Eraser Machine (ϵ-machine)

I. Introduction

- System Inference
  : Given observed configurations, σ obtain their underlying prob. dist. $p(\sigma)$

  ↳ ① reconstructing missing parts

$$p(\sigma_1 | \sigma_2 \sigma_3 \cdots \sigma_n) = \frac{p(\sigma)}{\sum_{\sigma_1} p(\sigma)}$$

  ② generating samples

$$p(\sigma) \longrightarrow \sigma$$

  ③ underlying model, e.g., Hopfield model, Boltzmann machine, Graphical models

$$p(\sigma) = \frac{e^{-E(\sigma)}}{Z} \qquad E(\sigma) = -\sum_i b_i \sigma_i - \sum_{j>k} w_{jk} \sigma_j \sigma_k$$

$$Z = \sum_{\{\sigma\}} e^{-E(\sigma)}$$

- Issue for computing partition function.

  For example, if $n=20$, # of $\sigma = 2^{20} \approx 10^6$

  How to overcome this issue? literature search

- Our idea

  by reweighting data configuration, trivialize to compute partition function.

  For example, at high temperature, $Z = \sum_{\{\sigma\}} e^{-\beta E(\sigma)} \approx 2^n$

  "Attain complete vacuity" by Lao Tzu

  ⇒ develop a very efficient algorithm to obtain underlying model distributions from general observations.

II. Maximum likelihood estimation (MLE)

- Data: $f(\sigma)$ — relative observed # of σ configuration

- Model: $p(\sigma)$

- Likelihood: $L = \prod_{\{\sigma\}} p(\sigma)^{f(\sigma)}$

$p^*(\sigma) = f(\sigma)$, solution maximizing $\log L$

- Specific model

$$p(\sigma) = \frac{e^{-E(\sigma)}}{Z}, \qquad E(\sigma) = -\sum_i b_i \sigma_i - \sum_{j<k} w_{jk} \sigma_j \sigma_k$$

$$Z = \sum_{\{\sigma\}} e^{-E(\sigma)} \qquad = w \cdot \sigma$$

$$\begin{cases} w \equiv (b_i, w_{jk}) \\ \sigma = (\sigma_i, \sigma_j \sigma_k) \end{cases}$$

for example, $n = 2$

$w = (b_1, b_2, w_{12})$

$\sigma = (\sigma_1, \sigma_2, \sigma_1 \sigma_2)$

Then,

$L = L(w)$

$w^* = ?$

① Hopfield solution $\quad w_I^* = \sum_{\{\sigma\}} \sigma_I \cdot f(\sigma) \equiv \langle \sigma_I \rangle_f$

↳ $b_i = \langle \sigma_i \rangle_f$, $\quad w_{jk} = \langle \sigma_j \sigma_k \rangle_f$

② MLE (Boltzmann machine)

$$\left. \frac{\partial \log L}{\partial w_I} \right|_{w = w_I^*} = 0, \qquad I \in \{1, 2, \cdots, n + \tfrac{n(n-1)}{2}\}$$

$$\frac{\partial \log L}{\partial w_I} = -\langle \sigma_I \rangle_f + \langle \sigma_I \rangle_p$$

$$\langle \sigma_I \rangle_f \equiv \sum_{\{\sigma\}} \sigma_I \cdot f(\sigma)$$

$$\langle \sigma_I \rangle_p \equiv \sum_{\{\sigma\}} \sigma_I \cdot \frac{e^{-E(\sigma)}}{Z}$$

$$w_I^{new} = w_I + \alpha \cdot \frac{\partial \log L}{\partial w_I}$$

α ← learning rate

Problem: { Hopfield solution → not accurate
Boltzmann machine → computationally heavy to obtain Z, even intractable for large $n > 20$

③ erasure machine (ϵ-machine)

- reweighting with an arbitrary distribution $q(\sigma)$

$$f(\sigma) \to \tilde{f}(\sigma) \propto f(\sigma) \cdot q(\sigma)^{\epsilon-1} \qquad \tilde{f}(\sigma) = \frac{f(\sigma) \cdot q(\sigma)^{\epsilon-1}}{\sum_{\sigma'} f(\sigma') \cdot q(\sigma')^{\epsilon-1}}$$

with a small parameter $\epsilon \in [0, 1]$.

$$p^*(\sigma) = f(\sigma)$$

$$\tilde{p}^*(\sigma) = \tilde{f}(\sigma)$$

Here, if $q(\sigma) = p^*(\sigma)$,

$$\tilde{p}^*(\sigma) \propto f(\sigma) \cdot f(\sigma)^{\epsilon-1}$$
$$\propto f^{\epsilon}(\sigma)$$
$$\propto [p^*(\sigma)]^{\epsilon}$$

$$\tilde{p}(\sigma) = \frac{e^{-\epsilon E(\sigma)}}{\tilde{Z}}, \quad \tilde{Z} = \sum_{\{\sigma\}} e^{-\epsilon E(\sigma)}$$

tempting to interprete ϵ as the inverse temperature β in statistical mechanics.

But it should be cautious !!

$$P^*(\sigma) = f(\sigma)$$

$$\tilde{P}^*(\sigma) \propto \left[P^*(\sigma)\right]^{\beta}$$ not by simply exponentiating with β

$$\tilde{P}^*(\sigma) \propto P^*(\sigma) \cdot \left[P^*(\sigma)\right]^{\epsilon-1} = \left[P^*(\sigma)\right]^{\epsilon}$$

Now let's work with reweighted data distribution

$$\tilde{L} = \prod_{\{\sigma\}} \tilde{P}(\sigma)^{\tilde{f}(\sigma)} \qquad \text{where } \tilde{P}(\sigma) = \frac{e^{-\epsilon E(\sigma)}}{\tilde{Z}}$$

$$\log \tilde{L} = \sum_{\{\sigma\}} \tilde{f}(\sigma) \cdot \log \tilde{P}(\sigma)$$

$$= \sum_{\{\sigma\}} \tilde{f}(\sigma) \cdot \left[-\epsilon \cdot E(\sigma) - \log \tilde{Z} \right]$$

$$\frac{\partial \log \tilde{L}}{\partial w_I} = \epsilon \langle \sigma_I \rangle_{\tilde{f}} - \epsilon^2 w_I$$

where $\langle \sigma_I \rangle_{\tilde{f}} \equiv \sum_{\{\sigma\}} \sigma_I \cdot \tilde{f}(\sigma)$

① Second term : contributes as a regularizor to prevent w_I from blowing up

② $\epsilon = 1$, $\frac{\partial \log \tilde{L}}{\partial w_I} = 0 \rightarrow$ Hopfield solution

To obtain the second term, we use a small ϵ expansion. (Supplementary material)

$$\tilde{Z} = \sum_{\{\sigma\}} e^{-\epsilon E(\sigma)}$$

$$= \sum_{\{\sigma\}} e^{\sum_i \epsilon b_i \sigma_i + \sum_{j<k} \epsilon w_{jk} \sigma_j \sigma_k}$$

$$= \sum_{\{\sigma\}} \prod_i \cosh(\epsilon b_i) \cdot \left[1 + \sigma_i \tanh(\epsilon b_i)\right] \prod_{j<k} \cosh(\epsilon w_{jk}) \times \left[1 + \sigma_j \sigma_k \tanh(\epsilon w_{jk})\right]$$

$$= 2^n \cdot \prod_i \cosh(\epsilon b_i) \prod_{j<k} \cosh(\epsilon w_{jk}) \cdot \Big[1 + \sum_{\alpha<\beta} \tanh(\epsilon b_\alpha) \cdot \tanh(\epsilon b_\beta) \times \tanh(\epsilon w_{\alpha\beta}) + \sum_{\substack{\alpha<\beta \\ \alpha<\gamma \\ \beta<\gamma}} \tanh(\epsilon w_{\alpha\beta}) \cdot \tanh(\epsilon w_{\alpha\gamma}) \times \tanh(\epsilon w_{\beta\gamma}) + \mathcal{O}(\epsilon^4) \Big]$$

- $\log \tilde{Z} = n \cdot \log 2 + \sum_i \log \cosh(\epsilon b_i) + \sum_{j<k} \log \cosh(\epsilon w_{jk}) + \mathcal{O}(\epsilon^3)$
- $\delta \cdot \log \tilde{Z} = \sum_i \tanh(\epsilon b_i) \cdot \epsilon \cdot \delta b_i + \sum_{j<k} \tanh(\epsilon w_{jk}) \cdot \epsilon \, \delta w_{jk} + \mathcal{O}(\epsilon^2)$

  $\simeq \sum_i \epsilon^2 b_i \, \delta b_i + \sum_{j<k} \epsilon^2 w_{jk} \, \delta w_{jk} = \sum_I \epsilon^2 w_I \cdot \delta w_I$
- $\dfrac{\partial \log \tilde{Z}}{\partial w_I} = \epsilon^2 w_I$

Algorithm

① reweight $\tilde{f}(\sigma) = \dfrac{f(\sigma) \cdot q(\sigma)^{\epsilon-1}}{\sum_{\{\sigma'\}} f(\sigma') \cdot q^{\epsilon-1}(\sigma')}$ for $q(\sigma)$ initially $q(\sigma) = \dfrac{e^{-E(\sigma)}}{Z}$

$E(\sigma) = -w \cdot \sigma$ ← random w

② $\langle O_I \rangle_{\tilde{f}} = \sum_{\{\sigma\}} O_I \cdot \tilde{f}(\sigma)$

③ $\dfrac{\partial \log \tilde{L}}{\partial w_I} = \epsilon \langle O_I \rangle_{\tilde{f}} - \epsilon^2 w_I$

④ Update $w_I^{new} = w_I + \alpha \cdot \dfrac{\partial \log \tilde{L}}{\partial w_I} = w_I + \alpha' \left(\langle O_I \rangle_{\tilde{f}} - \epsilon w_I \right)$

$\alpha' = \alpha \cdot \epsilon$

⑤ Iterate this until $D_{KL}(\tilde{f} \| \tilde{p})$ minimizes.

$\tilde{p}(\sigma) = \dfrac{e^{-\epsilon E(\sigma)}}{\tilde{Z}}$

$$D_{KL}(\tilde{f} \| \tilde{p}) = \sum_{\{\sigma\}} \tilde{f}(\sigma) \cdot \log \frac{\tilde{f}(\sigma)}{\tilde{p}(\sigma)}$$

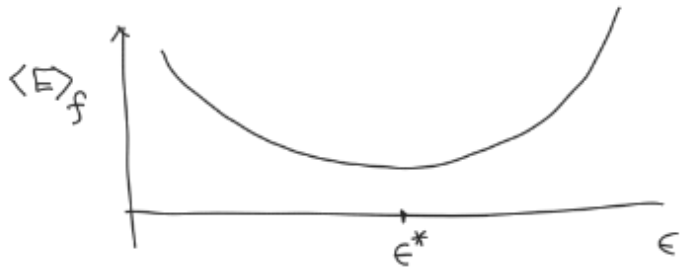
$$= \sum_{\{\sigma\}} \tilde{f}(\sigma) \cdot \log \tilde{f}(\sigma) - \tilde{f}(\sigma) \cdot \log \tilde{p}(\sigma)$$

$$= \sum_{\{\sigma\}} \tilde{f}(\sigma) \cdot \log \tilde{f}(\sigma) - \tilde{f}(\sigma) \cdot \left[-\epsilon E(\sigma) - \log \tilde{Z} \right]$$

$$= -S + \epsilon \langle E \rangle_{\tilde{f}} + \log \tilde{Z}$$

$$\simeq -S + \epsilon \langle E \rangle_{\tilde{f}} + n \cdot \log 2 + \sum_{I} \log \cosh(\epsilon w_I)$$

Optimal ϵ



$$\langle E \rangle_p = \sum_{\{\sigma\}} E(\sigma) \cdot p(\sigma), \quad p(\sigma) = \frac{e^{-E(\sigma)}}{Z}, \quad E(\sigma) = w \cdot \sigma$$

if $w = w^{true}$

$\Rightarrow E(\sigma) = E^{true}(\sigma)$ and $p(\sigma) = f(\sigma)$

Then, $\langle E \rangle_p = \langle E \rangle_f$

if the inferred values of $\langle E \rangle_f$ do not vary for similar (values of) hyperparameter ϵ, this means that $\langle E \rangle_f = \langle E_{true} \rangle_f$ is inferred correctly.

Elaborate this argument!

random ϵ

$$\epsilon = \frac{\text{random}}{\max[1, w]}$$

Fig2. Performance of ϵ-machine

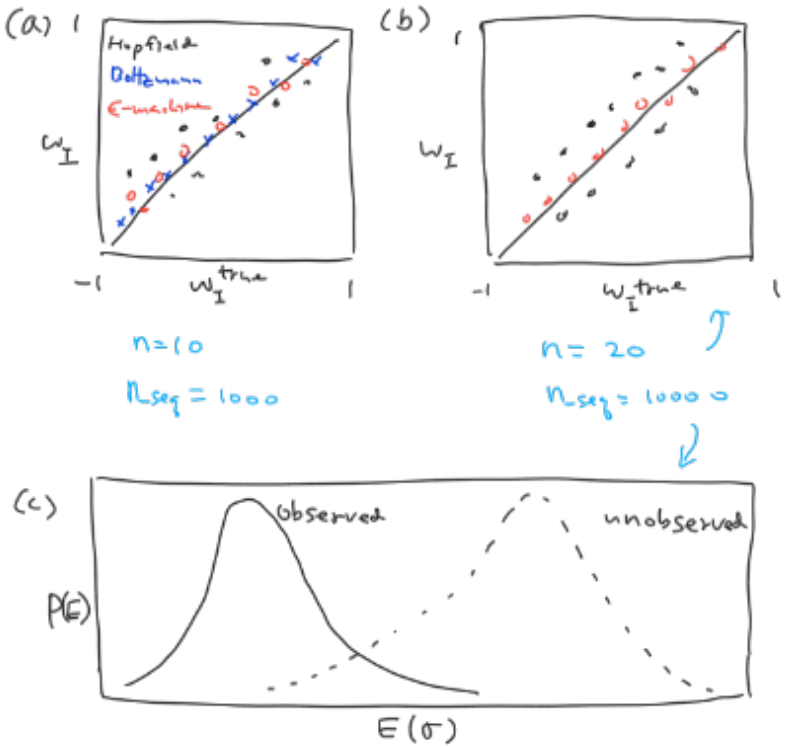
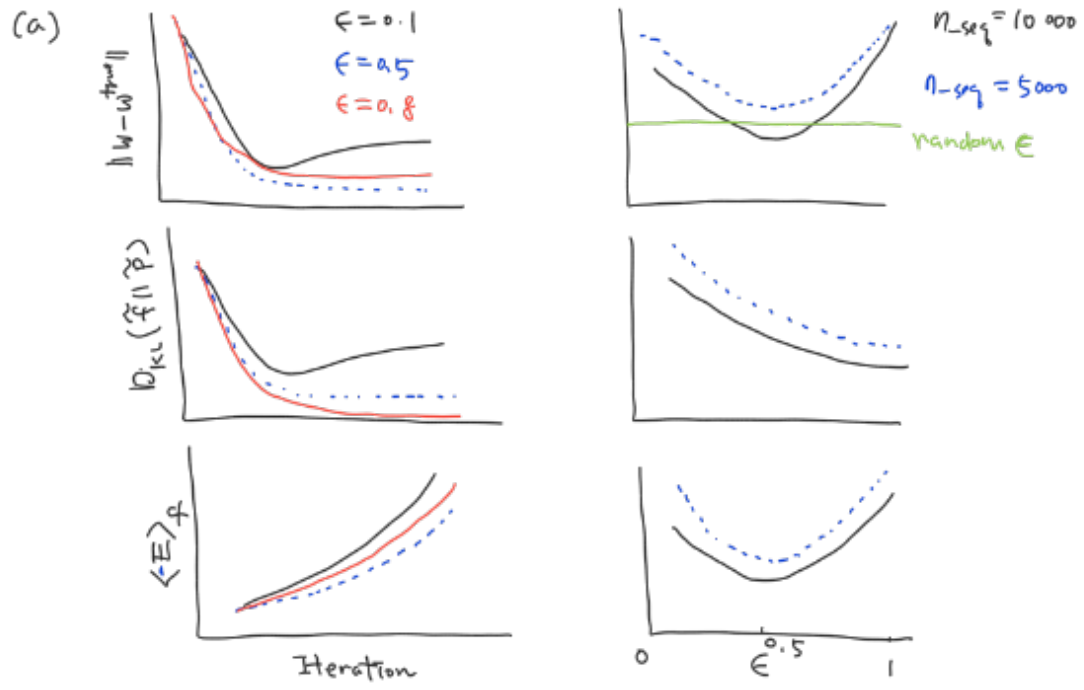


Fig.1 Learning of ϵ-machine

n=20,



Ⅲ. Discussions

- Applications (e.g. equilibrium inference of neural networks)
- Other ways to avoid computing exact partition function.

마지막 수정: 오후 12:34