A Concise (though Heuristic) Derivation of AMP

Weijia Zheng [1]

Department of Information Engineering
The Chinese University of Hong Kong

wjzheng@link.cuhk.edu.hk

May 15, 2025

[1] I mainly took reference to: "A Simple Derivation of AMP and its State Evolution via First-Order Cancellation" by P. Schniter. This is a very readable file on this topic.

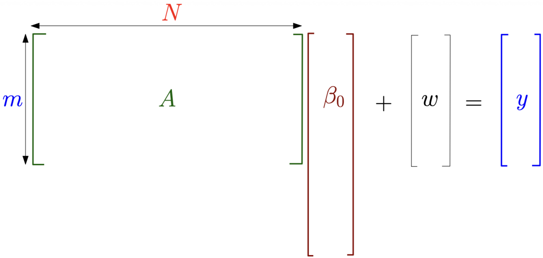
Overview

Overview

High dimensional linear regression problem

Liner regression formulation

Consider a problem of the form $\mathbf{y} = \mathbf{A}\beta_0 + \mathbf{w}$. We want to reconstruct β_0 from $\mathbf{y}$.



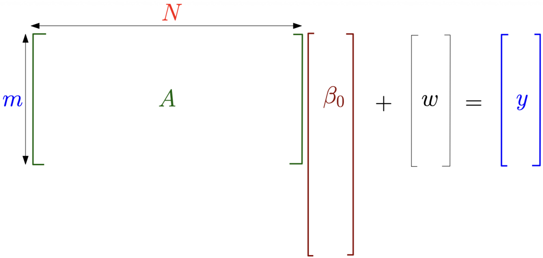
[2]

[2] Figure copied from "Approximate Message Passing for Statistical Inference and Estimation" (good lecture slides with Youtube video recording)

High dimensional linear regression problem

Liner regression formulation

Consider a problem of the form $\mathbf{y} = \mathbf{A}\boldsymbol{\beta}_0 + \mathbf{w}$. We want to reconstruct $\boldsymbol{\beta}_0$ from $\mathbf{y}$.



[2]

- $\mathbf{y}$: an observed length-m measurement vector
- $\mathbf{w}$: an unknown length-m noise. Assume $w \sim_{iid} \mathcal{N}(0, \tau_w)$
- $\mathbf{A}$: a known big $m \times N$ (normalized) matrix with $m < N$, $\frac{m}{N} \to \delta \in \Omega(1)$
- $\boldsymbol{\beta}_0$: a length-N signal vector to find

[2]Figure copied from "Approximate Message Passing for Statistical Inference and Estimation" (good lecture slides with Youtube video recording)

$$\begin{array}{cc} & N \\ m & \begin{bmatrix} \quad & A & \quad \end{bmatrix} \end{array} \begin{bmatrix} \beta_0 \\ \\ \\ \end{bmatrix} + \begin{bmatrix} w \end{bmatrix} = \begin{bmatrix} y \end{bmatrix}$$

Prior knowledge on β_0: sparsity

People may assume sparsity of β_0. That is, it has only $K \ll N$ nonzero entries.

$$m \left[\quad \overset{\xleftarrow{\quad N \quad} \xrightarrow{\quad\quad}}{A} \quad \right] \left[\beta_0 \right] + \left[w \right] = \left[y \right]$$

Prior knowledge on β_0: sparsity

People may assume sparsity of β_0. That is, it has only $K \ll N$ nonzero entries.

NP-hardness of sparse recovery in general

Assume K-sparse, the problem: for any given $\mathbf{A}$, find $\arg\min_{\beta} \|\mathbf{A}\beta - \mathbf{y}\|^2$ is NP-hard. In fact, even if we know the entries' values of the ground-truth β_0, the problem is still NP-hard. [a]

[a] For the NP-hardness: one can do reduction using Exact Cover by 3-Sets (X3C).

(Sparsity inspired) LASSO

LASSO and ISTA

$$\min_{\boldsymbol{\beta}} \underbrace{\frac{1}{2}\|\mathbf{y}-\mathbf{A}\boldsymbol{\beta}\|^2}_{\triangleq g(\boldsymbol{\beta})} + \lambda\|\boldsymbol{\beta}\|_1.$$

Iterative Soft-Thresholding Algorithm (ISTA) can solve this:

$$\mathbf{v} = \mathbf{y} - \mathbf{A}\boldsymbol{\beta}^t$$
$$\boldsymbol{\beta}^{t+1} = \texttt{soft}(\boldsymbol{\beta}^t + s\mathbf{A}^T\boldsymbol{v}^t; s\lambda).$$

Writing it into a more intuitive form:

$$\boldsymbol{\beta}^{t+1} = \underbrace{\texttt{soft}(\underbrace{\boldsymbol{\beta}^t - s\nabla g(\boldsymbol{\beta^t})}_{\text{grad. descent}}; s\lambda)}_{\text{impose sparsity}}.$$

$$\nabla g(\boldsymbol{\beta}) = \mathbf{A}^T(\mathbf{A}\boldsymbol{\beta} - \mathbf{y}).$$

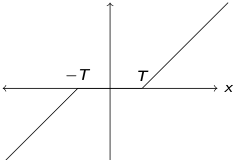


Figure: $\texttt{soft}(x, T)$.

One can tune the parameter λ to control the sparsity, and s here works as a stepsize.

LASSO is convex in $\boldsymbol{\beta}$.

AMP Framework

LASSO & ISTA are great, but...

LASSO is motivated by **sparsity** alone, and it does not consider the signal's prior distribution, which may sometimes be available. Thus, people want to integrate the knowledge of $\boldsymbol{\beta}_0 \sim_{iid} p_\beta$ into inference of $\boldsymbol{\beta}_0$.[a]

[a] AMP does not explicitly assume sparsity of $\boldsymbol{\beta}_0$.

The problem then changes to find an $\hat{\boldsymbol{\beta}}$ for $\mathbf{y} = \mathbf{A}\boldsymbol{\beta}_0 + \mathbf{w}, \quad$ while $\boldsymbol{\beta}_0 \sim_{iid} p_\beta$.

AMP Framework

LASSO & ISTA are great, but...

LASSO is motivated by **sparsity** alone, and it does not consider the signal's prior distribution, which may sometimes be available. Thus, people want to integrate the knowledge of $\boldsymbol{\beta}_0 \sim_{iid} p_\beta$ into inference of $\boldsymbol{\beta}_0$.[a]

[a] AMP does not explicitly assume sparsity of $\boldsymbol{\beta}_0$.

The problem then changes to find an $\hat{\boldsymbol{\beta}}$ for $\mathbf{y} = \mathbf{A}\boldsymbol{\beta}_0 + \mathbf{w}$, while $\boldsymbol{\beta}_0 \sim_{iid} p_\beta$.

We compare the procedure of AMP and ISTA at below.

Approximate message passing (AMP)

$$\mathbf{v}^t = \mathbf{y} - \mathbf{A}\boldsymbol{\beta}^t + \underbrace{\frac{\mathbf{v}^{t-1}}{m} \sum_{j=1}^{N} \eta'_{t-1}(r_j^{t-1})}_{\text{Onsager correction term}}$$

$$\boldsymbol{\beta}^{t+1} = \eta_t(\underbrace{\boldsymbol{\beta}^t + s\mathbf{A}^T\mathbf{v}^t}_{\triangleq \boldsymbol{r}^t}).$$

Iterative Soft Thresholding Algo. (ISTA)

$$\mathbf{v}^t = \mathbf{y} - \mathbf{A}\boldsymbol{\beta}^t$$

$$\boldsymbol{\beta}^{t+1} = \mathtt{soft}(\boldsymbol{\beta}^t + s\mathbf{A}^T\mathbf{v}^t; s\lambda).$$

There are some requirements in $\boldsymbol{A}$, the sensing/measurement matrix. In general, assume $\boldsymbol{A}$ to be entry-wisely iid generated with $\mathbb{E}a_{ij} = 0$ and $\mathbb{E}(a_{ij}^2) = \frac{1}{m}$ suffices.

In fact, in the paper we will go through, they assume $a_{ij} \in \mathcal{U}\{\pm\frac{1}{\sqrt{m}}\}$. But this is mainly to simplify the proof, and it can be extended to more general cases.

There are some requirements in $\boldsymbol{A}$, the sensing/measurement matrix. In general, assume $\boldsymbol{A}$ to be entry-wisely iid generated with $\mathbb{E}a_{ij} = 0$ and $\mathbb{E}(a_{ij}^2) = \frac{1}{m}$ suffices.

In fact, in the paper we will go through, they assume $a_{ij} \in \mathcal{U}\{\pm\frac{1}{\sqrt{m}}\}$. But this is mainly to simplify the proof, and it can be extended to more general cases.

AMP iteration

$$\mathbf{v}^t = \mathbf{y} - \mathbf{A}\beta^t + \underbrace{\frac{\boldsymbol{v}^{t-1}}{m}\sum_{j=1}^{N}\eta'_{t-1}(r_j^{t-1})}_{\text{Onsager correction term}}$$

$$\boldsymbol{\beta}^{t+1} = \eta_t(\underbrace{\boldsymbol{\beta}^t + s\mathbf{A}^T\mathbf{v}^t}_{\triangleq \boldsymbol{r}^t}).$$

$\boldsymbol{r}^t \in \mathbb{R}^N$, termed **"effective observation"**

$\eta_t(\cdot)$ is called a "denoising function"

$[\eta_t(\boldsymbol{r})]_j = \eta_t(r_j)$

In ISTA, $\eta_t = \texttt{soft}()$ and we do not consider any correction term

Main purpose of the paper

Simply to understand: why there is such an "Onsager term", and what should be chosen as the denoising function $\eta_t(\cdot)$.

Overview

Why the Onsager Correction?

AMP Iteration Recap

For $\mathbf{y} = \mathbf{A}\beta_0 + \mathbf{w}$, AMP iterates:

$$\mathbf{v}^t = \mathbf{y} - \mathbf{A}\beta^t + \underbrace{\frac{\mathbf{v}^{t-1}}{m} \sum_{j=1}^{N} \eta'_{t-1}(r_j^{t-1})}_{\text{Onsager correction term}},$$

$$\boldsymbol{\beta}^{t+1} = \eta_t \left(\boldsymbol{\beta}^t + \mathbf{A}^T \mathbf{v}^t\right) \triangleq \eta_t(\mathbf{r}^t).$$

Why the Onsager Correction?

AMP Iteration Recap

For $\mathbf{y} = \mathbf{A}\beta_0 + \mathbf{w}$, AMP iterates:

$$\mathbf{v}^t = \mathbf{y} - \mathbf{A}\beta^t + \underbrace{\frac{\mathbf{v}^{t-1}}{m} \sum_{j=1}^{N} \eta'_{t-1}(r_j^{t-1})}_{\text{Onsager correction term}},$$

$$\beta^{t+1} = \eta_t\left(\beta^t + \mathbf{A}^T\mathbf{v}^t\right) \triangleq \eta_t(\mathbf{r}^t).$$

Partial goal: Ensure $\mathbf{r}^t - \beta_0 \approx$ Gaussian noise. (See next page.)

Onsager term adjusts $\mathbf{v}^t$ to cancel error correlations.

Our focus soon

Derive the Onsager term by analyzing the difference between the effective observation and ground-truth signal $\mathbf{e}^t = \mathbf{r}^t - \beta_0$.

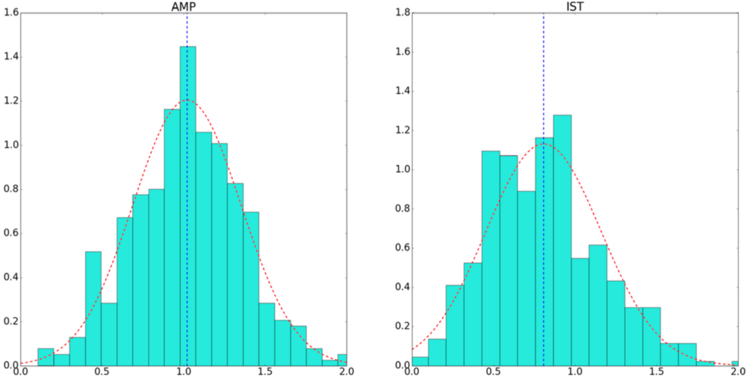
Correction terms work

Correction terms push the effective observation to β_0+ some (tiny) Gaussian.

Comparing with vanilla IST, such effect is decisive.

$$A : m \times N = 2000 \times 4000;\ \beta_0 \text{ has 500 non-zeros} \sim \text{iid unif } \pm 1$$

Histogram of $A^T r^t + \beta^t$ at indices where $\beta_0 = +1$ at $t = 10$



Error Analysis

Define the Error

Recall $\mathbf{r}^t = \beta^t + \mathbf{A}^T\mathbf{v}^t$. The error is: $\mathbf{e}^t = \mathbf{r}^t - \beta_0 = \left(\beta^t + \mathbf{A}^T\mathbf{v}^t\right) - \beta_0$. Substitute $\mathbf{v}^t = \mathbf{y} - \mathbf{A}\beta^t + \mathbf{u}^t$, where $\mathbf{u}^t$ denotes (any) correction term:

$$\mathbf{r}^t = \beta^t + \mathbf{A}^T\left(\mathbf{y} - \mathbf{A}\beta^t + \mathbf{u}^t\right).$$

Since $\mathbf{y} = \mathbf{A}\beta_0 + \mathbf{w}$, $\mathbf{A}^T\mathbf{y} = \mathbf{A}^T(\mathbf{A}\beta_0 + \mathbf{w})$. Then

$$\boldsymbol{e}^t = (\boldsymbol{I} - \boldsymbol{A}^T\boldsymbol{A})(\beta^t - \beta_0) + \boldsymbol{A}^T(\boldsymbol{w} + \boldsymbol{u}^t).$$

Error Analysis

Define the Error

Recall $\mathbf{r}^t = \boldsymbol{\beta}^t + \mathbf{A}^T\mathbf{v}^t$. The error is: $\mathbf{e}^t = \mathbf{r}^t - \boldsymbol{\beta}_0 = \left(\boldsymbol{\beta}^t + \mathbf{A}^T\mathbf{v}^t\right) - \boldsymbol{\beta}_0$.
Substitute $\mathbf{v}^t = \mathbf{y} - \mathbf{A}\boldsymbol{\beta}^t + \mathbf{u}^t$, where $\mathbf{u}^t$ denotes (any) correction term:

$$\mathbf{r}^t = \boldsymbol{\beta}^t + \mathbf{A}^T\left(\mathbf{y} - \mathbf{A}\boldsymbol{\beta}^t + \mathbf{u}^t\right).$$

Since $\mathbf{y} = \mathbf{A}\boldsymbol{\beta}_0 + \mathbf{w}$, $\mathbf{A}^T\mathbf{y} = \mathbf{A}^T(\mathbf{A}\boldsymbol{\beta}_0 + \mathbf{w})$. Then

$$\boldsymbol{e}^t = (\boldsymbol{I} - \boldsymbol{A}^T\boldsymbol{A})(\boldsymbol{\beta}^t - \boldsymbol{\beta}_0) + \boldsymbol{A}^T(\boldsymbol{w} + \boldsymbol{u}^t).$$

Zooming in the l-th entry

We focus more on $\boldsymbol{\beta}^t$. $\boldsymbol{\beta}^t = \eta_{t-1}(\boldsymbol{r}^{t-1})$. Its l-th entry is $\beta_l^t = \eta_{t-1}(r_l^{t-1})$.
And we decompose r_l^{t-1} as:

$$r_l^{t-1} = \beta_l^{t-1} + \sum_k a_{kl} v_k^{t-1} = \underbrace{\beta_l^{t-1} + \sum_{k \neq i} a_{kl} v_k^{t-1}}_{\triangleq r_{l \setminus i}^{t-1},\ \text{assumed indep. of } \{a_{ij}\}_j} + a_{il} v_i^{t-1}. \tag{1}$$

Error decomposition

Then the error (its j-th entry) becomes

$$e_j^t = \sum_i a_{ij} \left[\sum_{l \neq j} a_{il} (\beta_{0,l} - \beta_l^t) + w_i + u_i^t \right]$$

$$= \underbrace{\sum_i a_{ij} \sum_{l \neq j} a_{il} [\beta_{0,l} - \eta_{t-1}(r_{l \setminus i}^{t-1})] + \sum_i a_{ij} w_i}_{\text{Indepndence + CLT} \implies \sim_d \text{ Gaussian}}$$

$$+ \underbrace{\sum_i a_{ij} \left[u_i^t + \sum_{l \neq j} -\frac{v_i^{t-1}}{m} \eta'_{t-1}(r_{l \setminus i}^{t-1}) \right]}_{\triangleq T_1 \text{ ,want to make it small when } m \text{ is large}}$$

a_{ij} and v_i^{t-1} are coupled!

In the above, we used Taylor expansion:

$$\beta_l^t = \eta_{t-1}(r_l^{t-1}) = \eta_{t-1}(r_{l \setminus i}^{t-1} + a_{il} v_i^{t-1}) \approx \eta_{t-1}(r_{l \setminus i}^{t-1}) + a_{il} v_i^{t-1} \eta'_{t-1}(r_{l \setminus i}^{t-1}).$$

And, we used $a_{il}^2 = \frac{1}{m}$.

Focus on Correction Term

The last error term (the not Gaussian one) is the only term involving $\mathbf{u}^t$:

$$T_1 = \sum_i a_{ij} \left[u_i^t - \sum_{l \neq j} \frac{v_i^{t-1}}{m} \eta'_{t-1}(r_{l \setminus i}^{t-1}) \right] \tag{2}$$

Note that $\mathbf{u}^t$ is some correction term free to choose. And we wish such choice to make T_1 small.

We can see why Onsager is good by observing its form: $u_i^t \triangleq \frac{v_i^{t-1}}{m} \sum_{l=1}^N \eta'_{t-1}(r_l^{t-1})$.
We can proceed an estimation of T_1:

$$T_1 \approx_{\text{2 order Taylor}} \approx \sum_i a_{ij} \left[\frac{v_i^{t-1}}{m} \sum_{l=1}^{N} \eta'_{t-1}(r_l^{t-1}) - \sum_{l \neq j} \frac{v_i^{t-1}}{m} \eta'_{t-1}(r_{l \setminus i}^{t-1}) \right] \tag{3}$$

$$\approx \frac{1}{m} \sum_i a_{ij} v_i^{t-1} \left[\eta'_{t-1}(r_j^{t-1}) + \sum_{l \neq j} a_{il} v_i^{t-1} \eta''_{t-1}(r_{l \setminus i}^{t-1}) \right] \in \mathcal{O}(\frac{1}{\sqrt{m}}) \tag{4}$$

The two terms in eq. (4) are both $\mathcal{O}(\frac{1}{\sqrt{m}})$. And it is hard to design a better correction term (without sacrificing too much computational complexity.)

Overview

Completing Error Estimation

Recap: Error Decomposition

From the previous slide, the error e_j^t is:

$$e_j^t = \underbrace{\underbrace{\sum_i a_{ij} \sum_{l \neq j} a_{il} [\underbrace{\beta_{0,l} - \eta_{t-1}(r_{l \setminus i}^{t-1})}_{\triangleq \epsilon_{l \setminus i}^t}]}_{= S_1} + \underbrace{\sum_i a_{ij} w_i}_{= S_2}}_{\text{Indepndence + CLT} \implies \sim_d \text{ Gaussian}} + \underbrace{\cancel{T_1}}_{\text{ignored when } m \gg 1}$$

S_2 is much easier to handle, it has zero mean and variance $= \tau_w$. ($\boldsymbol{w}$'s power)

S_1 has mean zero. And it has variance $\frac{1}{m^2} \sum_{i=1}^m \sum_{l \neq j} (\epsilon_{l \setminus i}^t)^2 \approx \frac{n}{m} \frac{1}{n} \sum_{l=1}^n (\epsilon_l^t)^2$, where $\epsilon_l^t \triangleq \beta_{0,l} - \eta_{t-1}(r_l^{t-1})$ is the effective noise.

People denote $\mathcal{E}^t = \frac{1}{n} \sum_{l=1}^n (\epsilon_l^t)^2$. Then one can write

$$e_j^t = \beta_{0,j} - r_j^t \sim_{\text{approx.}} \mathcal{N}(0, \underbrace{\delta^{-1} \mathcal{E}^t + \underbrace{\tau_w}_{\text{fixed}}}_{\triangleq \tau_r^t}).$$

State Evolution: Predicting Performance

State Evolution Equations

Track error variance $\tau_r^t = \mathrm{Var}(r_j^t - \beta_{0,j})$:

$$\tau_r^t = \delta^{-1}\mathcal{E}^t + \tau_w, \quad \mathcal{E}^{t+1} = \mathbb{E}\left[\eta_t(\beta_0 + Z_t) - \beta_0\right]^2, \quad Z_t \sim \mathcal{N}(0, \underbrace{\tau_r^t}_{\approx \frac{\|v^t\|^2}{m}}).$$

$\mathcal{E}^t$: Mean squared error (MSE) of denoiser at iteration t.

It predicts AMP's MSE without running the algorithm.

Basically, $\mathcal{E}^t$ is what we can control in τ_r^t. Hence we want to choose a function $\eta_t()$ to minimize it!

Now, we see why people choose $\eta_t \leftarrow$ posterior mean estimator (PME).[3]

[3] One can use Tweedie's formula here, when β_0's prior is known.