

Deep Message Passing

Around the middle of the XVII century, a young man called Baruch Spinoza was dealing with a problem tougher than yours: finding the essence of God Himself.

CONTENTS

I	Introduction	2
II	Learning through message passing	2
III	Belief Propagation	2
	ABP updates	2
B	Experiments on MNIST	3
1	Varying batch-size	3
2	Varying ρ	3
3	Varying <i>maxiters</i>	3
4	Varying r	3
5	Varying damping	3
6	Varying dataset size	3
7	Varying initial weights	3
IV	Approximate Message Passing	3
A	Experiments on MNIST	4
1	Varying batch-size	4
2	Varying ρ	4
3	Varying <i>maxiters</i>	4
4	Varying r	4
5	Varying dataset size	4

I. INTRODUCTION

II. LEARNING THROUGH MESSAGE PASSING

TODO ENTIRE SECTION.

In this deep inference problem, we assume that a signal with prior P^{in} is fed to a deep feedforward networks with $L + 1$ layers of weights $\mathbf{W}^\ell \in \mathbb{R}^{N_{\ell+1} \times N_\ell}$, $\ell = 0, \dots, L$ and biases $\mathbf{b}^\ell \in \mathbb{R}^{N_{\ell+1}}$. The signal is propagated through stochastic neuron layers described by probability distributions P^ℓ conditioned on the preactivations, therefore we have the following Markov chain:

$$\mathbf{x}^0 \sim P^{\text{in}} \quad (1)$$

$$\mathbf{x}^{\ell+1} \sim P^{\ell+1} \left(\bullet \mid \mathbf{W}^\ell \mathbf{x}^\ell + \mathbf{b}^\ell \right) \quad \ell = 0, \dots, L \quad (2)$$

Only $\mathbf{y} = \mathbf{x}^{L+1}$ is observed, and the task is to reconstruct the original signal \mathbf{x}^0 . The posterior distribution $p(\mathbf{x}^{0:L}) = P(\mathbf{x}^{0:L} \mid \mathbf{x}^{L+1} = \mathbf{y})$ reads

$$p(\mathbf{x}^{0:L}) \propto \prod_{\ell=0}^L \prod_{k=1}^{N_{\ell+1}} P_k^{\ell+1} \left(x_k^{\ell+1} \mid \sum_{i=1}^{N_\ell} W_{ki}^\ell x_i^\ell \right) \prod_{k=1}^{N_0} P_k^{\text{in}}(x_k^0), \quad (3)$$

Typical channels are given by deterministic elementwise activation function $f_\ell(z)$ (e.g. $f_\ell(z) = \text{sign}(x)$ or $f_\ell(z) = \text{relu}(z) = \max(0, z)$), combined with Gaussian additive pre-activation noise with variance σ^2 . In such cases we have

$$P_k^\ell(x \mid z) = \int D\xi \delta(x - f_\ell(z + \sigma_k^\ell \xi))$$

We also call $\alpha_\ell = N_{\ell+1}/N_\ell$ the layer *expansion ratio*.

III. BELIEF PROPAGATION

A. BP updates

The AMP equations have been derived from the BP equation in the Appendix. First we introduce the neuron scalar entropy functions:

$$\varphi_k^0(B, A) = \log \int dx e^{-\frac{1}{2}A^2x^2+Bx} P_k^{\text{in}}(x) \quad (4)$$

$$\varphi_k^\ell(B, A, \omega, V) = \log \int dx dz e^{-\frac{1}{2}A^2x^2+Bx} P_k^\ell(x \mid z) e^{-\frac{(\omega-z)^2}{2V}} \quad \ell = 1, \dots, L \quad (5)$$

$$\varphi_k^{L+1}(\omega, V, y) = \log \int dz P_k^{L+1}(y \mid z) e^{-\frac{(\omega-z)^2}{2V}} \quad (6)$$

$$\psi_{ki}^\ell(H, G) = \log \int dw e^{-\frac{1}{2}G^2w^2+Hw} P_{ki}^\ell(w) \quad (7)$$

For convenience define $\varphi_i^{0,t} = \varphi_i^0(B_i^{0,t}, A_i^{0,t})$ and $\varphi_i^{\ell,t} = \varphi_i^\ell(B_i^{\ell,t}, A_i^{\ell,t}, \omega_i^{\ell-1,t}, V_i^{\ell-1,t})$ and $\varphi_i^{L+1,t} = \varphi_i^{L+1}(\omega_i^{L,t}, V_i^{L,t}, y_i)$.

Then, we can decompose the BP update rules in a forward and a backward step.

Forward pass. As the initial condition for the iterations, we set to zero the following quantities: $B_i^{\ell,t=0} = 0$, $A_i^{\ell,t=0} = 0$ and $g_k^{\ell,t=0} = 0$. The following iterations hold at time $t \geq 1$. In the FORWARD pass, starting from $\ell = 0$ and up to $\ell = L$, we have

$$\hat{x}_{ia \rightarrow k}^{\ell,t} = \partial_B \varphi_{ia \rightarrow k}^\ell \left(B_{ia \rightarrow k}^{\ell,t-1}, A_{ia}^{\ell,t-1}, \omega_{ia}^{\ell-1,t}, V_{ia}^{\ell-1,t} \right) \quad (8)$$

$$\Delta_{ia \rightarrow k}^{\ell+1,t} = \partial_B^2 \varphi_{ia \rightarrow k}^{\ell+1,t} \quad (9)$$

$$m_{ki \rightarrow a}^{\ell,t} = \partial_H \psi_{ki}^\ell (H_{ki \rightarrow a}^{t-1}, G_{ki}^{t-1}) \quad (10)$$

$$\sigma_{ki \rightarrow a}^{\ell,t} = \partial_H^2 \psi_{ki}^\ell (H_{ki \rightarrow a}^{t-1}, G_{ki}^{t-1}) \quad (11)$$

$$V_{ka}^{\ell,t} = \sum_i \left(\left(m_{ki \rightarrow a}^{\ell,t} \right)^2 \Delta_{ia \rightarrow k}^{\ell,t} + \Sigma_{ki \rightarrow a}^{\ell,t} (\hat{x}_{ia \rightarrow k}^{\ell,t})^2 + \sigma_{ki \rightarrow a}^{\ell,t} \Delta_{ia \rightarrow k}^{\ell,t} \right) \quad (12)$$

$$\omega_{ka \rightarrow i}^{\ell,t} = \sum_{i' \neq i} m_{ki \rightarrow a}^{\ell,t} \hat{x}_{ia \rightarrow k}^{\ell,t} \quad (13)$$

Here V^ℓ and ω^ℓ are computed as a function of the previous layer values $V^{\ell-1}$ and $\omega^{\ell-1}$.

Backward pass. In the BACKWARD sweep, starting from $\ell = L$ and down to $\ell = 0$, we have

$$g_{ka \rightarrow i}^{\ell,t} = \partial_\omega \varphi_{ka \rightarrow i}^{\ell+1} \left(B_{ka}^{\ell+1,t}, A_{ka}^{\ell+1,t}, \omega_{ka \rightarrow i}^{\ell,t}, V_{ka}^{\ell,t} \right) \quad (14)$$

$$\Gamma_{ka \rightarrow i}^{\ell,t} = -\partial_\omega^2 \varphi_{ka \rightarrow i}^{\ell+1,t} \quad (15)$$

$$A_{ia}^{\ell,t} = \sum_k \left((m_{ki \rightarrow a}^{\ell,t})^2 + \sigma_{ki \rightarrow a}^{\ell,t} \right) \Gamma_{ka \rightarrow i}^{\ell,t} - \sigma_{ki \rightarrow a}^{\ell,t} \left(g_{ka \rightarrow i}^{\ell,t} \right)^2 \quad (16)$$

$$B_{ia \rightarrow k}^{\ell,t} = \sum_{k' \neq k} m_{k'i \rightarrow a}^{\ell,t} g_{k'a \rightarrow i}^{\ell,t} \quad (17)$$

$$G_{ki}^{\ell,t} = \sum_a \left((\hat{x}_{ia \rightarrow k}^{\ell,t})^2 + \Delta_{ia \rightarrow k}^{\ell,t} \right) \Gamma_{ka \rightarrow i}^{\ell,t} - \Delta_{ia \rightarrow k}^{\ell,t} \left(g_{ka \rightarrow i}^{\ell,t} \right)^2 \quad (18)$$

$$H_{ki \rightarrow a} = \sum_{a' \neq a} \hat{x}_{ia' \rightarrow k}^{\ell,t} g_{ka' \rightarrow i}^{\ell,t} \quad (19)$$

Notice that A^ℓ and B^ℓ are computed as a function of the $A^{\ell+1}, B^{\ell+1}$ of the layer above, with the initial condition given by the output $\mathbf{x}^{L+1} = \mathbf{y}$ on the top layer.

B. Experiments on MNIST

1. Varying batch-size

2. Varying ρ

3. Varying maxiters

4. Varying r

5. Varying damping

6. Varying dataset size

7. Varying initial weights

IV. APPROXIMATE MESSAGE PASSING

Forward pass. As the initial condition for the iterations, we set to zero the following quantities: $B_i^{\ell,t=0} = 0, A_i^{\ell,t=0} = 0$ and $g_k^{\ell,t=0} = 0$. The following iterations hold at time $t \geq 1$. In the FORWARD pass, starting from $\ell = 0$ and up to $\ell = L$, we have

$$\hat{x}_{ia}^{\ell,t} = \partial_B \varphi_{ia}^{\ell,t-} \quad (20)$$

$$\Delta_{ia}^{\ell,t} = \partial_B^2 \varphi_{ia}^{\ell,t-} \quad (21)$$

$$m_{ki}^{\ell,t} = \partial_H \psi_{ki}^{\ell,t-} \quad (22)$$

$$\sigma_{ki}^{\ell,t} = \partial_H^2 \psi_{ki}^{\ell,t-} \quad (23)$$

$$V_{ka}^{\ell,t} = \sum_i \left(\left(m_{ki}^{\ell,t} \right)^2 \Delta_{ia}^{\ell,t} + \sigma_{ki}^{\ell,t} (\hat{x}_{ia}^{\ell,t})^2 + \sigma_{ki}^{\ell,t} \Delta_{ia}^{\ell,t} \right) \quad (24)$$

$$\omega_{ka}^{\ell,t} = \sum_i m_{ki}^{\ell,t} \hat{x}_{ia}^{\ell,t} + TODO : onsagsize(x)er \quad (25)$$

Here V^ℓ and ω^ℓ are computed as a function of the previous layer values $V^{\ell-1}$ and $\omega^{\ell-1}$.
Backward pass. In the BACKWARD sweep, starting from $\ell = L$ and up to $\ell = 0$, we have

$$g_{ka}^{\ell,t} = \partial_\omega \varphi_{ka}^{\ell+1,t} \quad (26)$$

$$\Gamma_{ka}^{\ell,t} = -\partial_\omega^2 \varphi_{ka}^{\ell+1,t} \quad (27)$$

$$A_{ia}^{\ell,t} = \sum_k \left((m_{ki}^{\ell,t})^2 + \sigma_{ki}^{\ell,t} \right) \Gamma_{ka}^{\ell,t} - \sigma_{ki}^{\ell,t} \left(g_{ka}^{\ell,t} \right)^2 \quad (28)$$

$$B_{ia}^{\ell,t} = \sum_k m_{ki}^{\ell,t} g_{ka}^{\ell,t} + TODO : onsager \quad (29)$$

$$G_{ki}^{\ell,t} = \sum_a \left((\hat{x}_{ia}^{\ell,t})^2 + \Delta_{ia}^{\ell,t} \right) \Gamma_{ka}^{\ell,t} - \Delta_{ia}^{\ell,t} \left(g_{ka}^{\ell,t} \right)^2 \quad (30)$$

$$H_{ki} = \sum_a \hat{x}_{ia}^{\ell,t} g_{ka}^{\ell,t} + TODO : onsager \quad (31)$$

Notice that A^ℓ and B^ℓ are computed as a function of the $A^{\ell+1}, B^{\ell+1}$ of the layer above, with the initial condition given by the output $\mathbf{x}^{L+1} = \mathbf{y}$ on the top layer.

A. Experiments on MNIST

1. Varying batch-size

2. Varying ρ

3. Varying maxiters

4. Varying r

5. Varying dataset size