Dual Propagation

Accelerating Contrastive Hebbian Learning with Dyadic Neurons

Rasmus Høier, D. Staudt, Christopher Zach

Chalmers University of Technology
Email: hier@chalmers.se
Code repo: github.com/rasmuskh/dual-propagation
Slideshow: github.com/rasmuskh/dualprop-slideshow

Motivation

Contrastive Hebbian learning (CHL, Xie and Seung (2003)) and equilibrium propagation (EP, Scellier and Bengio (2017))

- Biological plausibility
- Neuromorphic computing

Motivation

Contrastive Hebbian learning (CHL, Xie and Seung (2003)) and equilibrium propagation (EP, Scellier and Bengio (2017))

- Biological plausibility
- Neuromorphic computing

However, certain issues limit their usability.

1. First order gradient estimate
2. Extremely slow inference
3. Two globally synchronized inference phases

A dyadic objective

Select $\alpha \in [0, 1]$

$$\mathcal{L}_\alpha(\theta) := \min_{z^+} \max_{z^-} \alpha \ell(z_L^+) + (1-\alpha)\ell(z_L^-)$$
$$+ \sum_{k=1}^{L} \tfrac{1}{\beta_k}\big(G_k(z_k^+) - G_k(z_k^-) + (z_k^- - z_k^+)^\top W_{k-1}(\alpha z_{k-1}^+ + (1-\alpha) z_{k-1}^-)\big)$$

Linear units: $G_k = \frac{1}{2}\|\cdot\|^2$

ReLU units: $G_k = \frac{1}{2}\|\cdot\|^2 + \imath_{\geq 0}(\cdot)$

Softmax layer: $G_k = -H(\cdot)$

Choice of α

- $\alpha = 1$ and $\alpha = 0$:
  - LPOM-like (Li, Fang, and Lin 2019; Zach 2021) pure minimization objectives
  - In general slow iterative inference

Choice of α

- $\alpha = 1$ and $\alpha = 0$:
  - LPOM-like (Li, Fang, and Lin 2019; Zach 2021) pure minimization objectives
  - In general slow iterative inference
- $\alpha \in (0, 1) \setminus \frac{1}{2}$
  - Fast inference
  - Lacks convergence guarantee

Choice of α

- $\alpha = 1$ and $\alpha = 0$:
  - LPOM-like (Li, Fang, and Lin 2019; Zach 2021) pure minimization objectives
  - In general slow iterative inference
- $\alpha \in (0, 1) \setminus \frac{1}{2}$
  - Fast inference
  - Lacks convergence guarantee
- $\alpha = 1/2$:
  - Fast inference
  - Convergence guarantee (details in paper)

Choice of α

- $\alpha = 1$ and $\alpha = 0$:
  - LPOM-like (Li, Fang, and Lin 2019; Zach 2021) pure minimization objectives
  - In general slow iterative inference
- $\alpha \in (0, 1) \setminus \frac{1}{2}$
  - Fast inference
  - Lacks convergence guarantee
- $\alpha = 1/2$:
  - Fast inference
  - Convergence guarantee (details in paper)
- Assume $\alpha = 1/2$ from now on.

Connection to CHL and EP

Defining layerwise costs E_k as in CHL

$$E_k(z_k, z_{k-1}) := G_k(z_k) - z_k^T W_{k-1} z_{k-1}$$

and short-hand notation

$$\bar{z}_k := \tfrac{1}{2}(z_k^+ + z_k^-)$$

allows rewriting of $\mathcal{L}_{\frac{1}{2}}$:

$$\mathcal{L}_{\frac{1}{2}}(\theta) = \min_{z^+} \max_{z^-} \tfrac{1}{2}\ell(z_L^+) + \tfrac{1}{2}\ell(z_L^-) + \sum_{k=1}^{L} \tfrac{1}{\beta_k}\big(E(z_k^+, \bar{z}_{k-1}) - E(z_k^-, \bar{z}_{k-1})\big)$$

Similar to CHL and EP objectives but z_k^+ and z_k^- are inferred simultaneously and "tethered" via $\bar{z}_{k-1}$.

Inference

- Neurons within layers are not coupled.
- Fully optimize $\mathcal{L}_{\frac{1}{2}}$ with respect to z_k^+ and z_k^- in a single block-coordinate descent step!

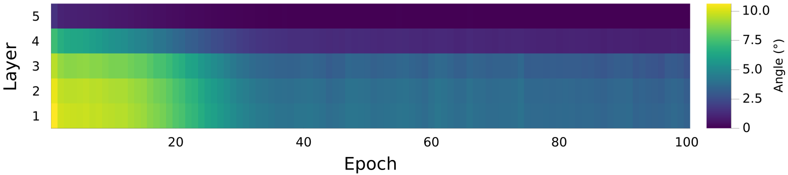
$$z_k^{\pm} \leftarrow f_k \left(\underbrace{\frac{1}{2} W_{k-1}(z_{k-1}^+ + z_{k-1}^-)}_{= W_{k-1}\bar{z}_{k-1}} \pm \frac{\beta_k}{2\beta_{k+1}} W_k^\top (z_{k+1}^+ - z_{k+1}^-) \right)$$

- Same runtime as BP and >100X faster than EP and CHL.

- Fully local contrastive Hebbian gradient

$$\frac{\partial}{\partial W_{k-1}}\mathcal{L}_{\frac{1}{2}} = \frac{1}{\beta_k}\left(\frac{\partial E(z_k^+, \bar{z}_{k-1})}{\partial_{W_{k-1}}} - \frac{\partial E(z_k^-, \bar{z}_{k-1})}{\partial_{W_{k-1}}}\right)$$
$$= \frac{1}{\beta_k}\left(z_k^- - z_k^+\right)\bar{z}_{k-1}^\top.$$

- Second order gradient estimate
- Discounting not needed ($\beta_k = 1 \ \forall \ k$ employed in experiments)
- Excellent gradient alignment between BP and DP



Experiments: MLP[1] (MNIST)

- **DP: runtime-efficient (forwards + backwards)**
- MS-DP: Multiple steps of inference and weight updates per datapoint
- L-DP: "Lazily" let old states persist providing potentially harmful feedback.
- **P-DP: parallel neuron updates**
- **R-DP-100: Random sequence of 100 layer-wise updates**

Method	BP	DP	MS-DP	L-DP	P-DP	R-DP-100
Test acc (%)	98.45 ± 0.04	98.43 ± 0.03	98.40 ± 0.02	98.42 ± 0.07	98.47 ± 0.04	98.48 ± 0.11

[1] MLP architecture: 784-1000-1000-1000-1000-10

Experiments: VGG16

DP matches back-propagation both in terms of runtime and accuracy.

Method		BP	DP	KP-DP†	EP*	DTP*
CIFAR10	Top-1	92.26 ± 0.23	92.30 ± 0.11	91.84 ± 0.11	88.6 ± 0.2	89.38 ± 0.20
CIFAR100	Top-1	69.63 ± 0.24	69.57 ± 0.51	70.40 ± 0.25	61.6 ± 0.1	–
	Top-5	88.13 ± 0.22	88.36 ± 0.13	88.57 ± 0.15	86.0 ± 0.1	–
ImageNet32x32	Top-1	41.28 ± 0.19	41.48 ± 0.19	–	36.5 ± 0.3	36.81
	Top-5	64.89 ± 0.11	64.90 ± 0.13	–	60.8 ± 0.4	60.54

(†) Dual propagation with Kolen-Pollack learning of feedback weights.

(*) High computational costs limit EP (Laborieux and Zenke 2022) and DTP (Ernoult et al. 2022) to 6-7 layer VGG-like networks.

Conclusion

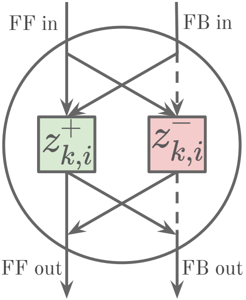
- DP matches BP both in terms of accuracy and runtime.
    - However, unlike BP neurons can operate asynchronously
    - Parallel and random update schemes viable
- Unlike previous CHL methods DP computes errors across compartments rather than across time.
    - Single phase CHL
    - Layerwise closed-form inference
    - Second order gradient estimate

Future work: Continuous inference and learning in a streaming setting.

Code repo: github.com/rasmuskh/dual-propagation
Slideshow: github.com/rasmuskh/dualprop-slideshow

Extra slides

Dyadic neurons



$$z_k^{\pm} \leftarrow f_k\left(\underbrace{\tfrac{1}{2}\, W_{k-1}(z_{k-1}^{+} + z_{k-1}^{-})}_{\text{FF in}} \pm \underbrace{\tfrac{\beta_k}{2\beta_{k+1}}\, W_k^{\top}(z_{k+1}^{+} - z_{k+1}^{-})}_{\text{FB in}}\right)$$

$$\text{FF out} = \tfrac{1}{2}(z_{k,i}^{+} + z_{k,i}^{-}) = \bar{z}_{k,i}$$

$$\text{FB out} = \tfrac{1}{2}(z_{k,i}^{+} - z_{k,i}^{-})$$

Comparisson of contrastive objectives

Dual propagation

$$\mathcal{L}_{\frac{1}{2}}(\theta) = \min_{z^+} \max_{z^-} \tfrac{1}{2}\ell(z_L^+) + \tfrac{1}{2}\ell(z_L^-) + \sum_{k=1}^{L} \tfrac{1}{\beta_k}\Big(E_k(z_k^+, \bar{z}_{k-1}) - E_k(z_k^-, \bar{z}_{k-1})\Big)$$

Contrastive Hebbian learning (Xie and Seung 2003)

$$\mathcal{L}_{CHL}(\theta) = \min_{\hat{z}} \max_{\check{z}} \sum_{k=1}^{L} \gamma^{k-L}(E_k(\hat{z}_k, \hat{z}_{k-1}) - E_k(\check{z}_k, \check{z}_{k-1}))$$

Equilibrium propagation (Scellier and Bengio 2017)

$$\mathcal{L}_{EP}(\theta) = \min_{z^\beta} \max_{z^0} \beta\ell(z_L^+) + \sum_{k=1}^{L}\Big(E_k(z_k^\beta, z_{k-1}^\beta) - E_k(z_k^0, z_{k-1}^0)\Big)$$

Back-of-the-envelope runtime comparisson

- CIFAR10 runtime per epoch in seconds for different implementations.
- Estimates based on numbers from supplemental material of (Laborieux and Zenke 2022) and (Ernoult et al. 2022).

Method	DP	H-EP	DTP
Seconds/epoch	3.5	1700	240
Layers	16	7	6

- Differences in hardware, software frameworks, model size and batch size make this comparison indicative only.

Ernoult, Maxence M et al. (2022). “Towards scaling difference target propagation by learning backprop targets”. In: *International Conference on Machine Learning*. PMLR, pp. 5968–5987.

Laborieux, Axel and Friedemann Zenke (2022). “Holomorphic Equilibrium Propagation Computes Exact Gradients Through Finite Size Oscillations”. In: *arXiv preprint arXiv:2209.00530*.

Li, Jia, Cong Fang, and Zhouchen Lin (2019). “Lifted proximal operator machines”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33, pp. 4181–4188.

Scellier, Benjamin and Yoshua Bengio (2017). “Equilibrium propagation: Bridging the gap between energy-based models and backpropagation”. In: *Frontiers in computational neuroscience* 11, p. 24.

Xie, Xiaohui and H Sebastian Seung (2003). “Equivalence of backpropagation and contrastive Hebbian learning in a layered network”. In: *Neural computation* 15.2, pp. 441–454.

Zach, Christopher (2021). “Bilevel Programs Meet Deep Learning: A Unifying View on Inference Learning Methods”. In: *arXiv preprint arXiv:2105.07231*.