldsymbol{x}$ from measurements $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \mathcal{N}(\mathbf{0}; \sigma_w^2 \boldsymbol{I})$ under large right-rotationally invariant $\boldsymbol{A}$:

  The MMSE $\mathcal{E}(\sigma_t^2)$ should satisfy the fixed-point equation[3]

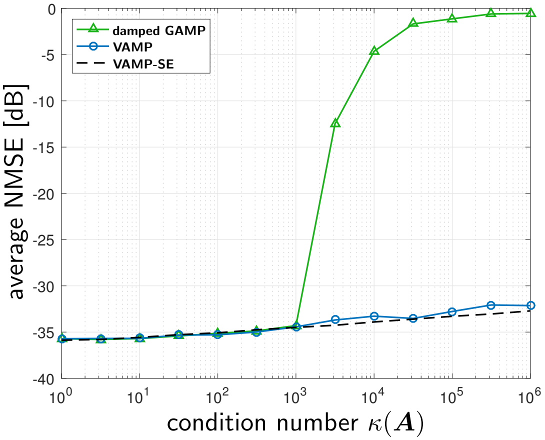
  $$1/\sigma_t^2 = R_{\boldsymbol{A}^\mathsf{T}\boldsymbol{A}/\sigma_w^2}(-\mathcal{E}(\sigma_t^2)),$$

  where $R_{\boldsymbol{C}}(\cdot)$ denotes the R-transform of matrix $\boldsymbol{C}$ and $\mathcal{E}(\sigma_t^2) \triangleq \mathrm{E}\left\{\left[\eta\big(x_j + \mathcal{N}(0, \sigma_t^2); \sigma_t^2\big) - x_j\right]^2\right\}$.

- It can be shown that VAMP's SE fixed-points obey the above equation.
- Thus, assuming that the replica prediction is correct, VAMP will generate MMSE estimates whenever these fixed-points are unique.

[3] Tulino, Caire, Verdu, Shamai—TIT'13

Numerical Results: 1-Bit Compressive Sensing



$N = 512$
$M/N = 4$

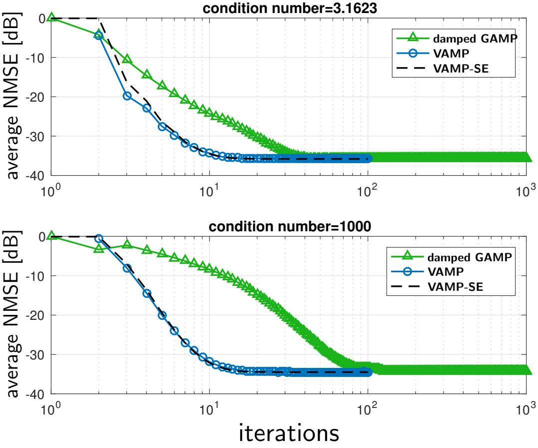
$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})V^{\mathsf{T}}$
$\boldsymbol{U}, \boldsymbol{V}$ drawn uniform
$s_i/s_{i-1} = \rho\ \forall i$
ρ determines $\kappa(\boldsymbol{A})$

x_j ∼Bernoulli-Gaussian
$\Pr\{x_j \neq 0\} = 1/32$

SNR $= 40$dB

VAMP is robust to ill-conditioned $\boldsymbol{A}$; GAMP is not.

Numerical Results: 1-Bit Compressive Sensing



$N = 512$
$M/N = 4$

$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})V^{\mathsf{T}}$
$\boldsymbol{U}, \boldsymbol{V}$ drawn uniform
$s_i/s_{i-1} = \rho\ \forall i$
ρ determines $\kappa(\boldsymbol{A})$

$x_j \sim$Bernoulli-Gaussian
$\Pr\{x_j \neq 0\} = 1/32$

SNR$= 40$dB

VAMP is much faster than damped GAMP.

Non-parametric Estimation

- So far we have considered estimating $\boldsymbol{x}$ from

$$\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{z};\boldsymbol{\theta}_z) \;\; \text{where} \;\; \boldsymbol{z} = \boldsymbol{A}\boldsymbol{x} \;\; \text{and} \;\; \boldsymbol{x} \sim p(\boldsymbol{x};\boldsymbol{\theta}_x),$$

  where $\boldsymbol{\theta}_z$ and $\boldsymbol{\theta}_x$ are parameters of the likelihood and prior.

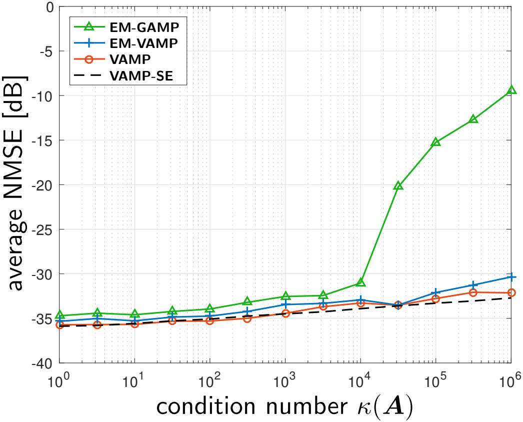
- What if $\boldsymbol{\theta}_z$ and $\boldsymbol{\theta}_z$ are unknown? Can we learn them from $\boldsymbol{y}$?

- Yes! The "EM-VAMP" approach[4] can be directly applied.

[4] Fletcher,Schniter—arXiv:1602.08207

Numerical Results: Nonparametric 1-Bit CS

Learning both σ_w^2 and BG parameters:



$N = 512$
$M/N = 4$

$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})V^{\mathsf{T}}$
$\boldsymbol{U}, \boldsymbol{V}$ drawn uniform
$s_i/s_{i-1} = \rho\ \forall i$
ρ determines $\kappa(\boldsymbol{A})$

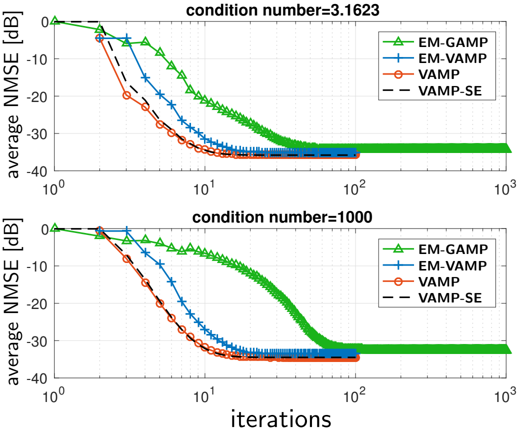
x_j ~Bernoulli-Gaussian
$\Pr\{x_j \neq 0\} = 1/32$

SNR $= 40$dB

EM-VAMP performs near oracle VAMP even with ill-conditioned $\boldsymbol{A}$.

Numerical Results: Nonparametric 1-Bit CS

Learning both σ_w^2 and BG parameters:



$N = 512$
$M/N = 4$

$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\boldsymbol{s})V^{\mathsf{T}}$
$\boldsymbol{U}, \boldsymbol{V}$ drawn uniform
$s_i/s_{i-1} = \rho\ \forall i$
ρ determines $\kappa(\boldsymbol{A})$

x_j ∼Bernoulli-Gaussian
$\Pr\{x_j \neq 0\} = 1/32$

SNR$= 40$dB

EM-VAMP slightly slower than VAMP but much faster than EM-GAMP.

Conclusions

- We proposed a new approach for inference under generalized linear models (GLMs).
- Applications include 1-bit compressive sensing, binary classification, (compressive) phase retrieval, photon-limited imaging, etc.
- Our approach builds on the recently proposed "vector AMP" algorithm, which (unlike AMP) is robust to the choice of measurement operator $\boldsymbol{A}$.
- After an initial SVD, our approach consumes only two matrix-vector multiplications per iteration and converges in ~ 10 iterations.
- Our approach can be easily extended to the nonparametric case, where the likelihood and/or prior have unknown parameters, via EM-VAMP.
- In the future, we hope to rigorously prove the state evolution of VAMP-GLM and analyze the performance of EM-VAMP-GLM.