Asymptotics of generalized linear models beyond Gaussian matrices

Cedric Gerbelot

Joint work with A. Abbara and F. Krzakala

Ecole Normale Superieure, PSL University, Paris, France

August 11, 2020

Supervised learning : learning from labeled examples

- Input space $\mathcal{A}$ (ex. $\mathbb{R}^N$)
- Output space $\mathcal{Y}$ (ex. $\{-1, 1\}$ for classification, $\mathbb{R}$ for regression)
- Training set $\mathcal{S}_M = (\mathbf{a}_i, y_i)_{i=1,\ldots,M}$ of (input,output) pairs.

Goal is to estimate a function h : $\mathcal{A} \to \mathcal{Y}$ to predict new outputs

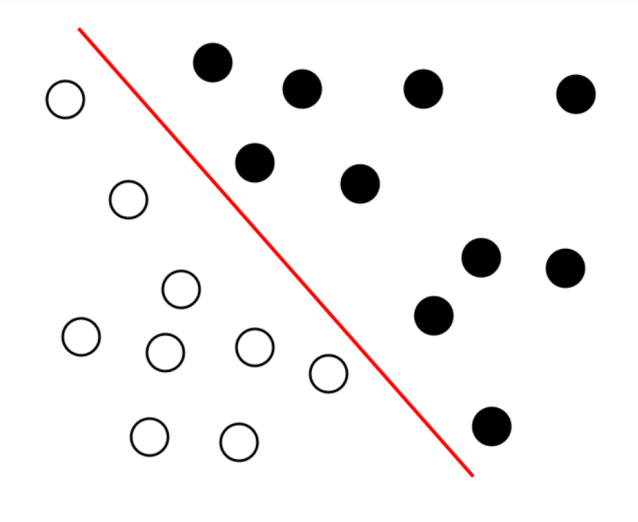

Figure 1: Classification

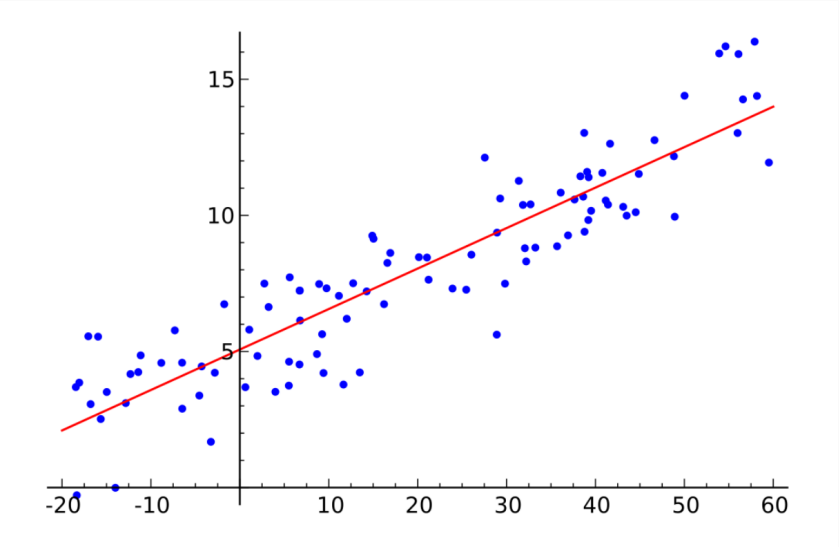

Figure 2: Regression

Choosing a parametrization : simple taste

Convex generalized linear model

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\min} \left\{ \sum_{i=1}^{M} g(\mathbf{a}_i^T \mathbf{x}, y_i) + f(\mathbf{x}) \right\} \tag{1}$$

where $\mathbf{y} = \phi(\mathbf{A}\mathbf{x}_0)$

$\mathbf{x}_0 \sim p_{x_0}$

- ground-truth $\mathbf{x}_0$ pulled from any (well-behaved) distribution
- f, g are convex functions
- **high dimensional limit** $M, N \to \infty$, fixed ratio $\alpha = M/N$

Examples

Linear regression (ridge, LASSO, elastic-net...)

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + f(\mathbf{x}) \right\}$$

Logistic regression

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \left\{ \sum_{i=1}^{N} \log(1 + \exp(-y_i \mathbf{a}_i^T \mathbf{x})) + f(\mathbf{x}) \right\}$$

Linear Support Vector Classifier

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \left\{ \sum_{i=1}^{N} \max(0, 1 - y_i \mathbf{a}_i^T \mathbf{x}) + f(\mathbf{x}) \right\}$$

Objective : how good is my regression ?

Asymptotic reconstruction performance

$$\rho_0 = \lim_{N\to\infty} \frac{1}{N}\|\mathbf{x}_0\|_2^2$$

$$q^* = \lim_{N\to\infty} \frac{1}{N}\|\mathbf{x}^*\|_2^2$$

$$m^* = \lim_{N\to\infty} \frac{1}{N}\mathbf{x}_0^T\mathbf{x}^*$$

Objective : how good is my regression ?

Asymptotic reconstruction performance

$$\rho_0 = \lim_{N\to\infty} \frac{1}{N}\|\mathbf{x}_0\|_2^2$$

$$q^* = \lim_{N\to\infty} \frac{1}{N}\|\mathbf{x}^*\|_2^2$$

$$m^* = \lim_{N\to\infty} \frac{1}{N}\mathbf{x}_0^T\mathbf{x}^*$$

$$\text{(MSE)} \quad E = \lim_{N\to\infty} \frac{1}{N}\|\mathbf{x}^* - \mathbf{x}_0\|_2^2 = \rho_0 - 2m^* + q^*$$

$$\text{(Angle)} \quad \theta = \lim_{N\to\infty} (\mathbf{x}_0, \mathbf{x}^*) = \arccos\left(\frac{m^*}{\sqrt{\rho_0 q^*}}\right)$$

Statistical physics of supervised learning

Hamiltonian :

$$\mathcal{H}(\mathbf{A}, \mathbf{y}) = \sum_{i=1}^{M} g(\mathbf{a}_i^T \mathbf{x}, y_i) + f(\mathbf{x})$$

Partition function :

$$\mathcal{Z} = \int_{\mathcal{A},\mathcal{Y}} \exp(-\beta \mathcal{H}(\mathbf{A}, \mathbf{y})) d\mu(\mathbf{A}) d\mu(\mathbf{y})$$

Statistical physics of supervised learning

Hamiltonian :

$$\mathcal{H}(\mathbf{A}, \mathbf{y}) = \sum_{i=1}^{M} g(\mathbf{a}_i^T \mathbf{x}, y_i) + f(\mathbf{x})$$

Partition function :

$$\mathcal{Z} = \int_{\mathcal{A},\mathcal{Y}} \exp(-\beta \mathcal{H}(\mathbf{A}, \mathbf{y})) d\mu(\mathbf{A}) d\mu(\mathbf{y})$$

Replica computation works very well

Figure 3: Convexity ? Too easy...

Proving replicas : some history and Bayes optimality

Gaussian interpolation $\rightarrow$ **Krzakala lecture**

(i) in math. stat. phys. [Guerra, Toninelli02], [Talagrand03]

(ii) Bayes optimal, i.i.d. Gaussian case [BDMK16]

(iii) Bayes optimal, correlated Gaussian/complex case [MLKZ20]

Proving replicas : some history and Bayes optimality

Gaussian interpolation $\rightarrow$ **Krzakala lecture**

(i) in math. stat. phys. [Guerra, Toninelli02], [Talagrand03]

(ii) Bayes optimal, i.i.d. Gaussian case [BDMK16]

(iii) Bayes optimal, correlated Gaussian/complex case [MLKZ20]

Bayes optimal is perfect for sig. proc.

$\Downarrow$

Machine learning is not Bayes optimal

Out of Bayes optimality : asymptotic errors

LASSO risk for Gaussian matrices [BM11]. $\mathbf{A}$ i.i.d. $\mathcal{N}(0,1)$.

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\min} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda_1 |x| \right\}$$

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{w}_0$$

Out of Bayes optimality : asymptotic errors

LASSO risk for Gaussian matrices [BM11]. $\mathbf{A}$ i.i.d. $\mathcal{N}(0,1)$.

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\min} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda_1 |x| \right\}$$

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{w}_0$$

Use **approximate message-passing**

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}^t + \frac{1}{\alpha}\mathbf{z}^{t-1}\langle \eta'(\hat{\mathbf{x}}^{t-1} + \frac{1}{\alpha}\mathbf{A}^T\mathbf{z}^{t-1}, \theta^{t-1})\rangle$$

$$\hat{\mathbf{x}}^{t+1} = \eta(\hat{\mathbf{x}}^t + \frac{1}{\alpha}\mathbf{A}^T\mathbf{z}^t, \theta^t)$$

η is the soft-thresholding operator (proximal of ℓ_1)

AMP state evolution equations

AMP for LASSO risk

$$\mathbf{z}^t = y - \mathbf{A}\hat{\mathbf{x}}^t + \frac{1}{\alpha}\mathbf{z}^{t-1}\langle \eta'(\hat{\mathbf{x}}^{t-1} + \frac{1}{\alpha}\mathbf{A}^T\mathbf{z}^{t-1}, \theta^{t-1})\rangle$$

$$\hat{\mathbf{x}}^{t+1} = \eta(\hat{\mathbf{x}}^t + \frac{1}{\alpha}\mathbf{A}^T\mathbf{z}^t, \theta^t)$$

State evolution, $Z \sim \mathcal{N}(0,1)$ [BM11+]

$$V = \mathbb{E}_{z,x_0}\{[\eta'(X_0 + \sqrt{\frac{\Delta_0 + E}{\alpha}}Z; \theta(V))]^2\}$$

$$E = \mathbb{E}_{z,x_0}\{[\eta(X_0 + \sqrt{\frac{\Delta_0 + E}{\alpha}}Z; \theta(V)) - X_0]^2\}$$

Same result as the replica computation [KMSSZ12]

Our goal

For non-Bayes optimal problem :

Can we go beyond i.i.d Gaussian A ?

For any convex loss and regularization g, f ?

Our goal

Can we go beyond i.i.d Gaussian F ?

Rotationally invariant matrix

$\mathbf{A} = \mathbf{UDV}^T$, $\mathbf{U}, \mathbf{V}$ Haar distributed, and $\mathbf{D}$ contains singular values with **arbitrary distribution with compact support**.

For any convex loss and regularization g, f ?

Any convex, **separable** g, f.

Build on replica results from [Kabashima07], [RGF09]

Brief recap

What we know so far:

- prove asymptotic errors/replica formula
- beyond Bayes optimal and i.i.d, generic f,g
- no Gaussian interpolation
- message-passing and state evolution

Sketch of proof : key points ...

Key points :

(i) build a sequence whose fixed point solves problem (1)

(ii) asymptotic statistical characterization, match replica prediction

(iii) ensure convergence of the sequence

At the fixed point of the sequence, we will have $\mathbf{x}^*$ and its statistical properties.

Sketch of proof : key points ... and how to handle them

Key points :

(i) Use **vector approximate message-passing** [Rangan et. al. 2019]

(ii) Statistical characterization with **state evolution equations**

(iii) Study the **convergence** of VAMP

Vector approximate message passing [RSF16]

Same intuition as AMP, for rot. inv. matrices

Algorithm 1 VAMP for the SLM

Require: LMMSE estimator $\boldsymbol{g}_2(\boldsymbol{r}_{2k}, \gamma_{2k})$ from (10), denoiser $\boldsymbol{g}_1(\cdot, \gamma_{1k})$, and number of iterations K.

1: Select initial $\boldsymbol{r}_{10}$ and $\gamma_{10} \geq 0$.
2: **for** $k = 0, 1, \ldots, K$ **do**
3: // Denoising
4: $\widehat{\boldsymbol{x}}_{1k} = \boldsymbol{g}_1(\boldsymbol{r}_{1k}, \gamma_{1k}), \quad \alpha_{1k} = \langle \boldsymbol{g}_1'(\boldsymbol{r}_{1k}, \gamma_{1k}) \rangle$
5: $\boldsymbol{r}_{2k} = (\widehat{\boldsymbol{x}}_{1k} - \alpha_{1k}\boldsymbol{r}_{1k})/(1 - \alpha_{1k})$
6: $\gamma_{2k} = \gamma_{1k}(1 - \alpha_{1k})/\alpha_{1k}$
7: // LMMSE estimation
8: $\widehat{\boldsymbol{x}}_{2k} = \boldsymbol{g}_2(\boldsymbol{r}_{2k}, \gamma_{2k}), \quad \alpha_{2k} = \langle \boldsymbol{g}_2'(\boldsymbol{r}_{2k}, \gamma_{2k}) \rangle$
9: $\boldsymbol{r}_{1,k+1} = (\widehat{\boldsymbol{x}}_{2k} - \alpha_{2k}\boldsymbol{r}_{2k})/(1 - \alpha_{2k})$
10: $\gamma_{1,k+1} = \gamma_{2k}(1 - \alpha_{2k})/\alpha_{2k}$
11: **end for**
12: Return $\widehat{\boldsymbol{x}}_{1K}$.

Figure 4: Vector AMP (linear regression)

State evolution for any spectrum

Generalized vector approximate message passing

Algorithm 2 VAMP for the GLM

Require: LMMSE estimators $\boldsymbol{g}_{x2}$ and $\boldsymbol{g}_{z2}$ from (15) or (16), denoisers $\boldsymbol{g}_{x1}$ and $\boldsymbol{g}_{z1}$, and number of iterations K.

Select initial $\boldsymbol{r}_{10}, \boldsymbol{p}_{10}, \gamma_{10} > 0, \tau_{10} > 0$.
for $k = 0, 1, \ldots, K$ **do**
  // Denoising $\boldsymbol{x}$
  $\widehat{\boldsymbol{x}}_{1k} = \boldsymbol{g}_{x1}(\boldsymbol{r}_{1k}, \gamma_{1k}), \quad \alpha_{1k} = \langle \boldsymbol{g}'_{x1}(\boldsymbol{r}_{1k}, \gamma_{1k}) \rangle$
  $\boldsymbol{r}_{2k} = (\widehat{\boldsymbol{x}}_{1k} - \alpha_{1k}\boldsymbol{r}_{1k})/(1 - \alpha_{1k})$
  $\gamma_{2k} = \gamma_{1k}(1 - \alpha_{1k})/\alpha_{1k}$
  // Denoising $\boldsymbol{z}$
  $\widehat{\boldsymbol{z}}_{1k} = \boldsymbol{g}_{z1}(\boldsymbol{p}_{1k}, \tau_{1k}), \quad \beta_{1k} = \langle \boldsymbol{g}'_{z1}(\boldsymbol{p}_{1k}, \tau_{1k}) \rangle$
  $\boldsymbol{p}_{2k} = (\widehat{\boldsymbol{z}}_{1k} - \beta_{1k}\boldsymbol{p}_{1k})/(1 - \beta_{1k})$
  $\tau_{2k} = \tau_{1k}(1 - \beta_{1k})/\beta_{1k}$
  // LMMSE estimation of $\boldsymbol{x}$
  $\widehat{\boldsymbol{x}}_{2k} = \boldsymbol{g}_{x2}(\boldsymbol{r}_{2k}, \boldsymbol{p}_{2k}, \gamma_{2k}, \tau_{2k}), \quad \alpha_{2k} = \langle \boldsymbol{g}'_{x2}(\ldots) \rangle$
  $\boldsymbol{r}_{1,k+1} = (\widehat{\boldsymbol{x}}_{2k} - \alpha_{2k}\boldsymbol{r}_{2k})/(1 - \alpha_{2k})$
  $\gamma_{1,k+1} = \gamma_{2k}(1 - \alpha_{2k})/\alpha_{2k}$
  // LMMSE estimation of $\boldsymbol{z}$
  $\widehat{\boldsymbol{z}}_{2k} = \boldsymbol{g}_{z2}(\boldsymbol{r}_{2k}, \boldsymbol{p}_{2k}, \gamma_{2k}, \tau_{2k}), \quad \beta_{2k} = \langle \boldsymbol{g}'_{z2}(\ldots) \rangle$
  $\boldsymbol{p}_{1,k+1} = (\widehat{\boldsymbol{z}}_{2k} - \beta_{2k}\boldsymbol{p}_{2k})/(1 - \beta_{2k})$
  $\tau_{1,k+1} = \tau_{2k}(1 - \beta_{2k})/\beta_{2k}$
end for
Return $\widehat{\boldsymbol{x}}_{1K}$.

Figure 5: GVAMP (GLM)

VAMP State evolution and replicas with rot. inv. matrices

The state evolution equations are a little thick...

The replica equations are worse ...

They match [TK20], [GAK20], [GAK20+]

Convergence of the sequence

Use convex analysis methods :

- Lipschitz constants
- Lyapunov function (control theory methods)
- use geometrical properties (strong convexity)

For both VAMP and GVAMP:

Convergent sequences for sufficiently strongly convex problems
[GAK20][GAK20+]

Numerical verification : learning a sign teacher

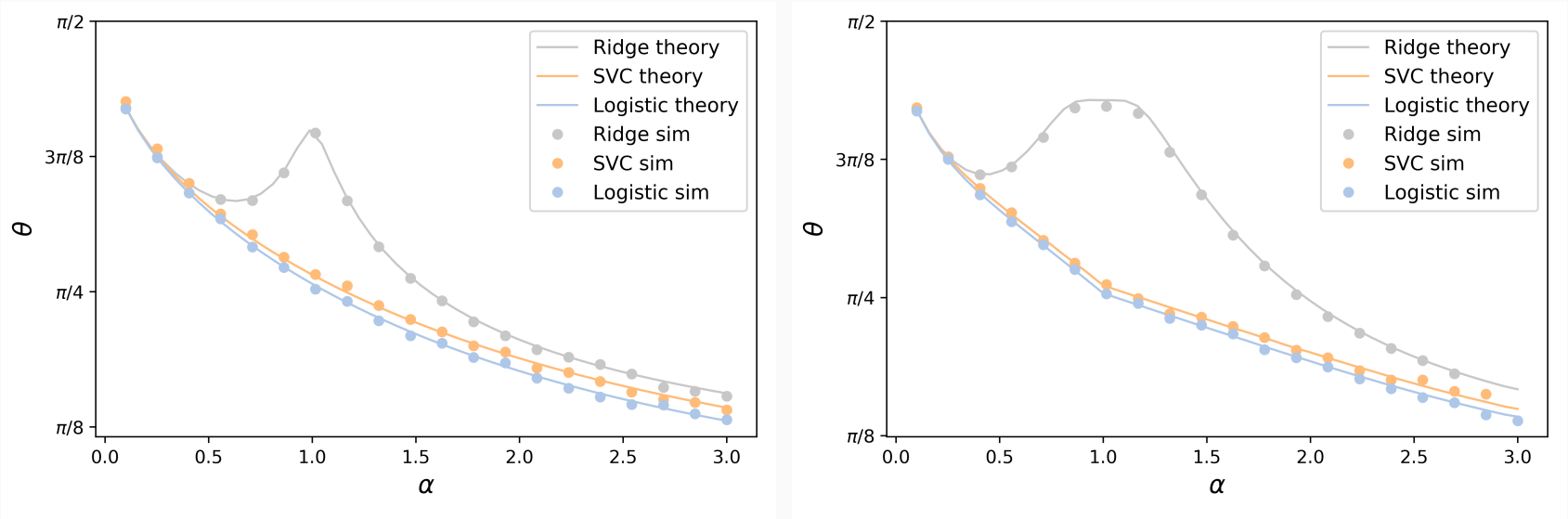

Figure 6: Reconstruction angle $\theta = (\mathbf{x}_0, \mathbf{x}^*)$ as a function of aspect ratio $\alpha = M/N$. **Left :** a Gaussian i.i.d. matrix **Right :** a random orthogonal invariant matrix with a squared uniform density of eigenvalues.

Thank you