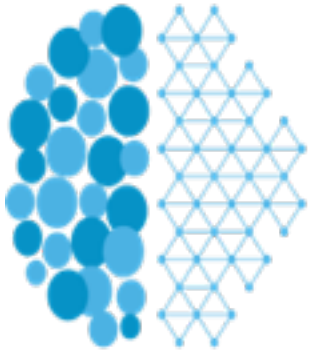
Towards biologically plausible deep learning

Yoshua Bengio

NIPS'2016 Workshops

Brains and Bits: Neuroscience Meets Machine Learning



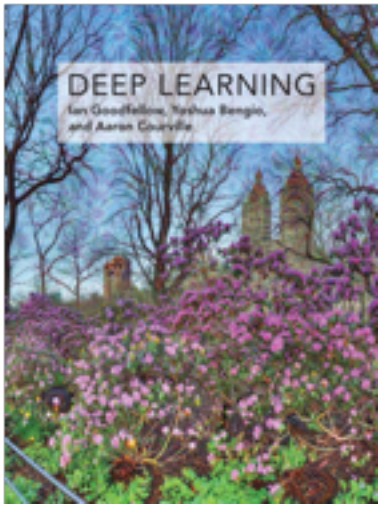
MILA

CIFAR
CANADIAN INSTITUTE FOR ADVANCED RESEARCH

Université de Montréal

IVADO

PLUG: **Deep Learning**, MIT Press book is out, chapters will remain online



Deep Learning Jobs in Montreal

- Faculty positions at all levels at U. Montreal
- Researcher positions at Element AI and Google Brain Montreal
- Researcher positions at U. Montreal (IVADO data science center)
- Studentships at all levels at U. Montreal

Something BIG is happening in Montreal

COME AND SEE OUR NEW DIGS AT ivado.ca

IVADO

HEC Montréal
Polytechnique Montréal
Université de Montréal

Central Issue in Deep Learning: Credit Assignment → Necessary to Jointly Coordinate Learning in a Large Network

- i.e., what should hidden layers do to be useful to other hidden layers and larger objectives of the network?

- Established approaches:
  - Backpropagation
  - Stochastic relaxation in Boltzmann machines
  - ~~REINFORCE~~

Variance scales linearly with number of neurons getting the credit, so REINFORCE, alone, cannot cut it

- Are these related?
- *How does the brain do it?*

Biologically Plausible Backprop

- Train an auto-encoder without backprop, with difference target propagation
  - *Lee et al, 2014, ECML 2015, arXiv:1412.7525*
- Showed that a rate-based update emulates STDP
  - *Bengio et al, 2015, arXiv:1509.05936*
- Showed that propagation of perturbations at fixed-point of a symmetrically connected recurrent net propagates gradients
  - *Bengio & Fischer, 2015, arXiv:1510.02777*
- Showed that the rate-based STDP update after propagation of perturbations corresponds to SGD on prediction error and introduced novel ML framework for ”fixed-point propagation” or ”equilibrium propagation”
  - *Scellier & Bengio, 2016, 1602.05179*
- New theory for gradient estimation in recurrently connected nets, showed fixed-point recurrent net can be trained on MNIST to 0% training error
  - *Scellier & Bengio, 2016, 1602.05179*
- Local reconstruction cost yields fast feedforward inference
  - *Bengio, Scellier, Bilaniuk, Sacramento & Senn, arXiv:1606.01651*

Variant of the energy function of the continuous Hopfield Net

- Energy (or Lyapunov) function

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

- Has derivative $\frac{\partial E(s)}{\partial s} = s - R(s)$

Different: s returns to 0 when s goes outside of the (0,1) interval

- Where $R(s) = \rho'(s) \circ (b + W\rho(s))$
- So $\dot{s} = \epsilon(R(s) - s) = -\epsilon\frac{\partial E}{\partial s}$ is gradient descent on the energy

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 move towards good configurations of *h* that explain *x*, given current synaptic weights.

The need for symmetry

- If we want $R_i \propto \sum_j W_{i,j}\rho(s_j)$
- and $R_i - s_i \propto \frac{\partial E(s)}{\partial s_i}$
- then, we need symmetry because otherwise we have

$$R_i(s) = \rho'(s_i)\left(\sum_{j\neq i}\frac{1}{2}(W_{i,j}+W_{j,i})\rho(s_j) + b_i\right)$$

Happy Coincidence

Autoencoders without forced symmetry end up with symmetric weights

Experimentally found*: (Vincent et al 2011)*

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

Exact Symmetry is Not Needed for Backprop to Work

Feedback Alignment: *(Lillicrap et al 2014)*

But it would be good if the learning algorithms tended to something equivalent.

How to perform fast inference in negative phase (sample from posterior)

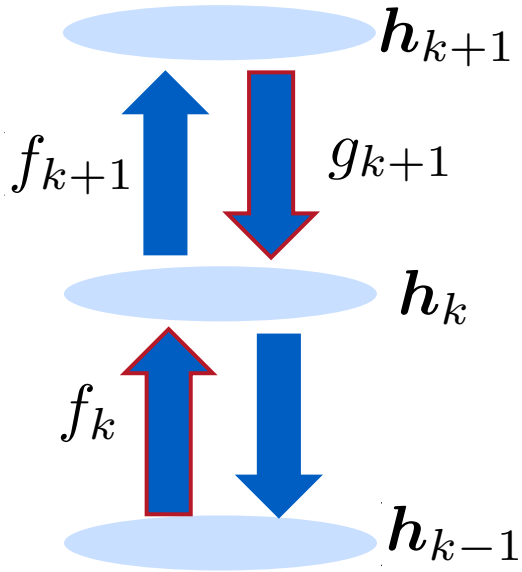
Bengio, Scellier, Bilaniuk, Sacramento & Senn, arXiv:1606.01651

Feedforward Initialization for Fast Inference of Deep Generative Networks is biologically plausible

- Sufficient conditions for feedforward computation to correspond to fixed point of recurrent relaxation:
  - Each pair of successive layers forms a good auto-encoder

$$\boldsymbol{h}_k = f_k(\boldsymbol{h}_{k-1}) = g_{k+1}(\boldsymbol{h}_{k+1})$$

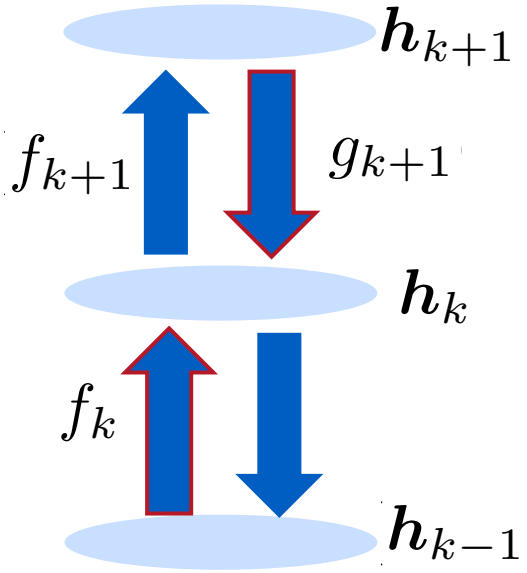
$$\boldsymbol{h}_k = f_k(\boldsymbol{h}_{k-1}) = g_{k+1}(f_{k+1}(\boldsymbol{h}_k))$$

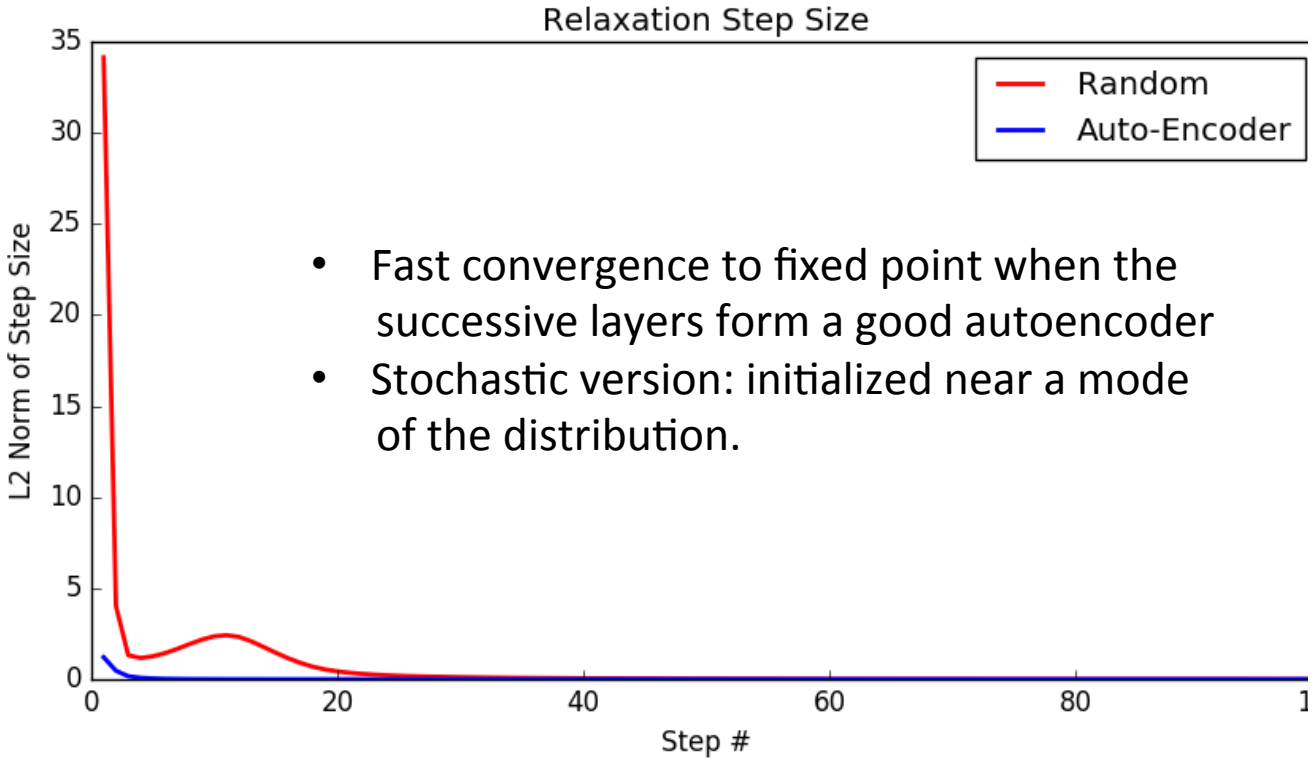


Feedforward Initialization for Fast Inference of Deep Generative Networks is biologically plausible

Bengio, Scellier, Bilaniuk, Sacramento & Senn, arXiv:1606.01651

- Bottom-up input = basal dendrite; top-down input = apical dendrite
- Mutual prediction criterion = auto-encoder reconstruction criterion





- Fast convergence to fixed point when the successive layers form a good autoencoder
- Stochastic version: initialized near a mode of the distribution.

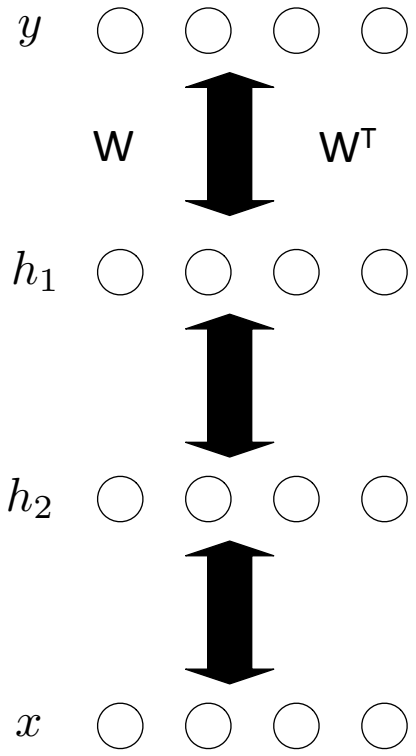
Propagation of errors = propagation of surprises = getting back in harmony

Bengio & Fischer, 2015, arXiv:1510.02777

Variation on the output *y* is propagated into a variation in h_1 mediated by the feedback weights W^T = transpose of feedforward weights W

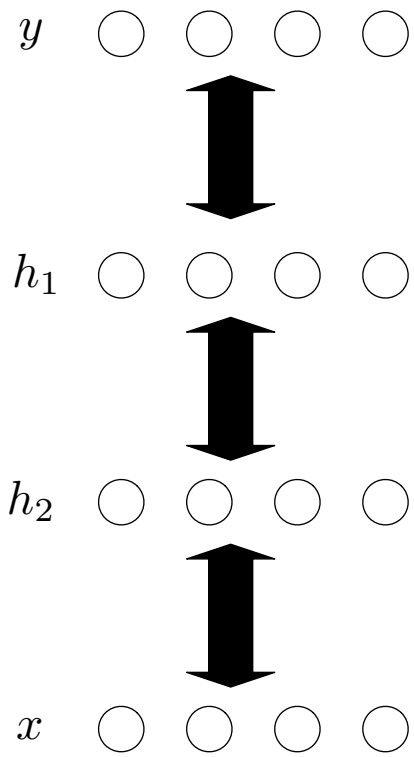
Then the variation in h_1 is transformed into a variation in h_2, etc.

And we show that $\dot{h}$ proportional to $-\frac{\partial C}{\partial h}$



Propagation of errors = Incremental Target Prop

- If temporal derivatives = error gradients
- Feedback paths compute "incremental targets" for the feedforward paths, moving the hidden activations in the right direction
- The top-down perturbations which are propagated represent the "surprise" signal while the feedback paths compute targets towards which the feedforward activations are moved



- **No structural difference between forward-prop and back-propagations (except for clamping of visible)**

Now mostly material from:

Equilibrium Propagation

Bridging the Gap Between Energy-Based Models and Backpropagation

arXiv:1602.0519

Benjamin Scellier & Yoshua Bengio

Montreal Institute for Learning Algorithms

How could we train a physical system that performs computations?

- Consider a physical system that performs potentially useful computations through its deterministic or stochastic dynamics
- It has parameters θ that could be tuned
- Tractable cost function C can measure how good are its answers
- The relationship between parameters and objective J (cost at equilibrium of the dynamics) is implicit (via the dynamics)
- How to estimate the gradient of the loss wrt parameters?

Equilibria of the Dynamics

- Deterministic case: dynamics converge to fixed points which are minima of an **GENERALLY UNKNOWN** energy function *F*

$$\frac{\partial F}{\partial s} = 0 \leftrightarrow \dot{s} = 0$$

- Stochastic case: dynamics converge in probability to the Boltzmann distribution associated with *F*

$$s \sim P(s) \propto e^{-F(s)}$$

Two Phases: Previous Work

1. **Almeida-Pineda** consider the same objective function as ours but propose another algorithm to compute the gradient, **Recurrent Backpropagation**, which requires a *different dynamics in the second phase.*
2. **Contrastive Hebbian Learning** (CHL) has *theoretical issues*: the update may be inconsistent if the two phases land in different modes of the energy function.
3. **Boltzmann Machine Learning** requires two independent phases, making an analogy with backpropagation less obvious.
4. **Contrastive Divergence** (CD) has *theoretical issues* too: it does not optimize any objective function.
5. **Xie-Seung** show the equivalence between CHL and backprop but require *weak feedback weights and different learning rates*.

Equilibrium Propagation solves all these issues at once, at least in theory, if not in practice.

Influence Parameter

CHL and Boltzmann Machine Learning have **two modes**:
- one mode with **clamped** outputs
- one mode with **free** outputs.

Here we introduce an **influence parameter** β which controls the level of influence of the external world on the input and output units.

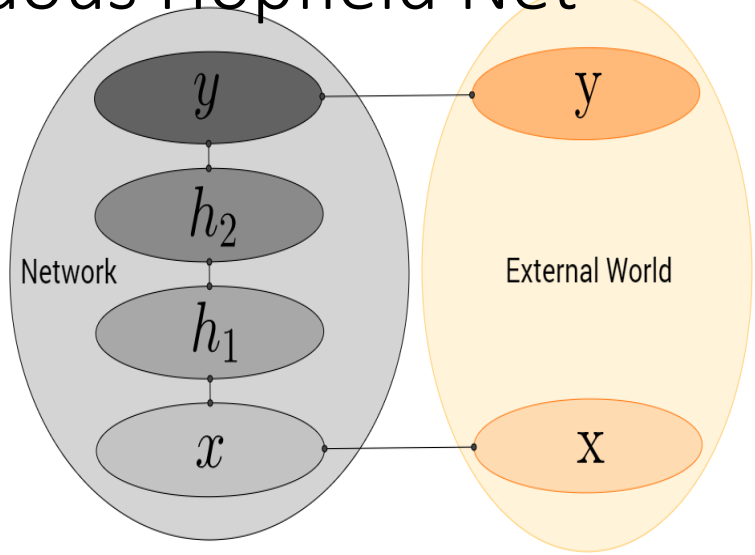
Example: Supervised Continuous Hopfield Net

State: $s = \{x, y, h\}$

External World: $\mathrm{v} = \{\mathrm{x}, \mathrm{y}\}$

Learned Parameter: θ

Influence Parameter: $\beta = \{\beta_x, \beta_y\}$



Example: Supervised Continuous Hopfield Net

- Total Energy:

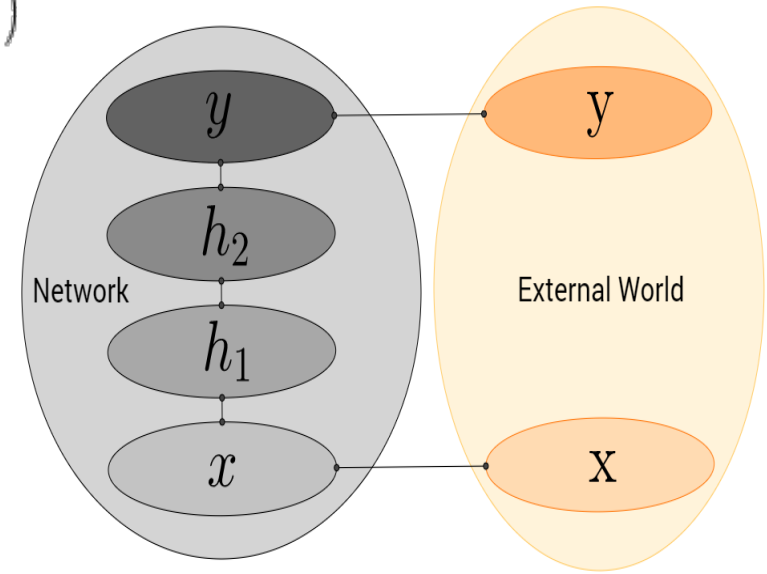
$$F(\theta, \beta, s, \mathrm{v}) = E(\theta, s) + A(\beta, s, \mathrm{v})$$

- Internal Potential Energy

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

- External Potential Energy

$$A(\beta, s, \mathrm{v}) = \frac{1}{2}\beta_x ||x - \mathrm{x}||^2 + \frac{1}{2}\beta_y ||y - \mathrm{y}||^2$$

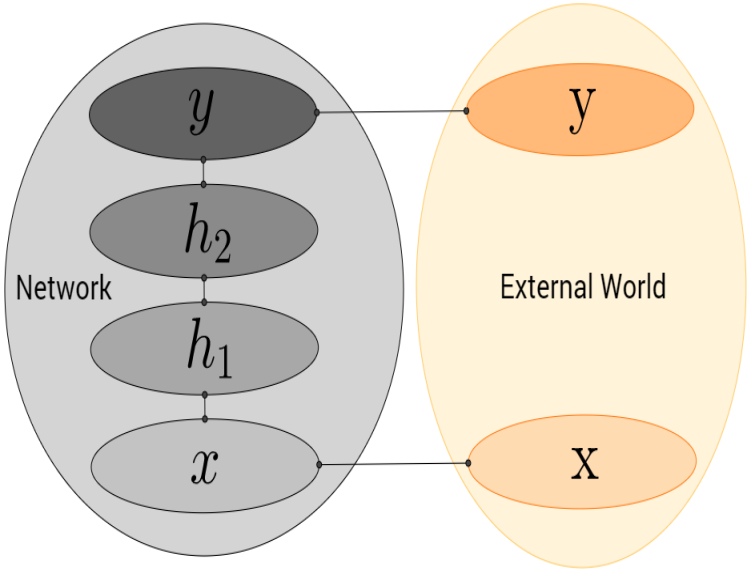


Solving issues of Contrastive Hebbian Learning

Contrastive Hebbian Learning Rule:

$$\Delta\theta \propto -\left(\frac{\partial E}{\partial \theta}(\theta, s^{\infty}) - \frac{\partial E}{\partial \theta}(\theta, s^{0})\right)$$

s^0 : fixed point with **free outputs**,
s^∞ : fixed point with **fully clamped outputs**.



Theoretical problem of CHL: no meaningful objective (difference in energies could be <0)

Equilibrium Propagation Update Rule:

$$\Delta\theta \propto -\lim_{\xi \to 0} \frac{1}{\xi}\left(\frac{\partial E}{\partial \theta}(\theta, s^{\xi}) - \frac{\partial E}{\partial \theta}(\theta, s^{0})\right)$$

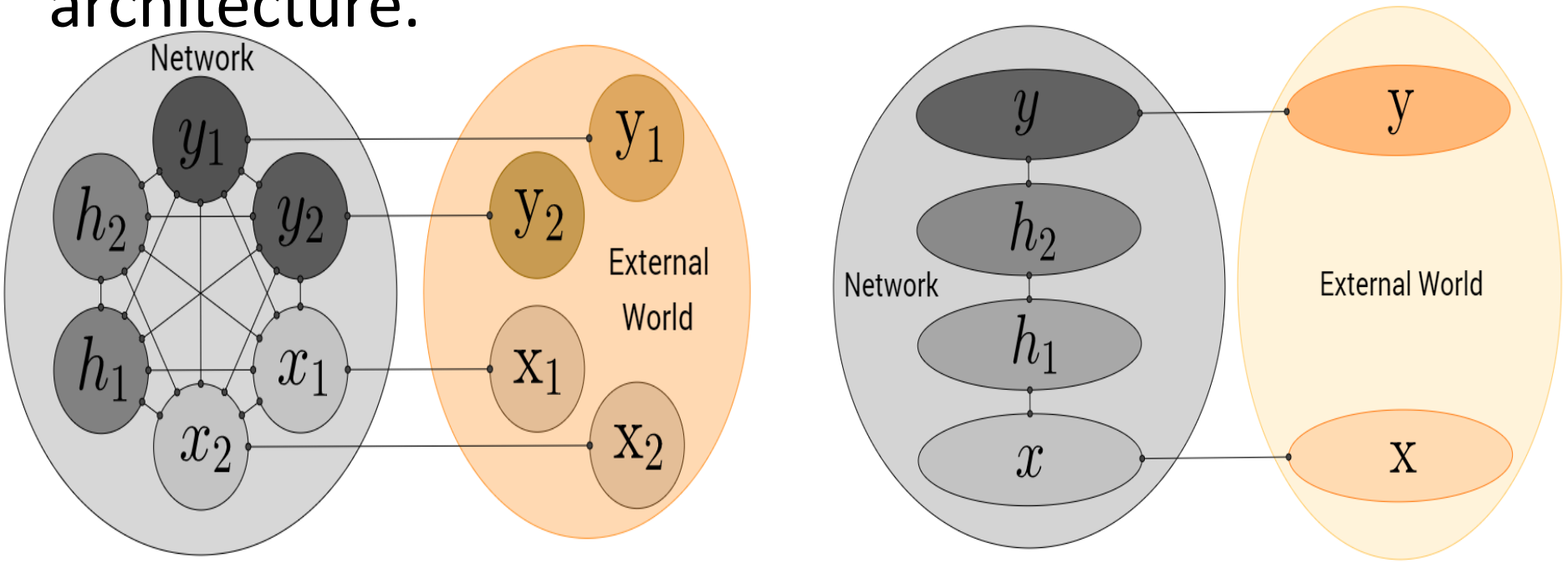
s^0 : fixed point with **free outputs**,
s^ξ : fixed point with **weakly clamped outputs**.

The second phase corresponds to nudging the fixed point s^0 towards the fixed point s^ξ, which has lower cost value.

$$C := \frac{1}{2}||y - \mathrm{y}||^2$$

More General Setting

Equilibrium-Prop works for any architecture, even a fully connected network, or one with lateral connections. The connection with Backprop is more obvious when the network has a layered architecture.



Main Theorem

- Gradient on the objective function (cost at equilibrium) can be estimaed by a ONE-DIMENSIONAL finite-difference

$$\frac{d}{d\theta}J_\beta^\delta(\theta,\mathrm{v}) = \lim_{\xi\to 0}\frac{1}{\xi}\left(\frac{\partial F}{\partial\theta}\left(\theta,\beta+\xi\delta,s_{\theta,\mathrm{v}}^{\beta+\xi\delta},\mathrm{v}\right) - \frac{\partial F}{\partial\theta}\left(\theta,\beta,s_{\theta,\mathrm{v}}^{\beta},\mathrm{v}\right)\right)$$

Small nudging

Sufficient statistic after nudging

Sufficient statistic before nudging

Stochastic version:

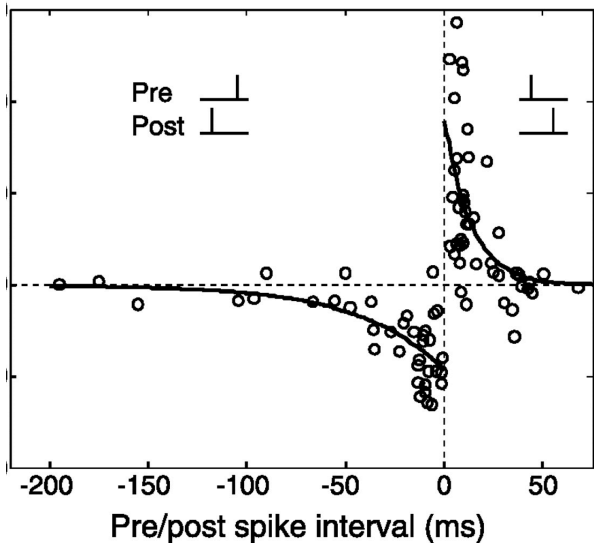
$$\frac{d}{d\theta}\widetilde{J}_\beta^\delta(\theta,\mathrm{v}) = \lim_{\xi\to 0}\frac{1}{\xi}\left(\mathbb{E}_{\theta,\mathrm{v}}^{\beta+\xi\delta}\left[\frac{\partial F}{\partial\theta}(\theta,\beta+\xi\delta,s,\mathrm{v})\right] - \mathbb{E}_{\theta,\mathrm{v}}^{\beta}\left[\frac{\partial F}{\partial\theta}(\theta,\beta,s,\mathrm{v})\right]\right)$$

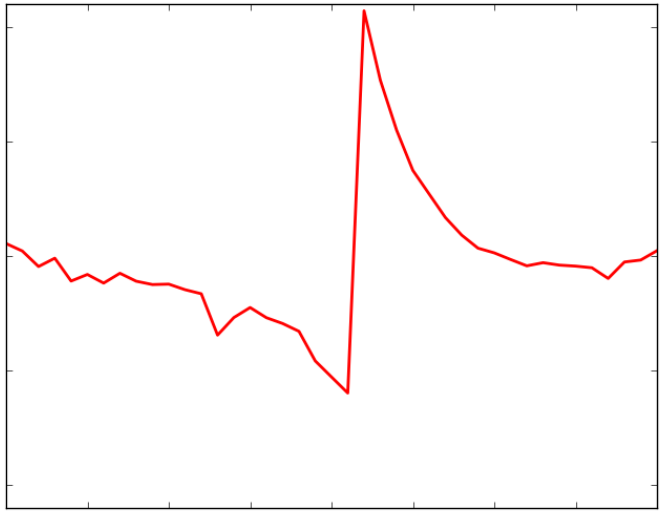
The STDP Connection

- Inspiration from Hinton 2007 (talk at Deep Learning Workshop @ NISP); see also April 2016 talk by Hinton @ Stanford, "Can the brain do back-propagation?"
- *Bengio et al 2015 "STDP as presynaptic activity times rate of change of postsynaptic activity" arXiv:1509.05936*
  - shows that weight updates

$$\Delta W_{i,j} \propto \frac{d\rho(s_i)}{dt}\rho(s_j)$$

  - replicates the STDP experimental signature. If symmetry is added we get the same weight update as Eq.Prop.





Equilibrium Propagation Yields STDP - A Differential Contrastive Hebbian Update

With energy function

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

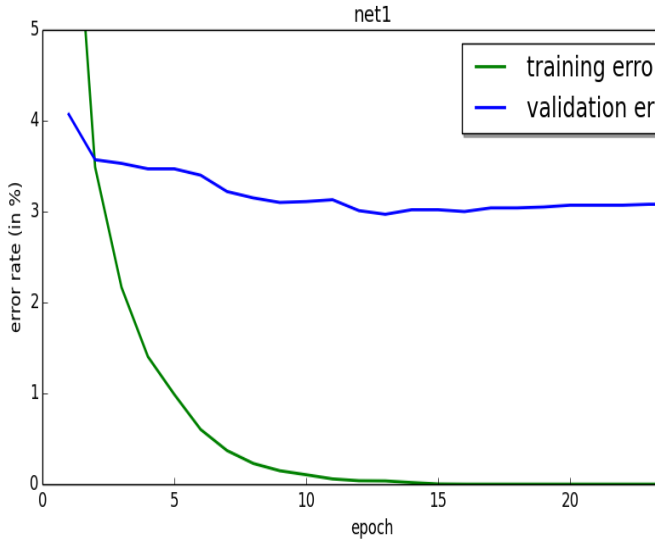
The SGD update is

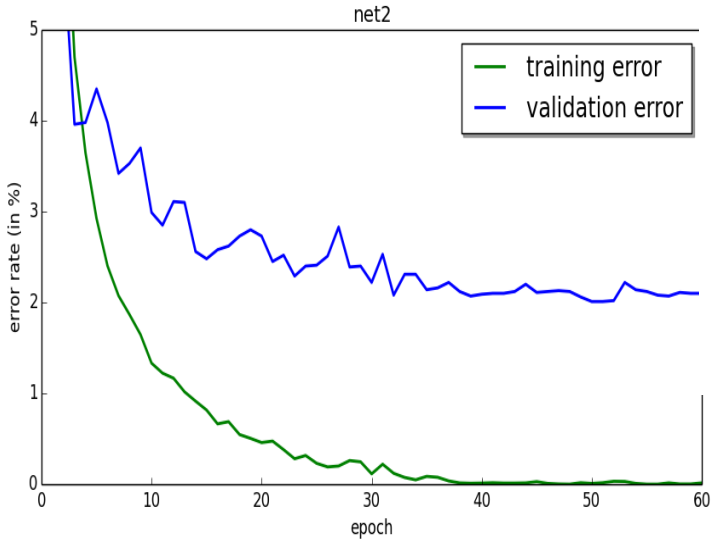
$$\Delta W_{ij} \propto \frac{d}{dt} \rho(s_i) \rho(s_j)$$

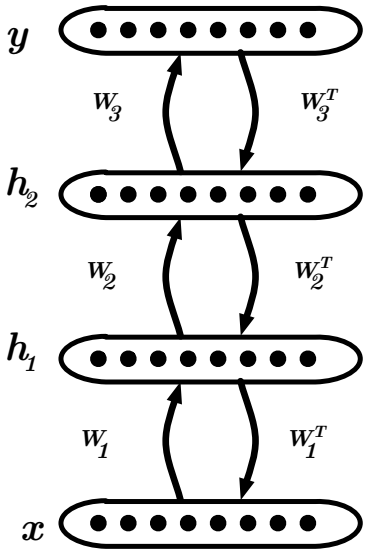
while in the positive phase (and no change in the neg. phase)

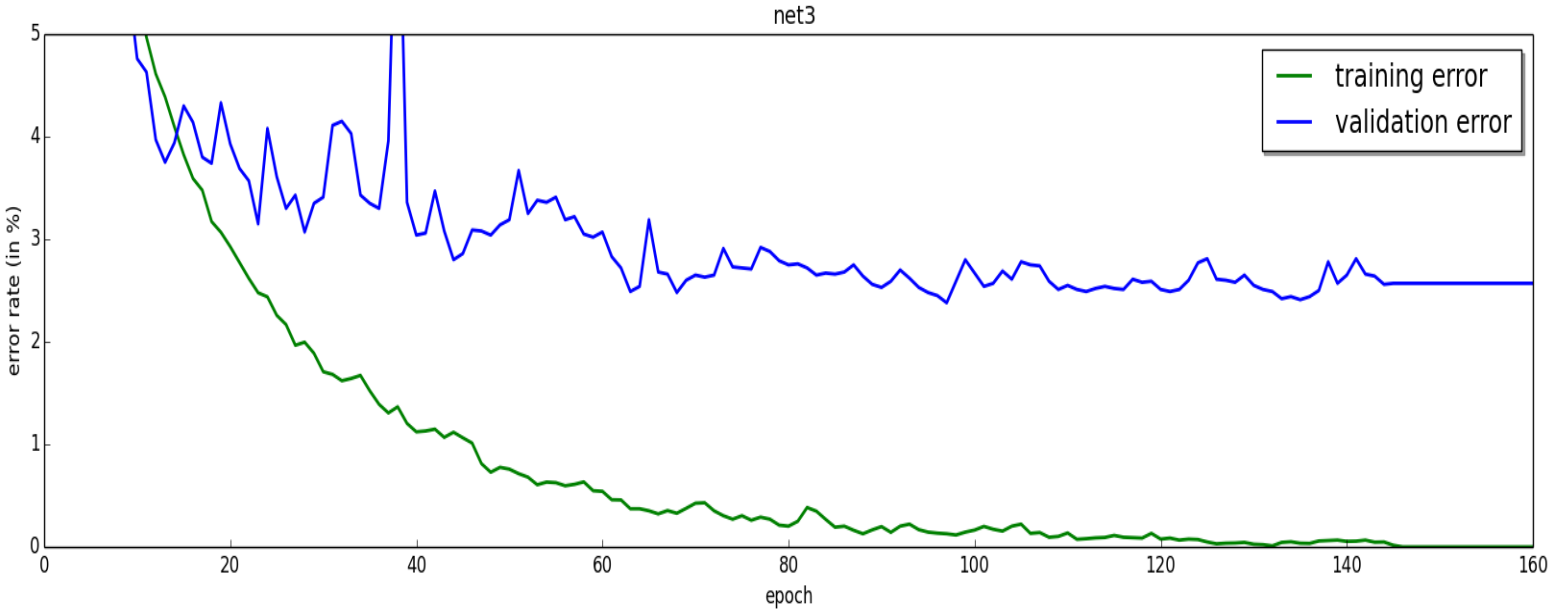
Note the symmetry constraint. As shown in Bengio et al 2015, this matches the ordinary STDP profile of Bi & Poo 2001.

Experimental Results



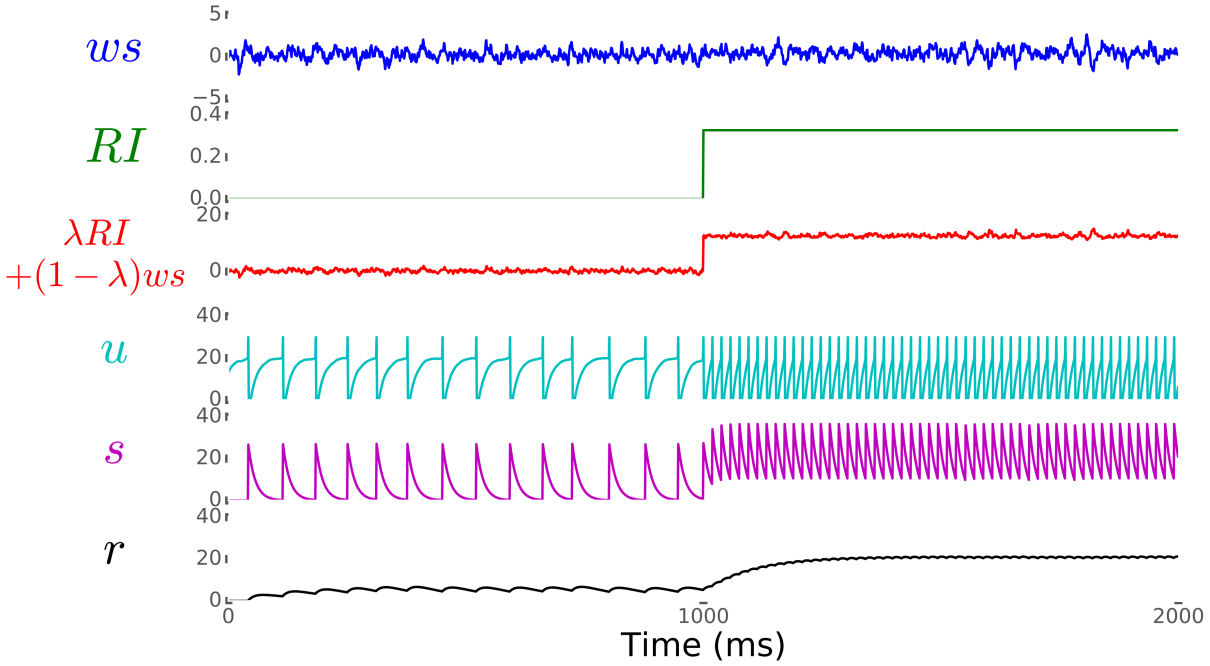


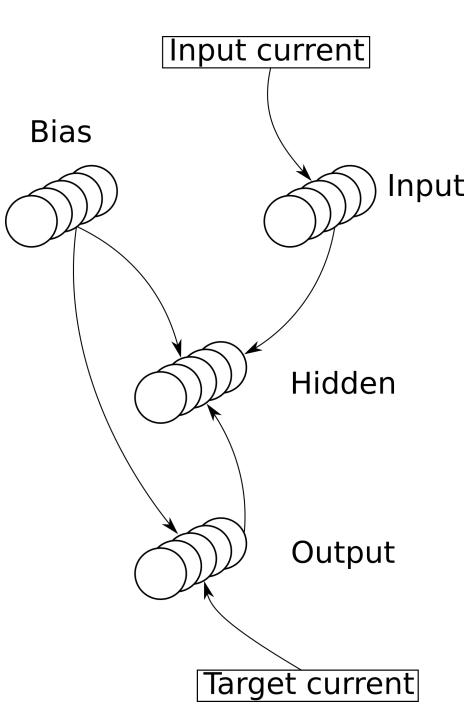




Results with Spikes (Mesnard, Gerstner, Brea 2016)

- '*Towards deep learning with spiking neurons in energy based models with contrastive Hebbian plasticity*', presented this morning in the NIPS 2016 workshop on 'Computing with spikes'





Inherits Properties of Backprop

- Unlike finite-difference methods in parameter space, backprop is equivalent to finite difference IN A SINGLE DIRECTION, THE DIRECTION OF THE COST GRADIENT. Same here.

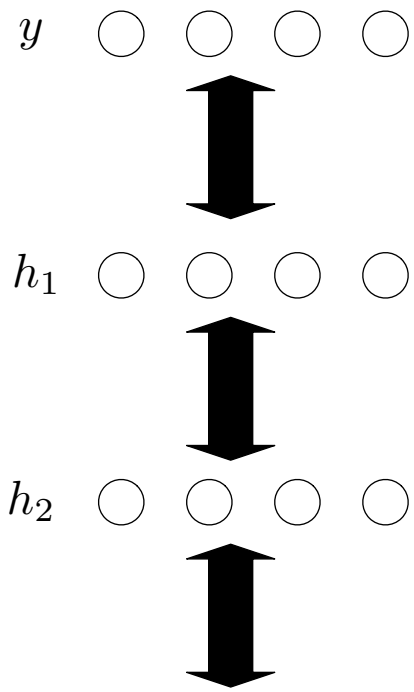
- In the case where the network has a multi-layer structure, we can show that the propagation of perturbations (nudges) corresponds to back-propagation of gradients
  - First shot at showing this in
    - *Bengio & Fischer, Early Inference in Energy-Based Models Approximates Back-Propagation, arXiv:1510.02777*

Propagation of errors = Incremental Target Prop

(see Hinton's talk at Stanford, 27 April 2016, Can the brain do back-propagation)

- When nudging (perturbation) is propagated, temporal derivatives = error gradients wrt hidden activations of neg. phase

- Feedback paths compute "incremental targets" for the feedforward paths, moving the hidden activations in the right direction
- The top-down perturbations which are propagated represent the "surprise" signal while the feedback paths compute targets towards which the feedforward activations are moved



- **No structural difference between forward-prop and back-propagations (except for nudging of outputs)**

Equilibrium Propagation Includes Ordinary Backprop for Feedforward Nets as Special Case

- Consider the internal energy function

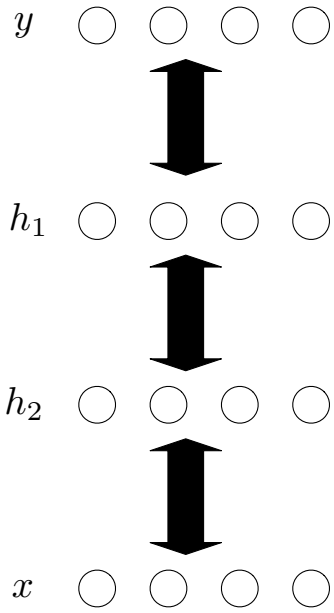
$$E = \sum_l ||h_l - f_l(h_{l-1})||^2$$

With layered architecture, h_l = *l*-th layer of activations, $h_0 = x$

f_l = parametrized computation at *l*-th layer.

- E has a global minimum at $h_l = f_l(h_{l-1})$
- It is also a mode associated with stationary distribution.

Equilibrium Propagation Includes Ordinary Backprop for Feedforward Nets as Special Case



- With this feedforward-compatible energy-function

$$E = \sum_l ||h_l - f_l(h_{l-1})||^2$$

- Negative phase is EQUIVALENT to feedforward prop.
- Positive phase: nudge outputs, nudges propagated backwards
- Equilibrium-propagation estimates the same gradient as backprop in a feedforward net, but using a physical (analog) dynamical system which implements the above energy function, with no need for a separate circuit for backpropagation.

Open Problems

- Get rid of local minima of energy formulation and generalize to system defined by its dynamics, learn the transition operator, **thus avoiding the weight symmetry constraint**

- Generalize these ideas to unsupervised learning (ongoing)

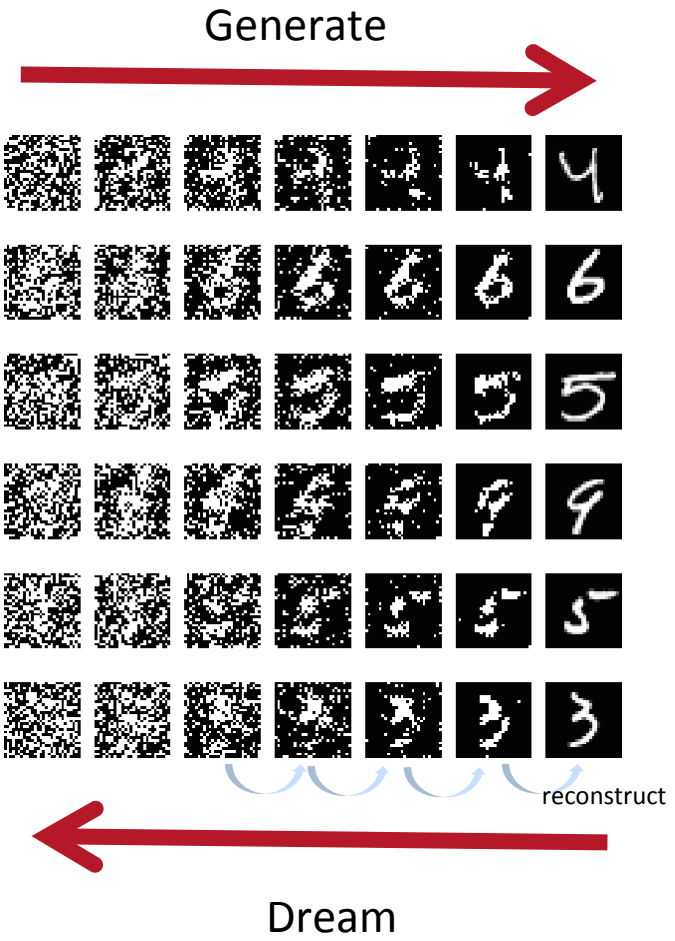
- What about backprop through time?

STDP vs reverse-STDP: Dreams?

- Equilibrium-propagation gives rise to STDP-like updates, where a future state is considered "better" than the previous state, closer to the observed data.
- This works because we start from action/prediction and then get a feedback from the outside world = target.
- This is not so meaningful if the output variable is multimodal.
- Then it seems to make more sense to start from the data and move towards where the model wants to go, like in CD and minimizing reconstruction error in genral.
- However, this gives rise to reverse-STDP (the past is the target), i.e., STDP with opposite sign. Makes sense for DREAMS?

Variational Walkback
Goyal, Ke, Lamb, Bengio, submitted to ICLR 2017

- Sample a data point (dream of the seed)
- Start running the Markov Chain of the brain's transition operator
- Gradually increase temperature (more noise)
- At each step, update parameters to make previous state more likely than next state (a kind of reconstruction error)
- This makes the model FORGET the states it visits in this noisy dream-like simulation



Brain Implementations of GANs and Actor-Critic: Questioning the Single Objective Optimization Dogma

- A GAN-like discriminative objective or the critic in an actor-critic setup could be used to train a predictor or actor resp., using Equilibrium-propagation.
- Issues:
  - The weight updates in the actor/predictor are controlled separately from the updates in the discriminator/critic.
  - A very deep actor/predictor and discriminator/critic raises the question of plausibility of the timing constraint (time to go back-and-forth several times across a very deep net?)

MILA

Université de Montréal