Small Steps Towards Biologically Plausible Deep Learning

Yoshua Bengio

11 December 2015

NIPS'2015

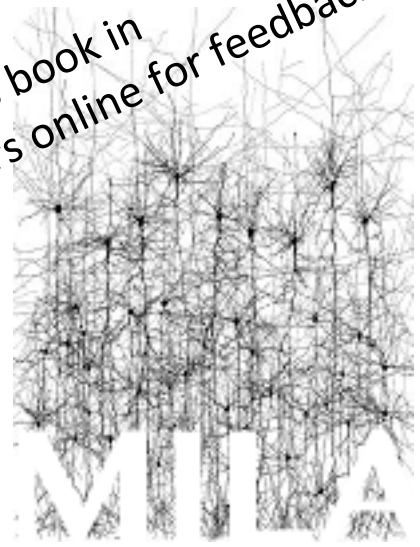
Statistical Methods for

Neural Systems Workshop

CIFAR
CANADIAN INSTITUTE FOR ADVANCED RESEARCH

Université de Montréal

Deep Learning, MIT Press book in preparation, draft chapters online for feedback

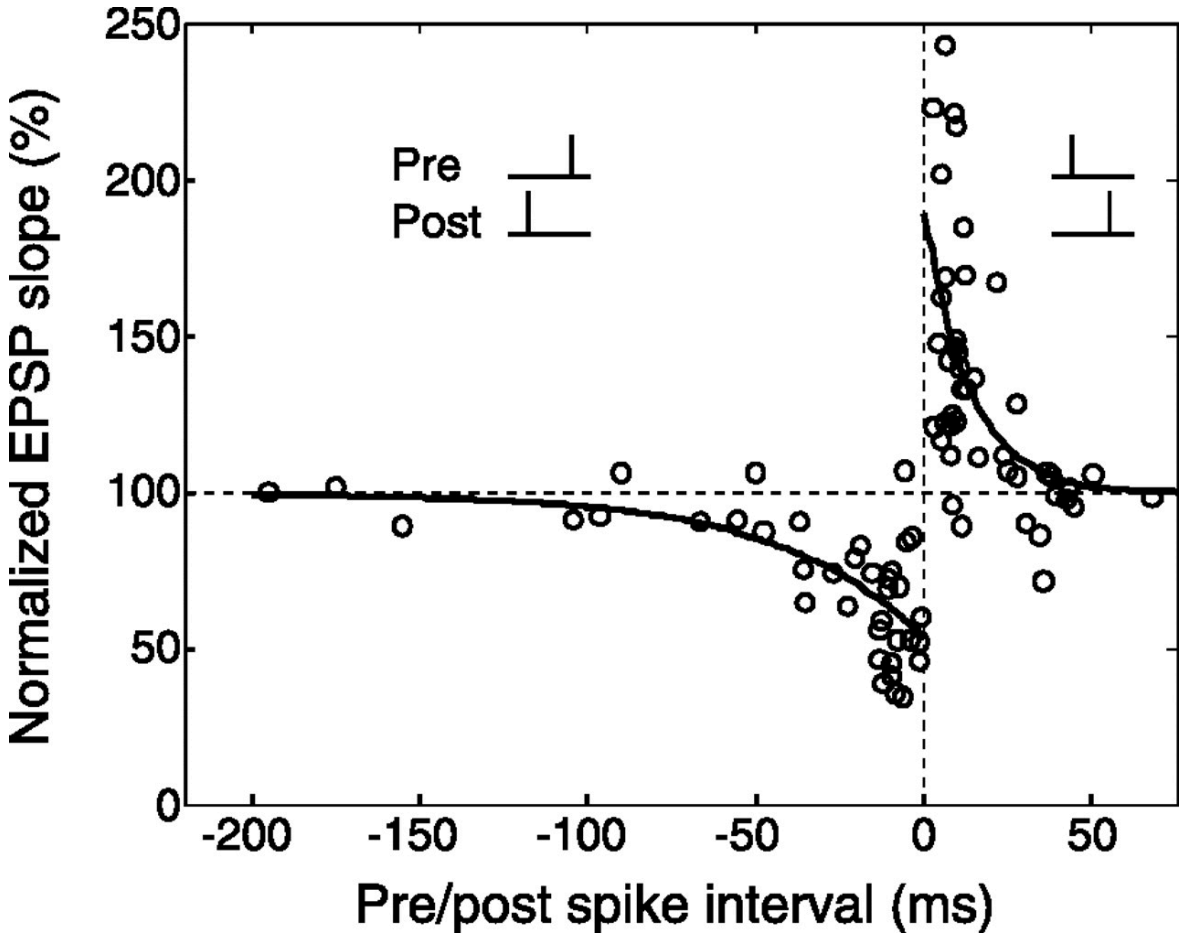


Central Issue in Deep Learning: Credit Assignment

- What should hidden layers do?
- Established approaches:
  - Backpropagation
  - Stochastic relaxation in Boltzmann machines
- Are these related?
- How does the brain do it?

What is the brain's learning algorithm? Cue: Spike-Timing Dependent Plasticity

- Observed throughout the nervous system, especially in cortex
- STDP: weight increases if post-spike just after pre-spike, decreases if just before.
- Timing counted only if spike on only one side within window



Hypothesis #1

Inspired by hypothesis from Hinton 2007 (Deep Learning Workshop talk)

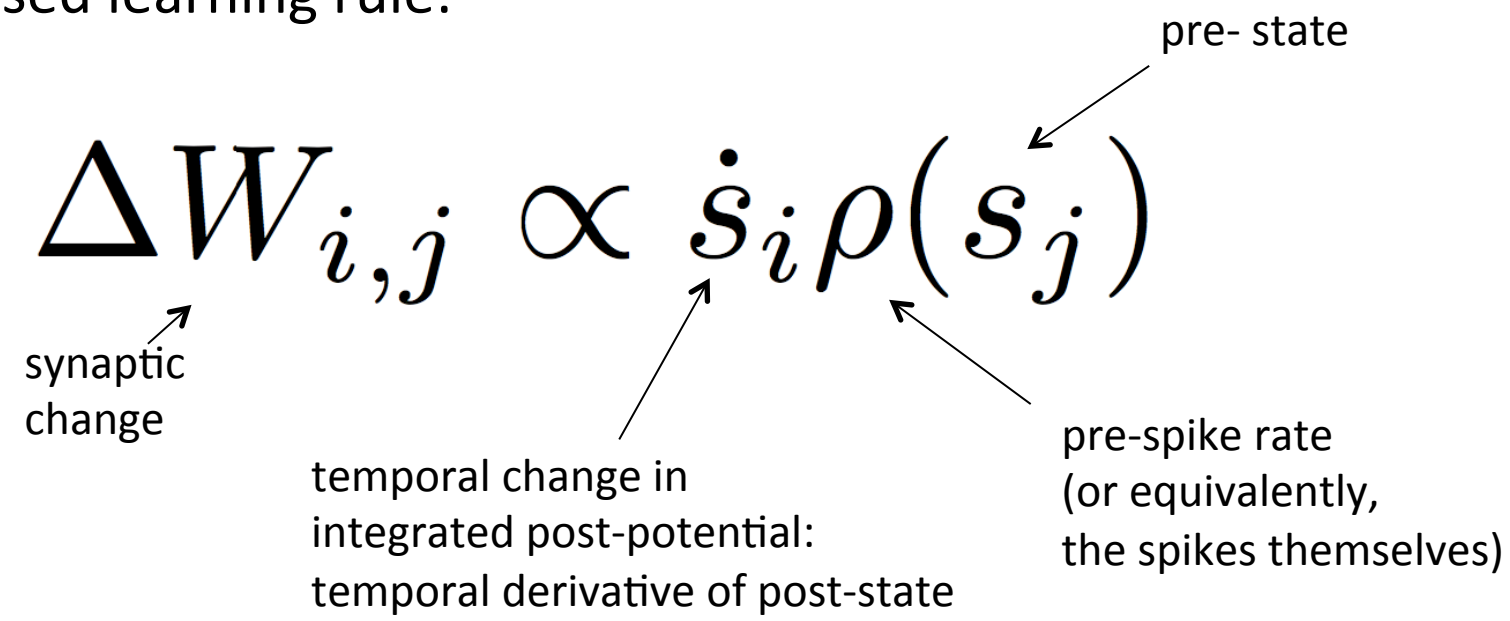
STDP is explained by a learning rule with this form:

Weight change proportional to post-synaptic rate of change times pre-synaptic spike.

Proposed Interpretation of STDP

Inspired by Hinton 2007 (Deep Learning Workshop talk)

- Let *s* = continuous-valued state of all neurons
  = soma integrated voltage potential (avg out effect of spikes)
- Proposed learning rule:



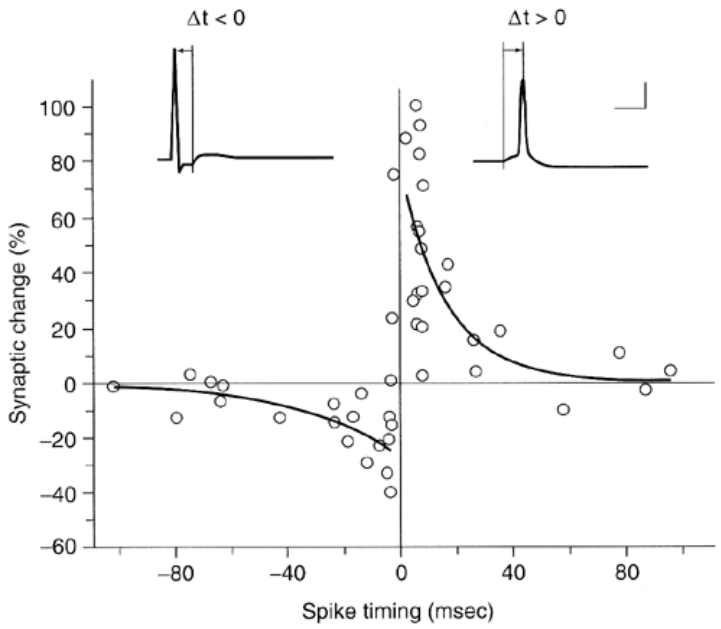
Happy Coincidence

In simulations, this learning rule fits the classical STDP curves

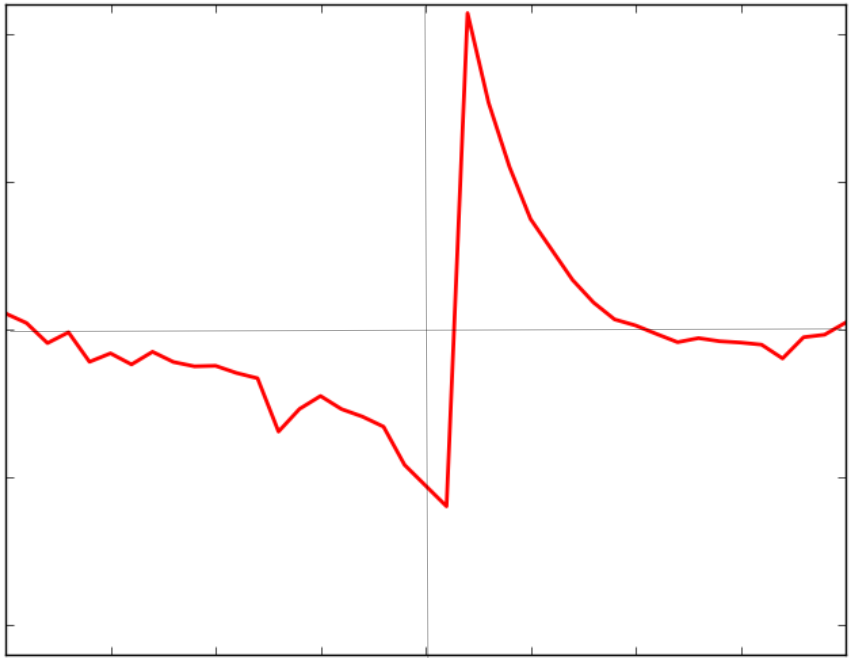
Comparative Behavior: Simulations supports hypothesis

"An Objective Function for STDP", Bengio et al., arXiv 2015

Weight change vs post minus pre spike timing difference



Biological observation
(Bi & Poo 2001)

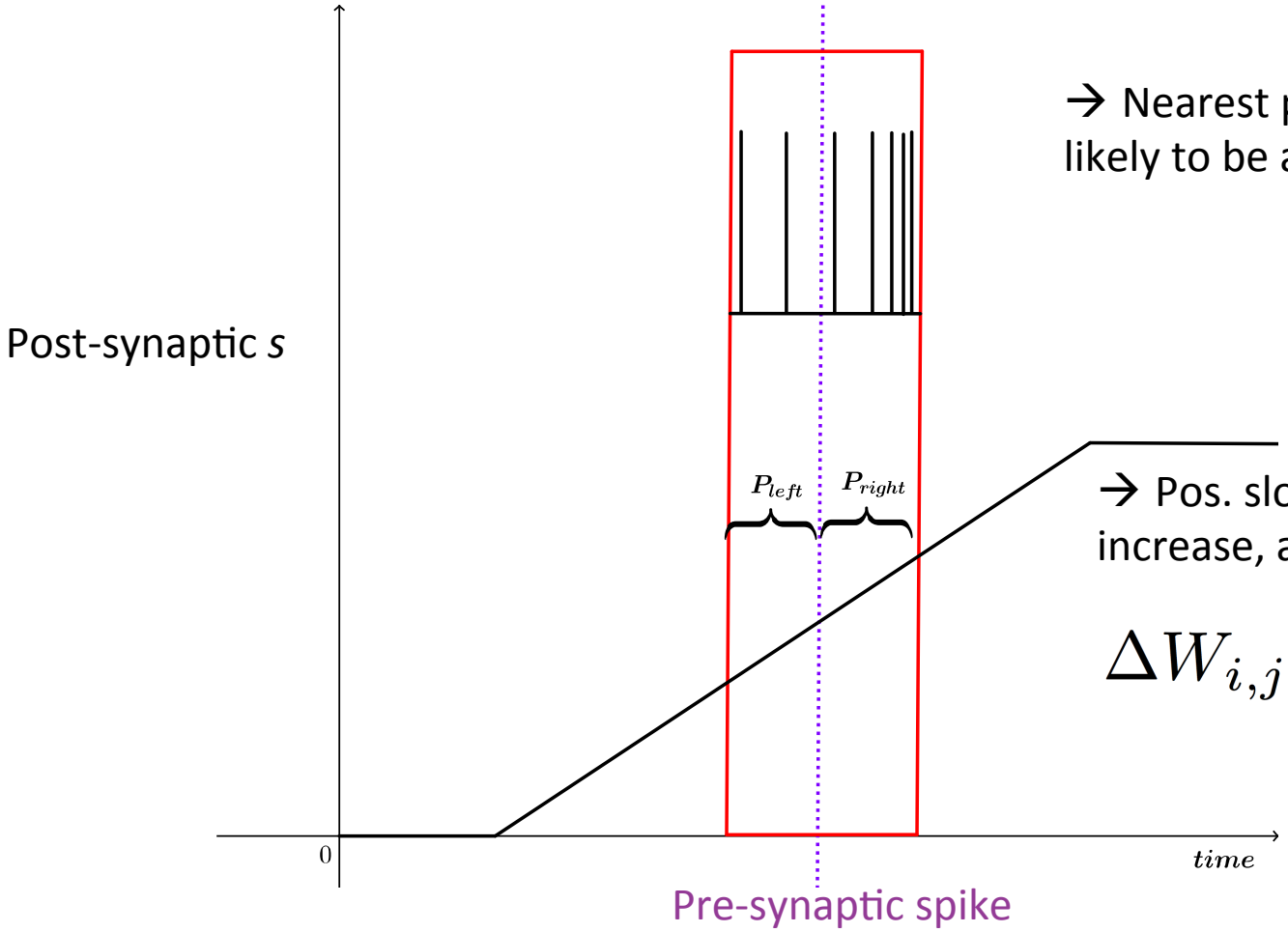


Our simulation, using SGD
on proposed objective fn, i.e.

$$\Delta W_{i,j} \propto \dot{s}_i \rho(s_j)$$

Why it matches the STDP curve

- When post-synaptic *s* increases, probability of post-spike is larger after some event (pre-spike) than before



→ Nearest post-spike more likely to be after pre-spike

→ Pos. slope yields weight increase, and vice-versa

$$\Delta W_{i,j} \propto \dot{s}_i \rho(s_j)$$

Happy Coincidence

This learning rule corresponds to SGD on a local objective function

Does it get us closer to a machine learning interpretation? YES, if...

Proposed update rule corresponds to the SGD update of this ***predictive*** objective function (easy log-lik. interpretation for sequential structure, but what about within-frame dependencies?)

$$J_{\mathrm{STDP}} = \frac{1}{2}\left|\left|f_\theta(\overbrace{s_{t-1}, \eta_{t-1}}^{s_t}) - s_{t+1}\right|\right|^2$$

parametrized state update

prev. state

Injected noise

next state

IF $\dfrac{\partial f_i(s_{t-1}, \eta_{t-1})}{\partial W_{i,j}} \propto \rho(s_{t-1,j})$:

$$\frac{\partial J_{\mathrm{STDP}}}{\partial W_{i,j}} = \frac{\partial J_{\mathrm{STDP}}}{\partial s_{t,i}} \frac{\partial f(s_{t-1}, \eta_{t-1})}{\partial W_{i,j}}$$

$$\propto \frac{\partial J_{\mathrm{STDP}}}{\partial s_{t,i}} \rho(s_{t-1,j})$$

$$\propto (s_{t,i} - s_{t+1,i})\rho(s_{t-1,j})$$

Derived hypothesis #2

How can we satisfy this condition?

Neuron = Leaky integrator

- x = state of visible / clamped units
- h = state of hidden / unclamped units

Like **gradient descent** on squared difference between R and h w/ l.r. ϵ

$$s_{t+1} = f(s_t, \eta_t) = s_t + \epsilon(R(\tilde{s}_t) - s_t)$$

must satisfy

$$\frac{\partial R_i(\tilde{s})}{\partial W_{i,j}} \propto \rho(\tilde{s}_j)$$

weighted sum of input spike rates

corrupted prev. state

value towards which integrator converges

Reformulated hypothesis #2

Happy Coincidence

Denoising auto-encoders with reconstruction function R(s) converge towards R(s)-s = gradient of energy

(Alain & Bengio, ICLR 2013)

Hypothesis #3

Inspired by Hopfield nets and Boltzmann machines

NEURAL COMPUTATION = INFERENCE:

Neural activations tend to noisily move towards configurations making neurons' activations more compatible with each other according to some energy function

Happy Coincidence

Leaky integration + hypothesis #2 + symmetry + noise = Langevin MCMC

Neural Computation as Inference

- **Langevin MCMC (and most MCMC) = small steps going down the energy, plus injecting randomness**

$$z_{t+1} = z_t - \frac{\sigma^2}{2}\frac{\partial E(z_t)}{\partial z_t} + \sigma \text{GaussianNoise}$$

- inference to find good configurations of *h* that explain *x*, given current synaptic weights.

$$\tilde{s}_{t+1} = \tilde{s}_t - \epsilon\frac{\partial E(\tilde{s}_t)}{\partial \tilde{s}_t} + \eta_{t+1} - (1-\epsilon)\eta_t$$

$$\begin{aligned} s_{t+1} &= s_t + \epsilon(R(\tilde{s}_t) - s_t) \\ &= s_t + \epsilon(R(\tilde{s}_t) - \tilde{s}_t + \tilde{s}_t - s_t) \\ &= s_t + \epsilon(-\frac{\partial E(\tilde{s}_t)}{\partial \tilde{s}_t} + \eta_t) \end{aligned}$$

$$R(s) - s \propto \frac{\partial \log P(s)}{\partial s} = -\frac{\partial E(s)}{\partial s}$$

The need for symmetry

- If $R_i \propto \sum_j W_{i,j}\rho(s_j)$
- and $R_i - s_i \propto \frac{\partial E(s)}{\partial s_i}$ or $R_i \propto \frac{\partial L(s)}{\partial s_i}$
- then, by symmetry of second derivatives

$$W_{i,j} \propto \frac{\partial R_i}{\partial s_j} \propto \frac{\partial}{\partial s_j}\frac{\partial L(s)}{\partial s_i} = \frac{\partial^2 L(s)}{\partial s_i \partial s_j} \propto \frac{\partial R_j}{\partial s_i} \propto W_{j,i}$$

- we get **symmetry of the weights**

Hypothesis #4

Inspired by Hopfield nets and Boltzmann machines

There is an inference network made of neuronal unit (one or more neurons) such that the synaptic influence between any pair of such units is symmetric:

$$W_{i,j} \approx W_{j,i}$$

Happy Coincidence

Autoencoders without forced symmetry end up with symmetric weights

(Vincent et al 2011)

WHY? *(Arora et al 2015, arXiv 1511.05653)* $h \approx \mathrm{rect}(W\mathrm{rect}(W^T h))$

Happy Coincidence

There exists an energy function satisfying the previous hypotheses on R: R is the gradient of the energy and it's a weighted sum of presynaptic spikes

A Neural Energy Function

To satisfy conditions $\frac{\partial R_i(\tilde{s})}{\partial W_{i,j}} \propto \rho(\tilde{s}_j)$ & $R(s) = s - \frac{\partial E(s)}{\partial s}$

define

$$E(s) = \sum_i \frac{s_i^2}{2} - \frac{1}{2}\sum_{i \neq j} W_{i,j}\rho(s_i)\rho(s_j) - \sum_i b_i \rho(s_i)$$

Yields

This is new!

Must be **symmetric**

$$R_i(s) = \rho'(s_i)\left(b_i + \sum_j W_{i,j}\rho(s_j)\right)$$

Very Happy Coincidence

Early Inference in Continuous-Variable Energy-Based Models Approximates Back-Propagation

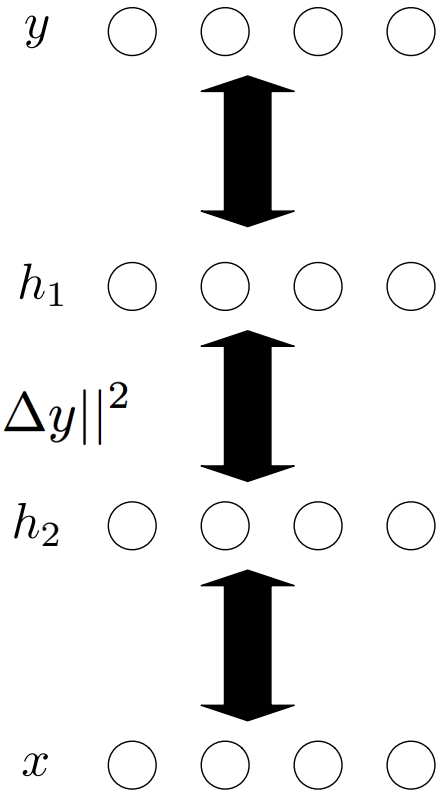
The Connection to Backprop

"Early Inference in Energy-Based Model Approximates Back-Propagation", Bengio, arXiv 2015

$$s = (x, h, y) \quad h = (h_1, h_2)$$

- Near a fixed point of the update $R(s) \approx s$
- Consider what happens when input *x* is clamped and *h* and *y* have settled to $\hat{h}$ and $\hat{y}$ then external signal drives $\hat{y}$ towards a target value *y, creating a perturbation*

$$\Delta y = \epsilon(y - \hat{y}) \quad C = \frac{1}{2}||R_y(\hat{s}) - y||^2 = \frac{1}{2\epsilon^2}||\Delta y||^2$$



Now the closest layer h_1 gets updated

and the perturbation is propagated just

like back-prop would mandate, and similarly this perturbation gets propagated to h_2

Propagation of Perturbations: Lemma

- Consider $L(s) = \frac{1}{2}||s||^2 - E(s)$

  so

$$R_y(\hat{s}) = \frac{\partial L(\hat{s})}{\partial \hat{y}}, R_{h_1}(\hat{s}) = \frac{\partial L(\hat{s})}{\partial \hat{h}_1}$$

- By symetry of second derivatives, we get that

$$\frac{\partial R_{h_1}(\hat{s})}{\partial \hat{y}} = \frac{\partial^2 L}{\partial \hat{y} \partial \hat{h}_1} = \left(\frac{\partial^2 L}{\partial \hat{h}_1 \partial \hat{y}} \right)^T = \frac{\partial R_y(\hat{s})}{\partial \hat{h}_1}^T$$

Propagation of Perturbations: Thm

- At the fixed point (before the perturbation) $\frac{\partial R_y(\hat{s})}{\partial \hat{h}_1} = \frac{\partial \hat{y}}{\partial \hat{h}_1}$
- So the 1st layer perturbation is

$$\Delta h_1 = -\epsilon^2 \frac{\partial R_y(\hat{s})}{\partial \hat{h}_1}^T \frac{\partial C}{\partial \hat{y}} + o(\epsilon^2)$$

$$= -\epsilon^2 \frac{\partial \hat{y}}{\partial \hat{h}_1}^T \frac{\partial C}{\partial \hat{y}} + o(\epsilon^2)$$

$$= -\epsilon^2 \frac{\partial C}{\partial \hat{h}_1} + o(\epsilon^2)$$

- And similarly can show $\Delta h_2 = -\epsilon^3 \frac{\partial C}{\partial \hat{h}_2} + o(\epsilon^3)$

Resulting Weight Update = SGD on Prediction Error on Visible Units

- If $\Delta W_{i,j} \propto \dot{s}_i \rho(s_j)$ (SGD on STDP objective function)
- and $\dot{s}_i \propto \frac{\partial C}{\partial s_i}$, $\frac{\partial R_i(s)}{\partial W_{i,j}} \propto \rho(s_j)$
- then

$$\Delta W_{i,j} \propto \frac{\partial C}{\partial W_{i,j}}$$

- But that is only for the feedforward weights!

Many Open Questions Remain

- Trying to bridge the gap between neuroscience and deep learning has seemingly helped us bridge the gap between Boltzmann machines and backprop
- Many exciting & happy coincidences … and many questions!

- What about the other contributions to the weight update?
- How about when we are not at a fixed point?
- How to handle the unsupervised case?
- What if we do not have a true energy function (only approximate symmetry)?

MILA: Montreal Institute for Learning Algorithms