Decelle, Furtlehner, Seoane (NeurIPS 2021)

Agoritsas, Catania, Decelle, Seoane (ICML 2023)

On the effect of MCMC in the training of EBMs

Beatriz Seoane

LISN Paris-Saclay University

EBMs... Yes! ... But!

EBMs

Pros: Appealing for modeling and interpretability applications

Cons: **Very hard to train** :

- The quality of the training is hard to control
- Sampling are unstable

Generating samples : what one expects

Empirical *Model*

$$p_{\mathrm{data}}(x) \sim \frac{e^{-E_{\boldsymbol{\theta}}(\boldsymbol{x})}}{Z_{\boldsymbol{\theta}}}$$

Dominated minimum free-energy configurations

$$\{\boldsymbol{x}\}_{\mathrm{eq},\boldsymbol{\theta}} \sim \mathcal{D}$$

$\Longrightarrow$

Markov Chain Monte Carlo
Langevin dynamics

Generate new samples

$$\left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\mathrm{data}}} = \left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\boldsymbol{\theta}}} \quad \forall \theta_i$$

Generating samples : what one expects

MCMC sampling steps

(some time to equilibrate/converge)

10^4 steps

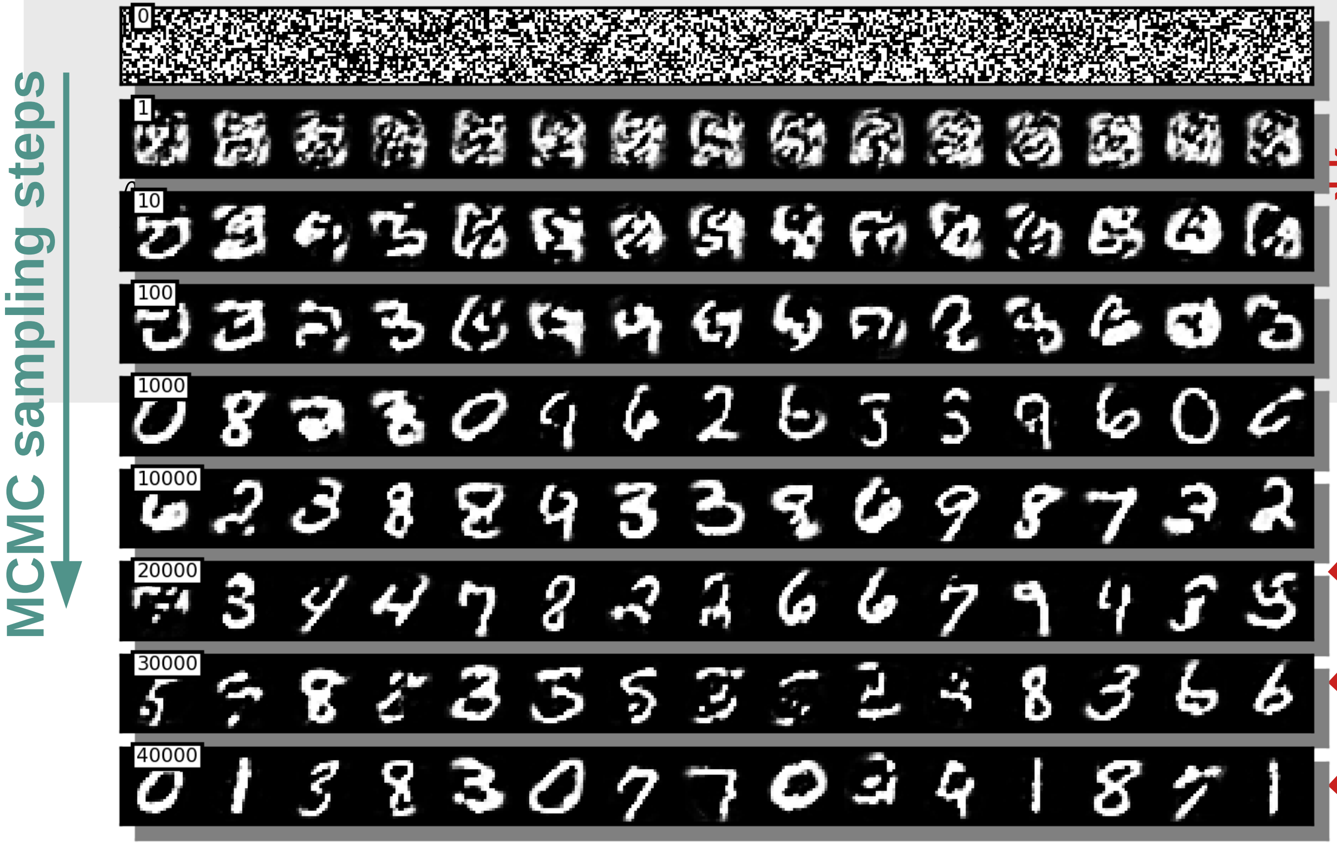
Trained model $E_{\boldsymbol{\theta}}(\boldsymbol{x})$

$\Downarrow$

Generate samples with $p_{\boldsymbol{\theta}}(\boldsymbol{x})$

$$\left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\text{data}}} = \left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\boldsymbol{\theta}}} \quad \forall \theta_i$$

Generating samples : what one expects...



Markov Chain Monte Carlo
Langevin dynamics

⇒ Generate new samples

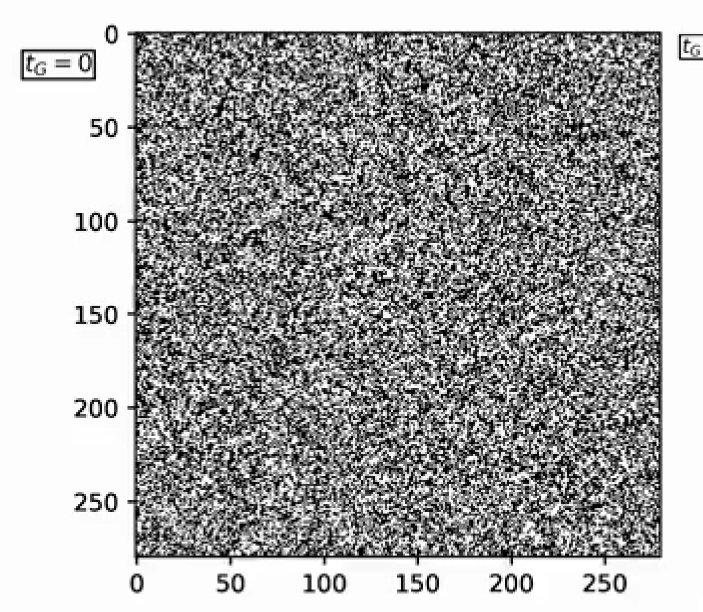
$$\left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\mathrm{data}}} = \left\langle \frac{\partial E}{\partial \theta_i} \right\rangle_{p_{\boldsymbol{\theta}}} \quad \forall \theta_i$$

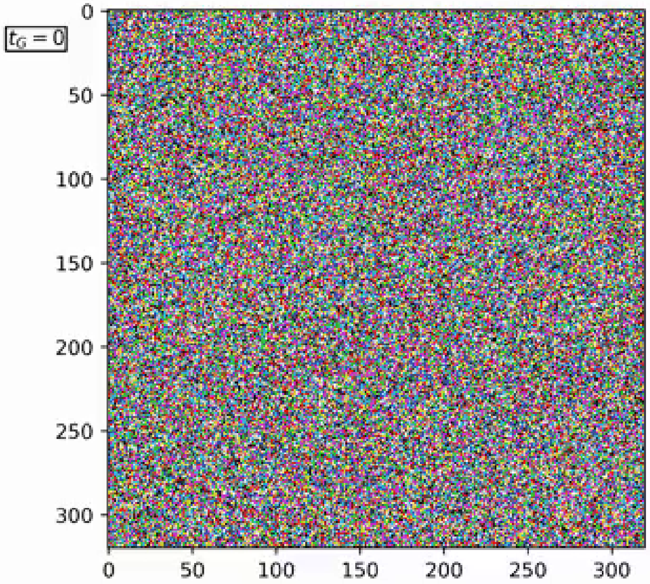
+10^4 steps

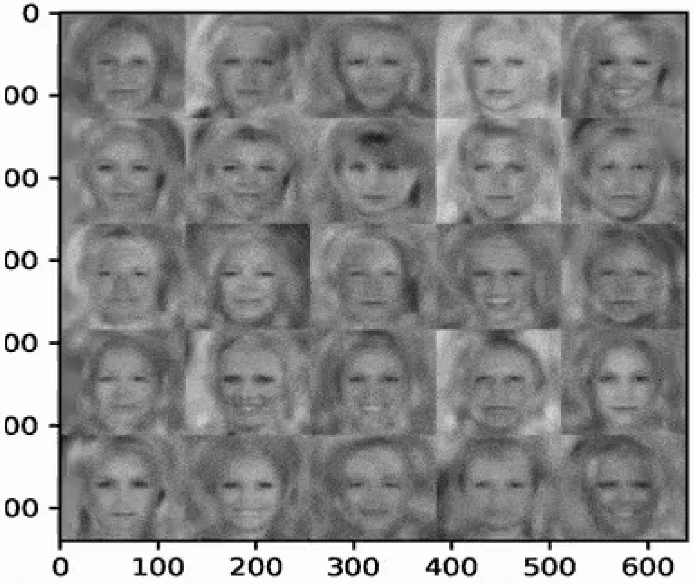
+10^4 steps

+10^4 steps

But this is not what one typically observes...







What’s going wrong?

The sampling problem

$$\boldsymbol{\nabla}\mathcal{L} = \boxed{\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\mathrm{data}}}} - \boxed{\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}}}$$

Easy

Hard $\Rightarrow$ Markov Chain MC sampling

The sampling problem

$$\nabla \mathcal{L} = \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\mathrm{data}}} - \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}}$$

Easy

Hard $\Rightarrow$ Markov

RBM

$$\frac{\partial \mathcal{L}}{\partial W_{ia}} = \langle x_i h_a \rangle_{p_{\mathcal{D}}} - \langle x_i h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \eta_a} = \langle h_a \rangle_{p_{\mathcal{D}}} - \langle h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \zeta_i} = \langle x_i \rangle_{p_{\mathcal{D}}} - \langle x_i \rangle_{\mathcal{E}}$$

The sampling problem

$$\nabla \mathcal{L} = \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\text{data}}} - \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}}$$

Easy

Hard ⇒ Markov

RBM

$$\frac{\partial \mathcal{L}}{\partial W_{ia}} = \langle x_i h_a \rangle_{p_\mathcal{D}} - \langle x_i h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \eta_a} = \langle h_a \rangle_{p_\mathcal{D}} - \langle h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \zeta_i} = \langle x_i \rangle_{p_\mathcal{D}} - \langle x_i \rangle_{\mathcal{E}}$$

$$\boldsymbol{x}_{\text{gen}}^{(m)} \qquad m = 1, \ldots, n_{\text{chains}}$$

$\boldsymbol{X}_{\text{gen}} \sim P_{\boldsymbol{\theta}}$ Via a Markov Chain Monte Carlo process

$$\langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} \approx \frac{1}{n_{\text{chains}}} \sum_{m=1}^{n_{\text{chains}}} \nabla E(\boldsymbol{x}_{\text{gen}}^{(m)})$$

The sampling problem

$$\nabla \mathcal{L} = \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\text{data}}} - \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}}$$

Easy

Hard $\Rightarrow$ Markov

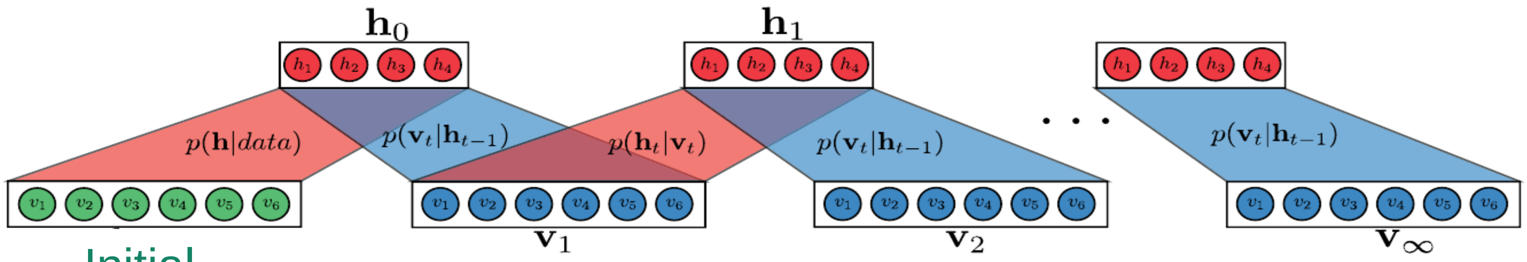
RBM

$$\frac{\partial \mathcal{L}}{\partial W_{ia}} = \langle x_i h_a \rangle_{p_{\mathcal{D}}} - \langle x_i h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \eta_a} = \langle h_a \rangle_{p_{\mathcal{D}}} - \langle h_a \rangle_{\mathcal{E}}$$

$$\frac{\partial \mathcal{L}}{\partial \zeta_i} = \langle x_i \rangle_{p_{\mathcal{D}}} - \langle x_i \rangle_{\mathcal{E}}$$

Alternating/Block Gibbs sampling



Initial configuration

Glauber dynamics

$$p(h_\nu = 1|\boldsymbol{v}) = \sum_{\boldsymbol{h}_{-\nu}} \frac{p(\boldsymbol{h}, \boldsymbol{v})}{p(\boldsymbol{v})} = \text{sigmoid}\left(b_\nu + \sum_i v_i w_{i\nu}\right)$$

$$p(v_i = 1|\boldsymbol{h}) = \sum_{\boldsymbol{v}_{-i}} \frac{p(\boldsymbol{h}, \boldsymbol{v})}{p(\boldsymbol{h})} = \text{sigmoid}\left(a_i + \sum_\mu w_{i\mu} h_\mu\right)$$

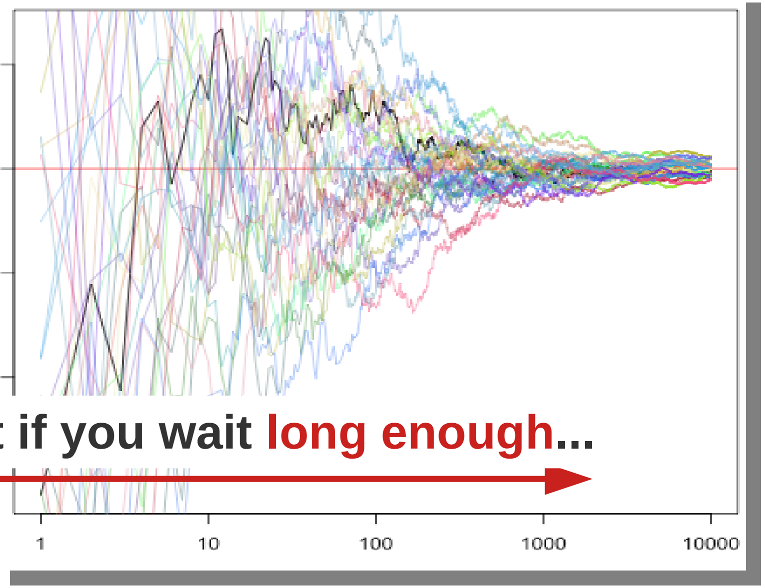
The sampling problem

$$\nabla \mathcal{L} = \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\text{data}}} - \langle -\nabla E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}}$$

Easy

Hard ⇒ Markov Chain MC sampling

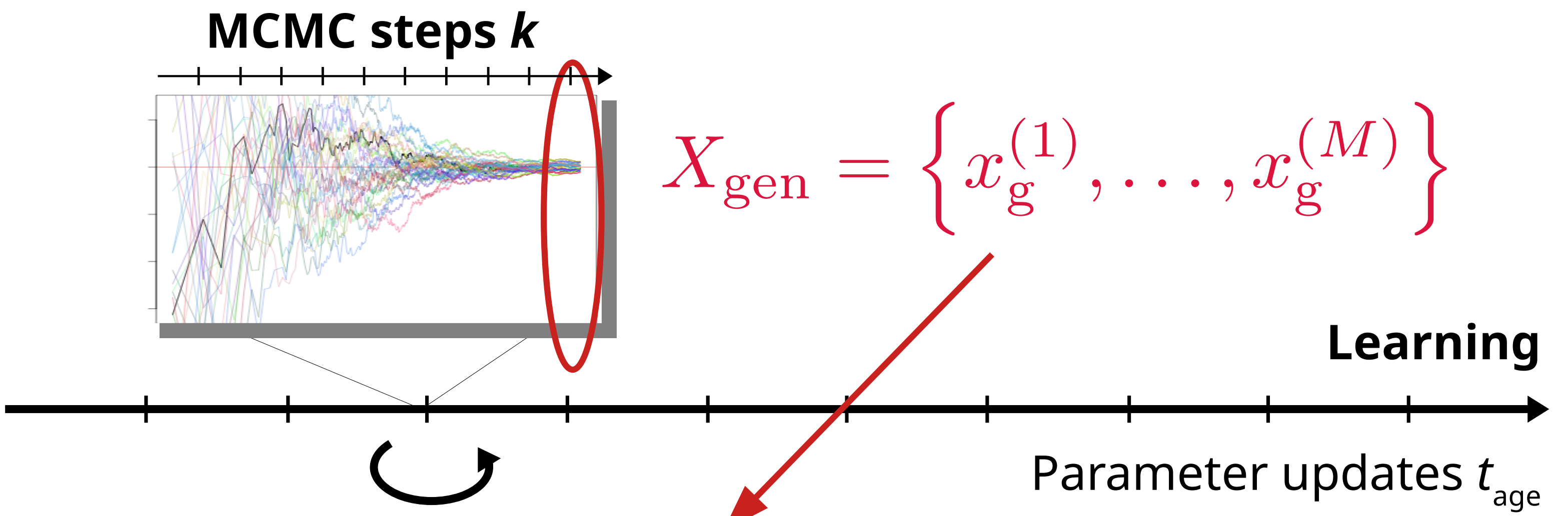
$p_0(x)$



But if you wait long enough...

k MCMC steps > thermalization/convergence time

Gibbs sampling + learning



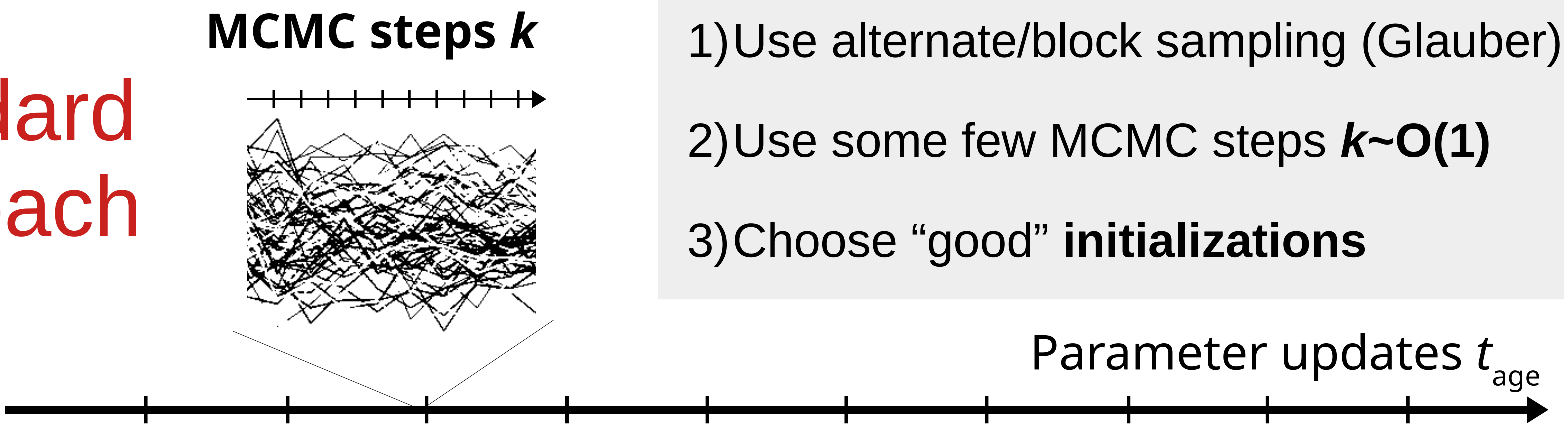
$$X_{\text{gen}} = \left\{ x_{\text{g}}^{(1)}, \ldots, x_{\text{g}}^{(M)} \right\}$$

$$\theta_i^{(t+1)} \leftarrow \theta_i^t + \gamma \left.\frac{\partial \mathcal{L}}{\partial \theta_i}\right|_{\theta=\theta_i^{(t)}}$$

Repeat this process
~10^5-10^6 times

Gibbs sampling + learning

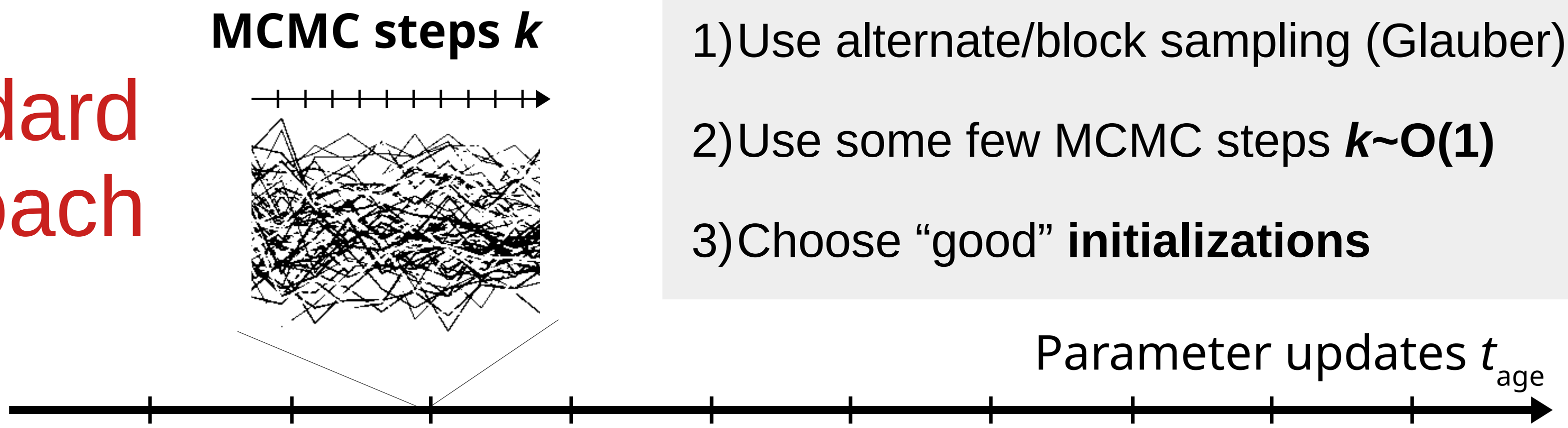
Standard approach



1) Use alternate/block sampling (Glauber) dynamics
2) Use some few MCMC steps ***k*~O(1)**
3) Choose “good” **initializations**

Gibbs sampling + learning

Standard approach

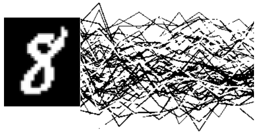


1) Use alternate/block sampling (Glauber) dynamics

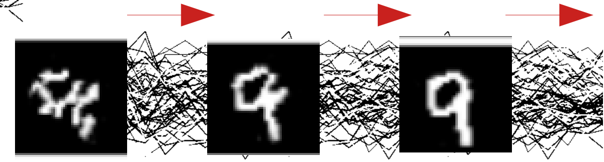
2) Use some few MCMC steps ***k*~O(1)**

3) Choose “good” **initializations**

- Contrastive divergence (CD) *[Hinton (2002)]* **Init – dataset**

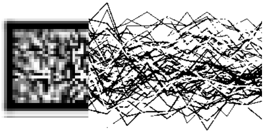


- Persistence CD (PCD) *[Tieleman (ICML 2008)]* **Init – previous end point**



- **Other solutions :** TAP fixed points *[Gabrié, Tramel, and Krzakala (NeurIPS 2015)],*
  optimized sampling techniques (Parallel tempering, Simulated annealing, the Tethered MC method]

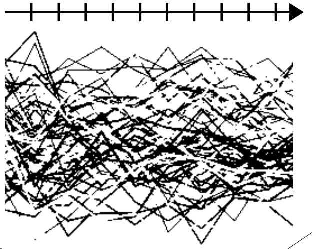
- **Random initialization**
  - *[Nijkamp, Hill, Han, Wu, Zhu., (NeurIPS 2019*
  - *Decelle, Furtlehner, Seoane (NeurIPS 2021)]*



Gibbs sampling + learning

Standard approach

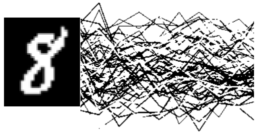
MCMC steps *k*



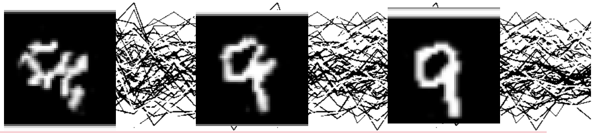
1) Use alternate/block sampling (Glauber) dynamics
2) Use some few MCMC steps ***k*~O(1)**
3) Choose "good" **initializations**

Parameter updates t_{age}

- Contrastive divergence (CD) *[Hinton (2002)]* **Init – dataset**



- Persistence CD (PCD) *[Tieleman (ICML 2008)]* **Init – previous end point**



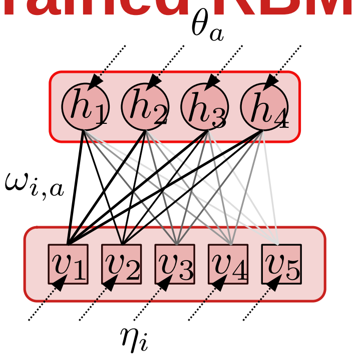
- **Other solutions :** TAP fixed points *[Gabrié, Tramel, and Krzakala (NeurIPS 2015)],*

Total disinterest in controlling whether *k* was long enough to ensure a proper sampling

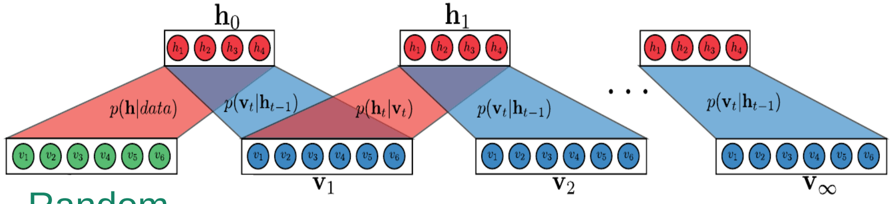
Decelle, Furtlehner, Seoane (NeurIPS 2021)]

Generating new samples

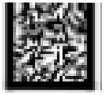
Trained RBM



Alternating/Block **Gibbs sampling**



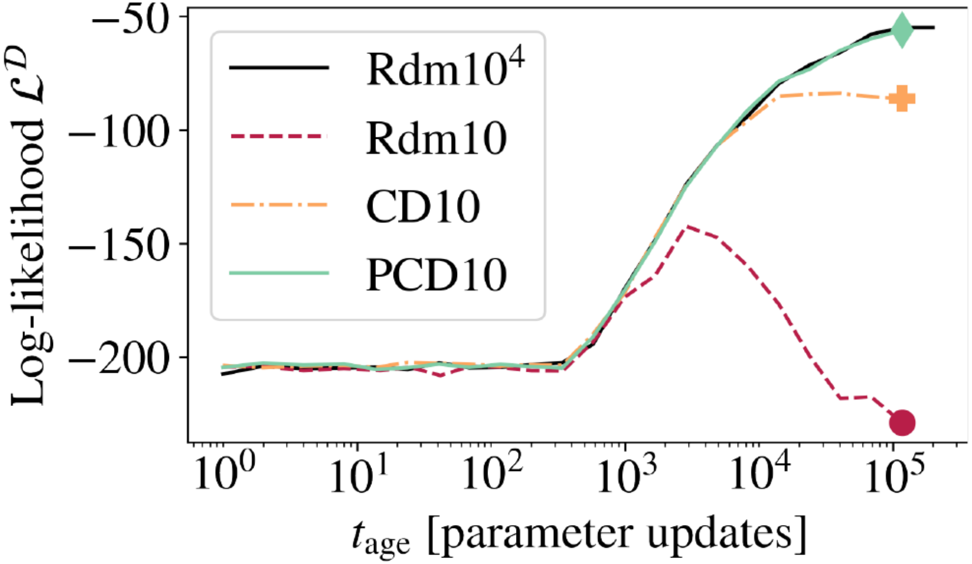
Random configuration

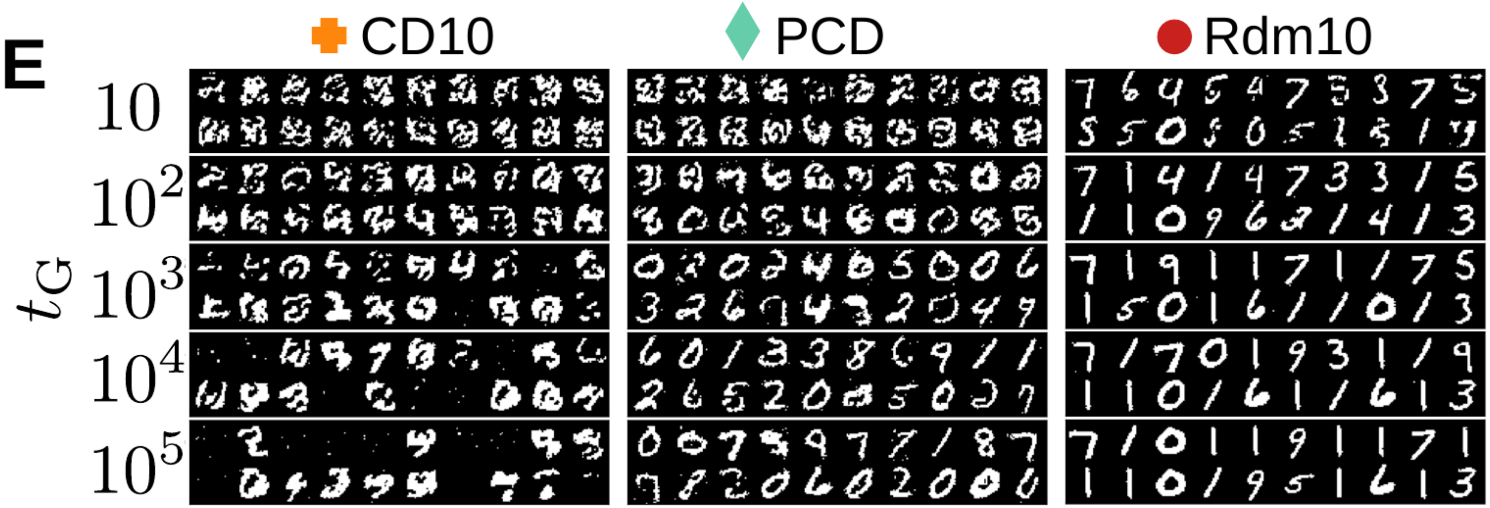


Converge to equilibrium

Generating samples : typical situation

The typical measures used to control the learning are not a good estimator of the quality of the generator



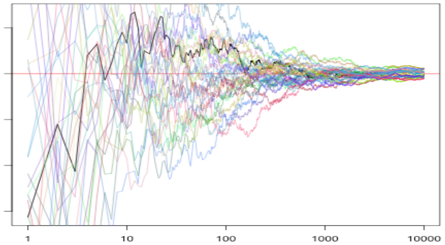


Equilibrium vs. Non-eq. regimes

Decelle, Furtlehner, Seoane
ArXiv:2105.13889
(NeurIPS2021)

Training

$\nabla_{\Theta}\mathcal{L}$



k

- **Equilibrium** $k > t_{\text{therm}}$
  (Convergent MCMC)

- **Non-equilibrium** $k < t_{\text{therm}}$
  (Non-convergent MCMC)

Sampling

$$p_{\text{data}}(x) \sim p_{\boldsymbol{\theta}}(x) = \frac{e^{-E_{\boldsymbol{\theta}}(x)}}{Z}$$

- **Learns a good model for the data**

$$p_{\text{data}}(x) \sim p(k, \boldsymbol{p}_0, x)$$

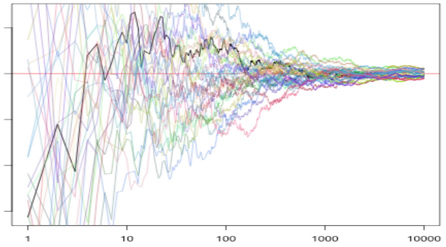
- **“Learns the dynamics”**
  Memory effects

Equilibrium vs. Non-eq. regimes

Decelle, Furtlehner, Seoane
ArXiv:2105.13889
(NeurIPS2021)

Training

$\nabla_{\Theta}\mathcal{L}$



k

- **Equilibrium** $k > t_{\mathrm{therm}}$
  (Convergent MCMC)

- **Non-equilibrium** $k < t_{\mathrm{therm}}$
  (Non-convergent MCMC)

Sampling

Good for modeling

$$p_{\mathrm{data}}(x) \sim p_{\boldsymbol{\theta}}(x) = \frac{e^{-E_{\boldsymbol{\theta}}(x)}}{Z}$$

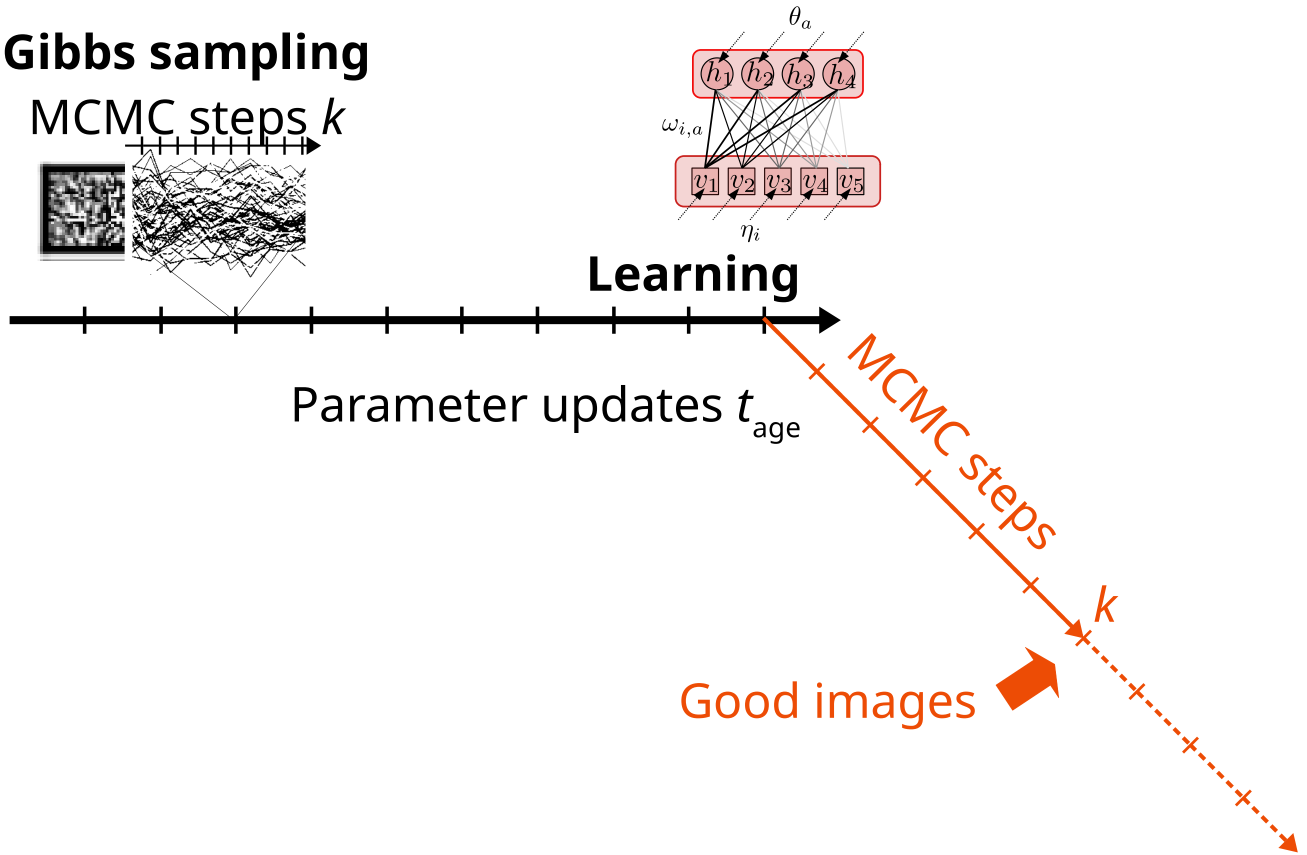
- **Learns a good model for the data**

Optimal for generation

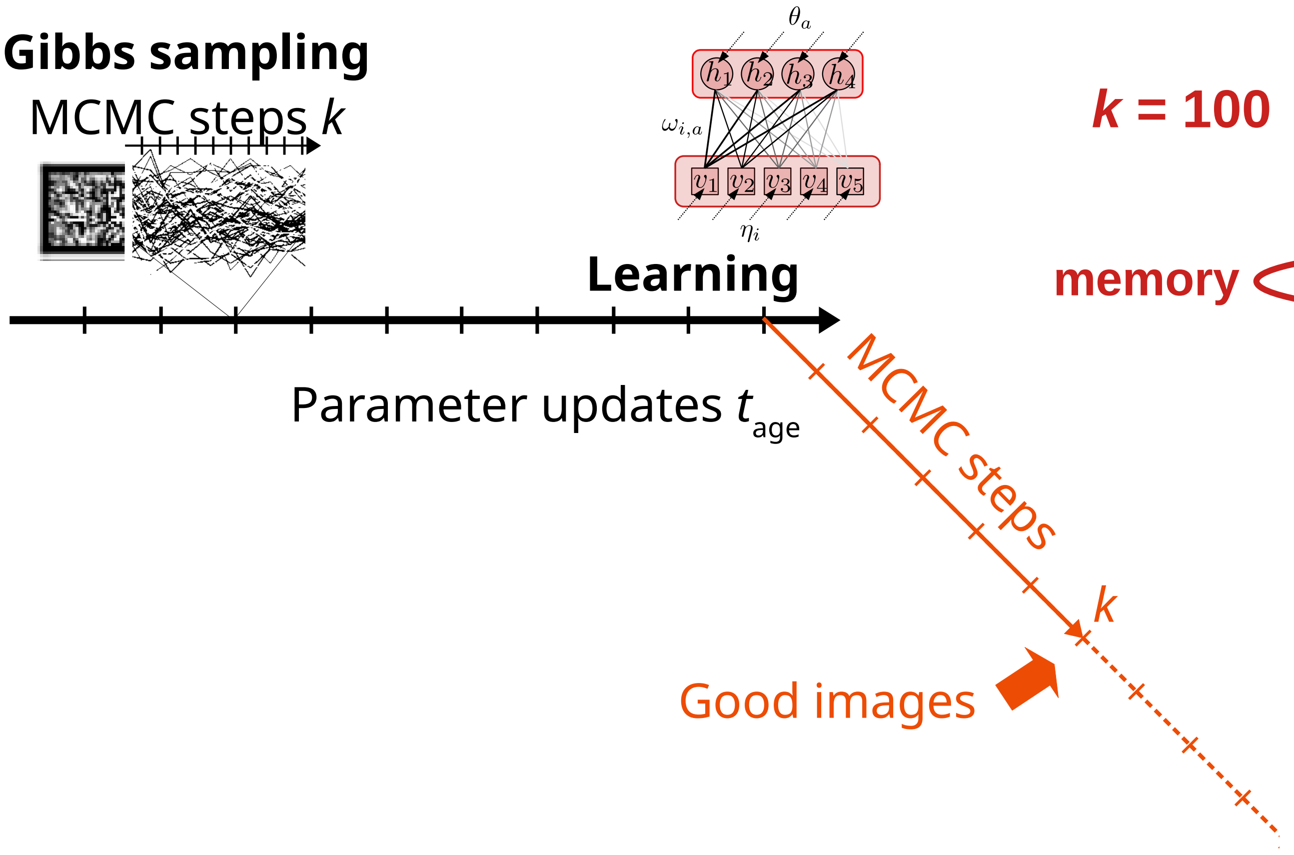
$$p_{\mathrm{data}}(x) \sim p(k, \boldsymbol{p}_0, x)$$

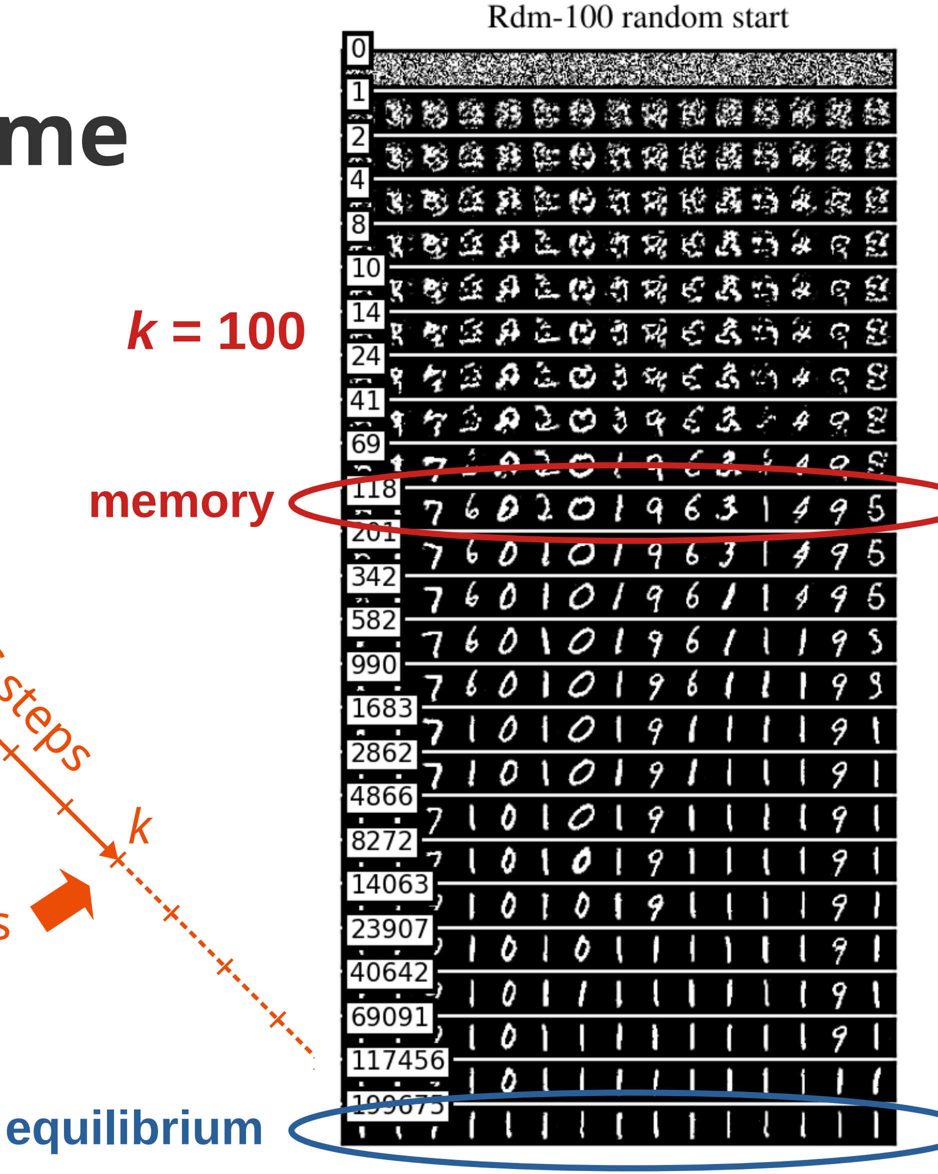
- **“Learns the dynamics”**
  Memory effects

Out-of-equilibrium regime

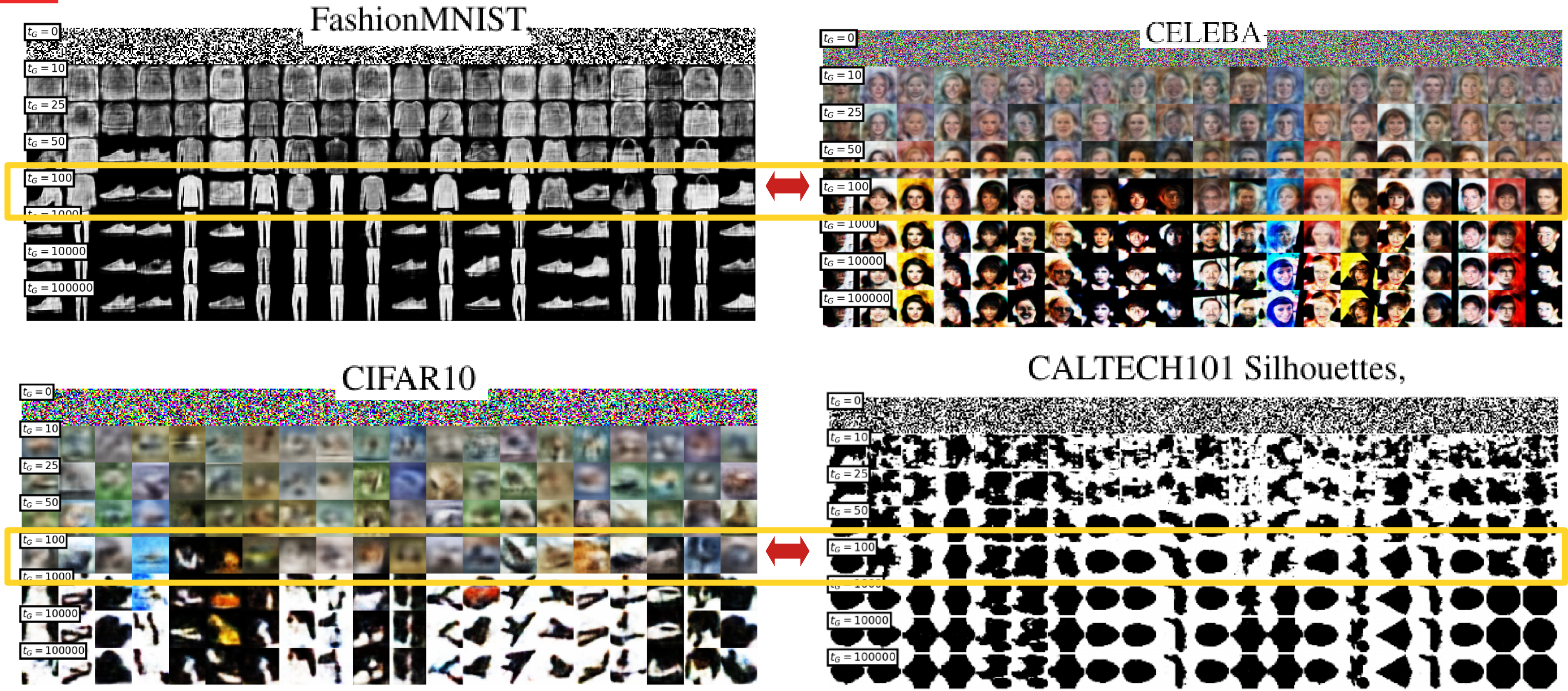


Out-of-equilibrium regime

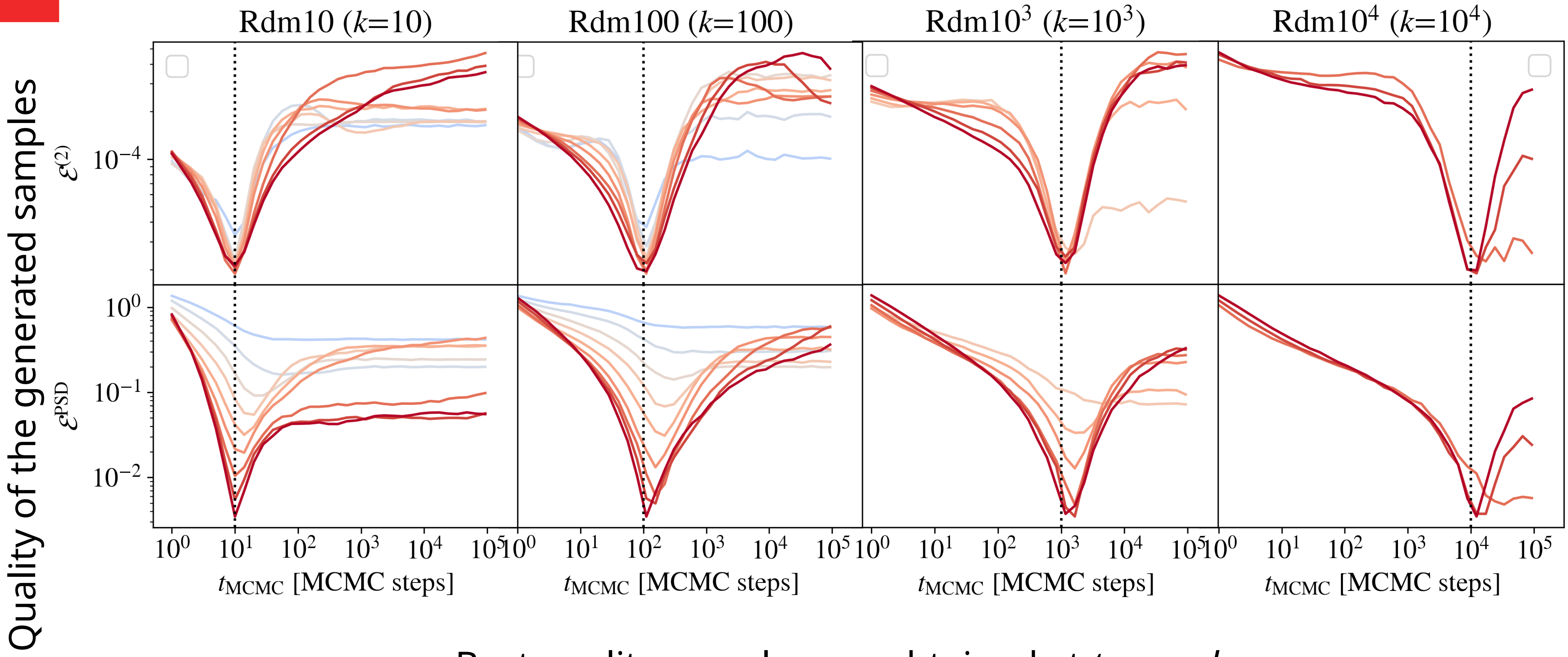




Out-of-equilibrium regime



Non-equilibrium regime : generation

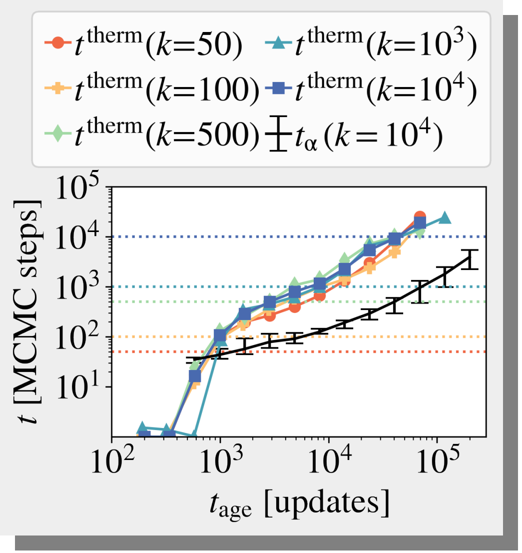


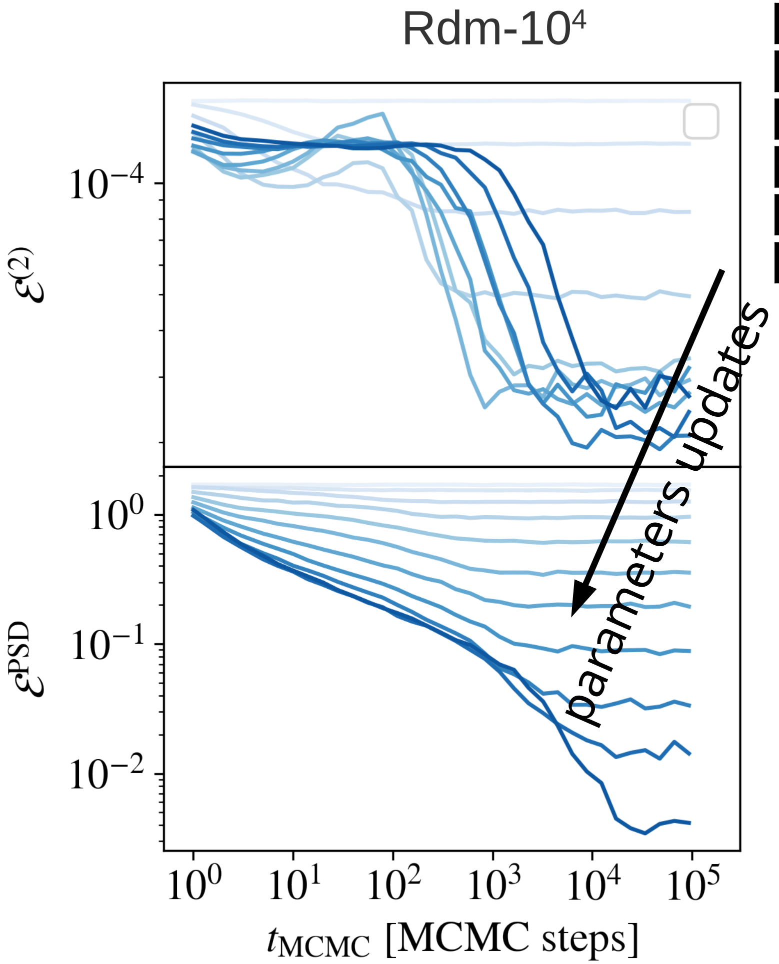
Best quality samples are obtained at $t_{\mathrm{MCMC}} \sim k$

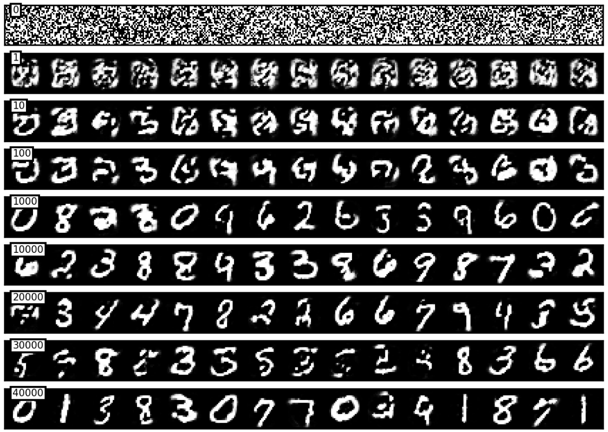
[Decelle, Furtlehner, Seoane NeurIPS (2021)]

Equilibrium regime

Memory effects are caused by the lack of convergence of the Markov chains

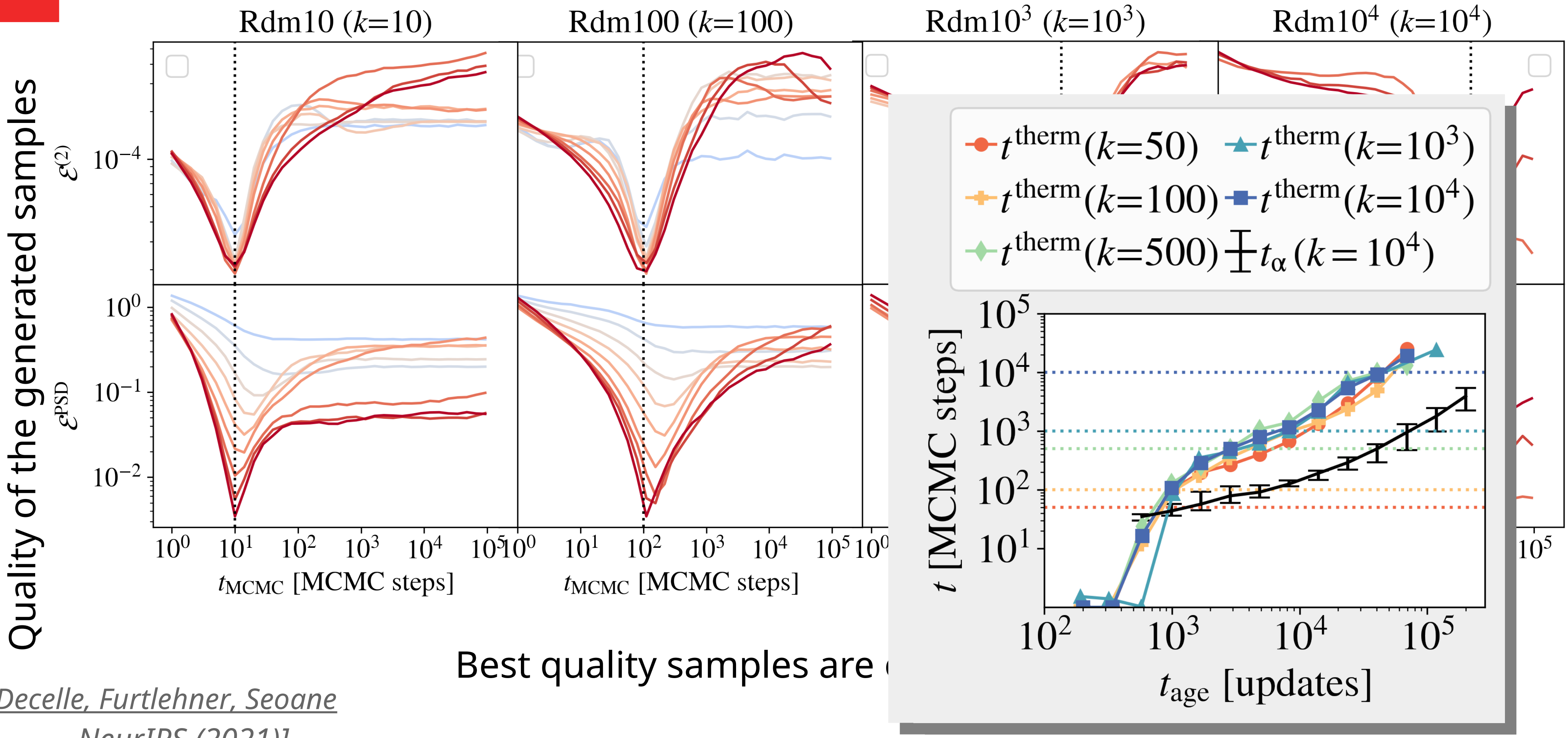






Dynamics are much faster

Non-equilibrium regime : generation

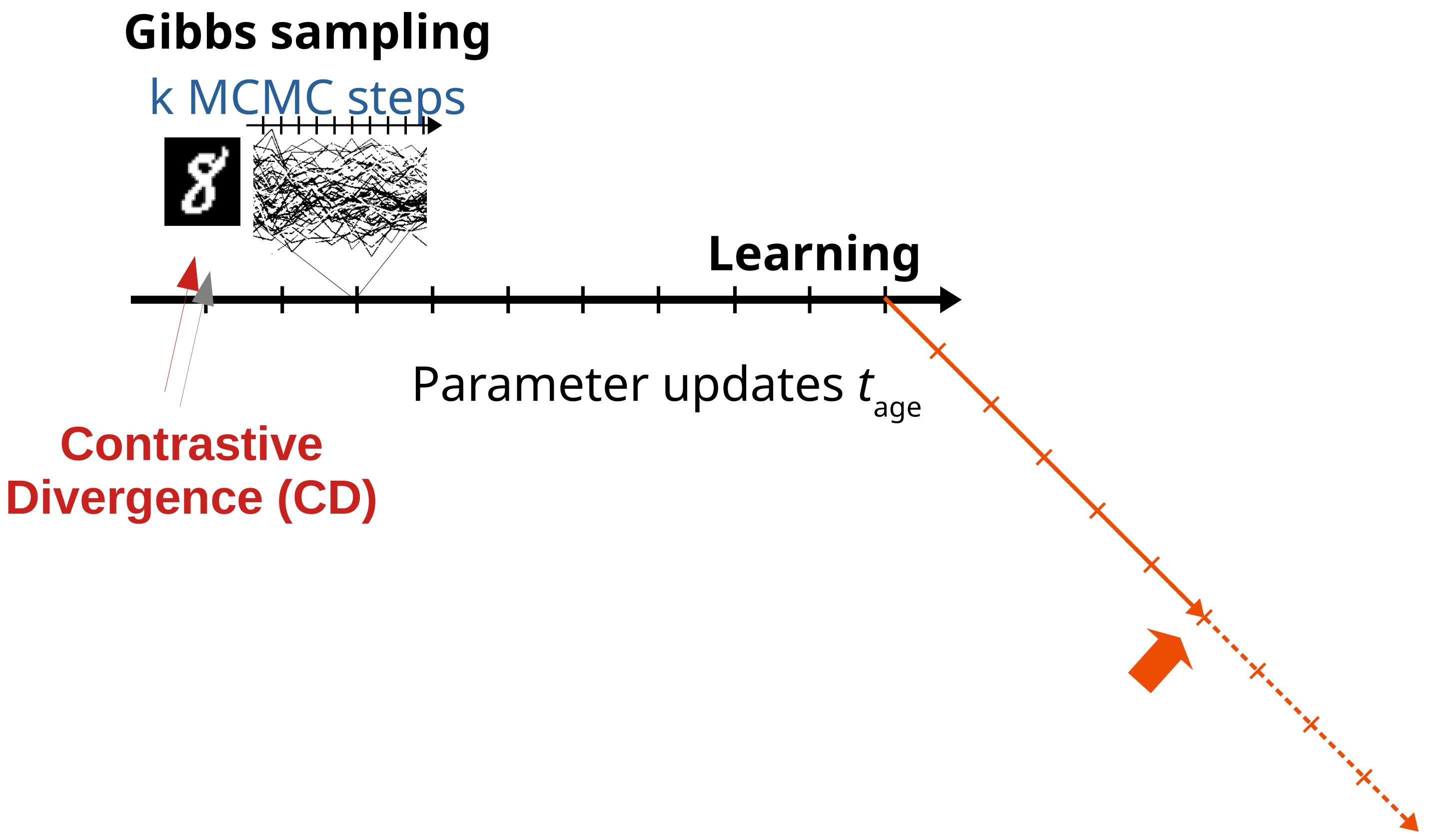


Best quality samples are (

[Decelle, Furtlehner, Seoane NeurIPS (2021)]

Dangers of Contrastive Divergence (CD)

Hinton, *"A practical guide to training restricted Boltzmann machines"* (2012) k=1

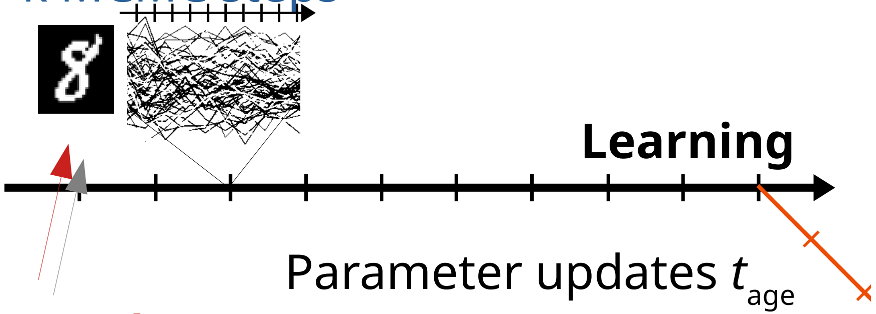


Dangers of Contrastive Divergence (CD)

Hinton, *"A practical guide to training restricted Boltzmann machines"* (2012) k=1

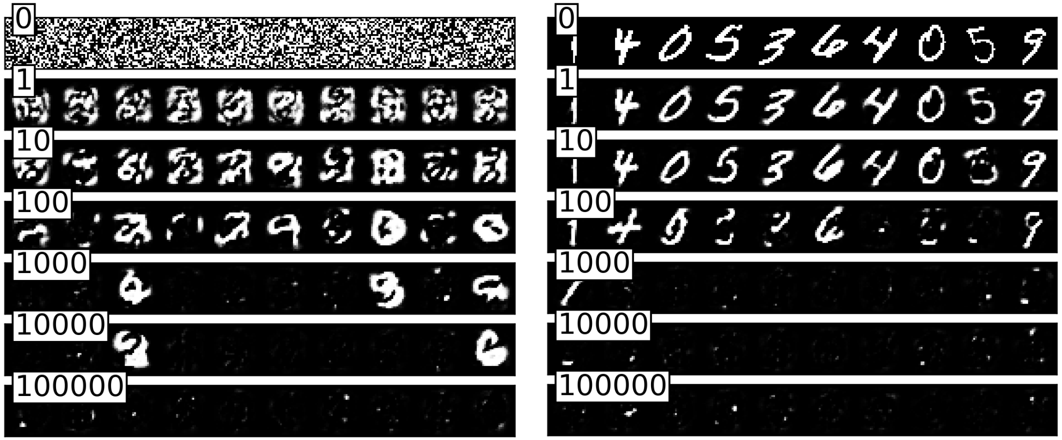
Gibbs sampling

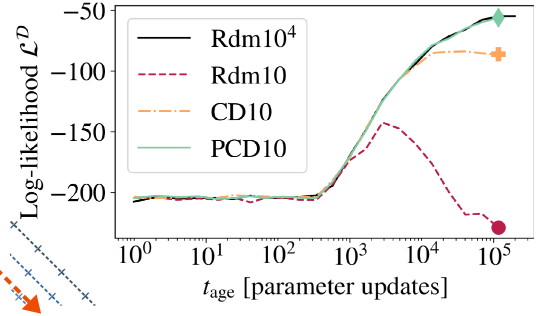
k MCMC steps



Contrastive Divergence (CD)

CD-10



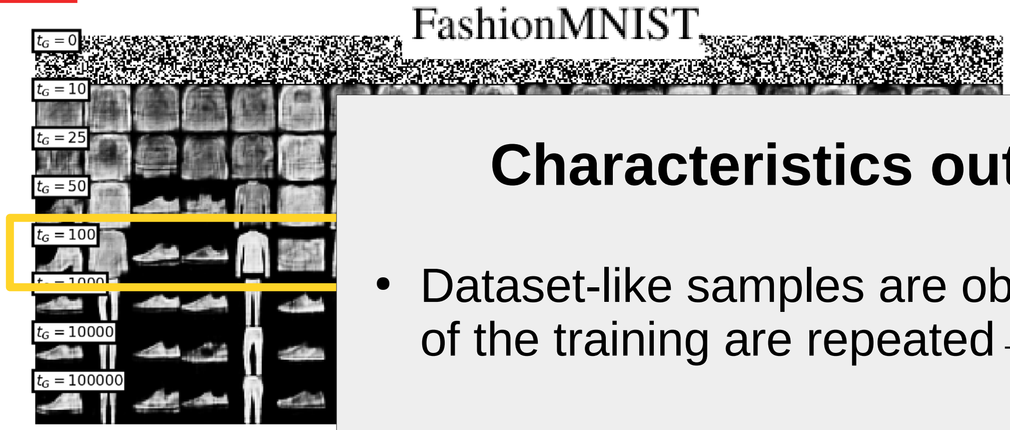


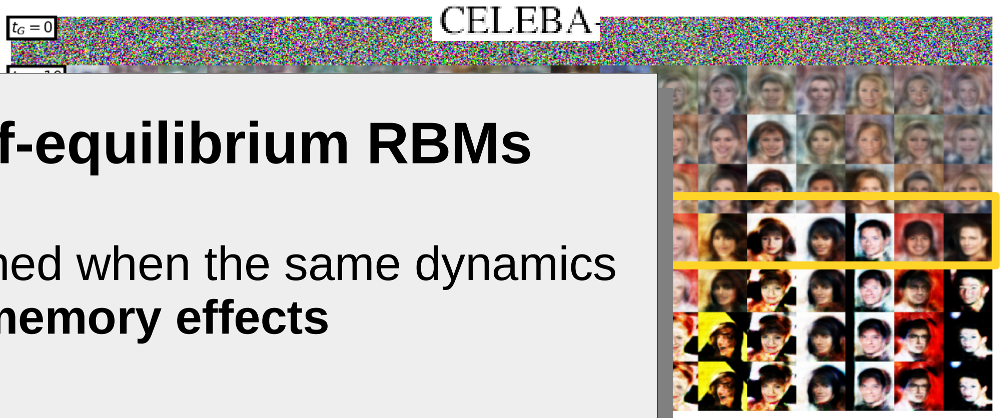
CD is a very bad recipe !

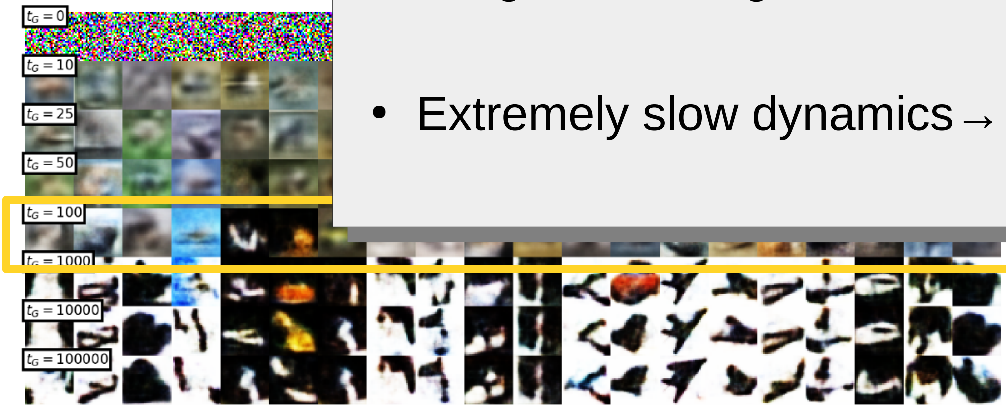
Salakhutdinov, R., & Murray, I. ICML (2008)

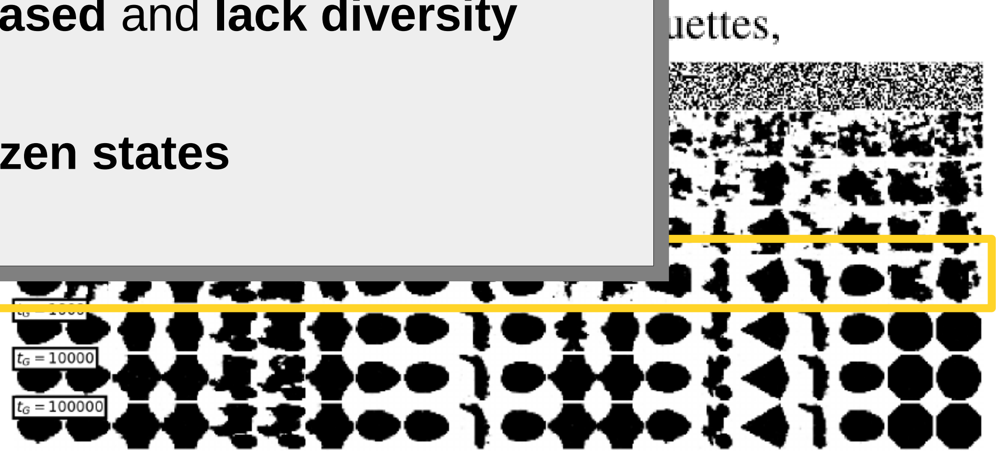
Desjardins, Courville, Bengio, Vincent, Delalleau, AISTATS, 2010

Out-of-equilibrium regime







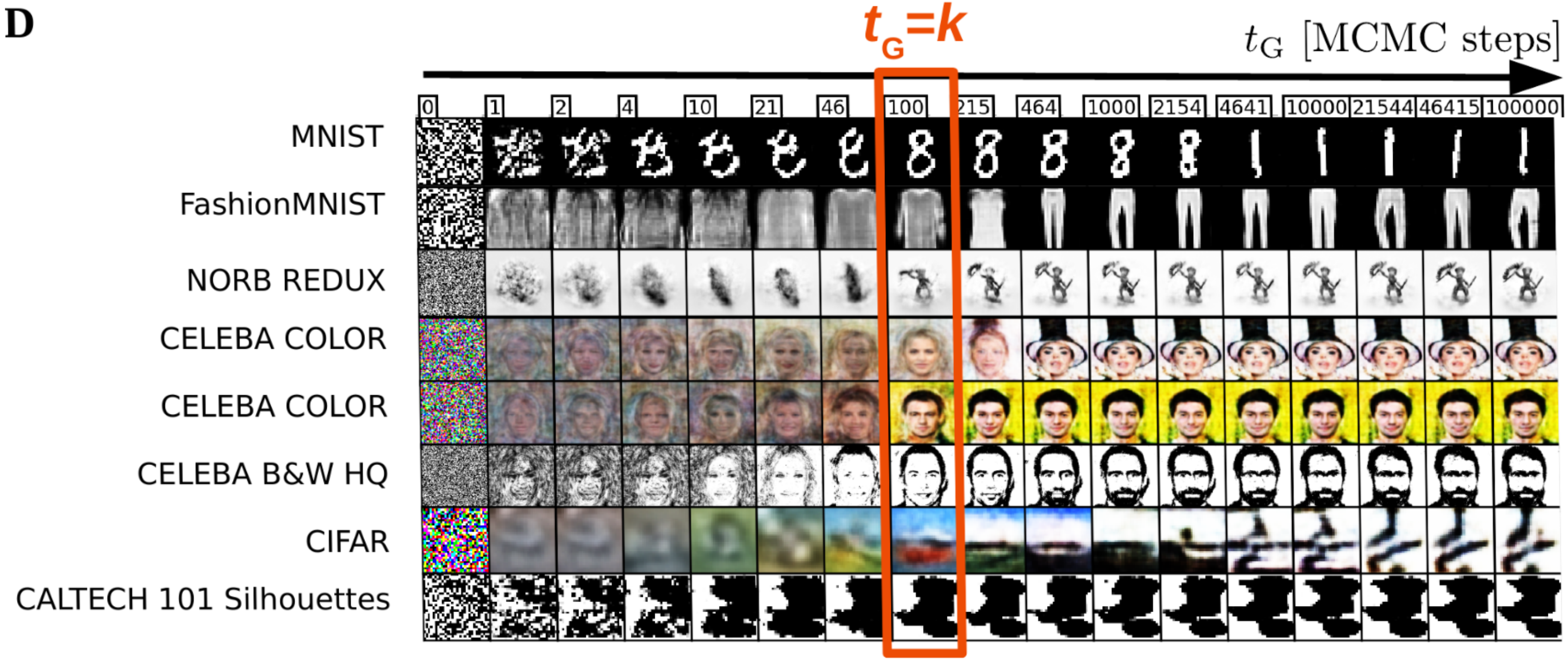


Characteristics out-of-equilibrium RBMs

- Dataset-like samples are obtained when the same dynamics of the training are repeated → **memory effects**
- Long-time configurations are **biased** and **lack diversity**
- Extremely slow dynamics → **frozen states**

Out-of-equilibrium regime :
best for sample generation

[Decelle, Furtlehner, Seoane NeurIPS (2021)]

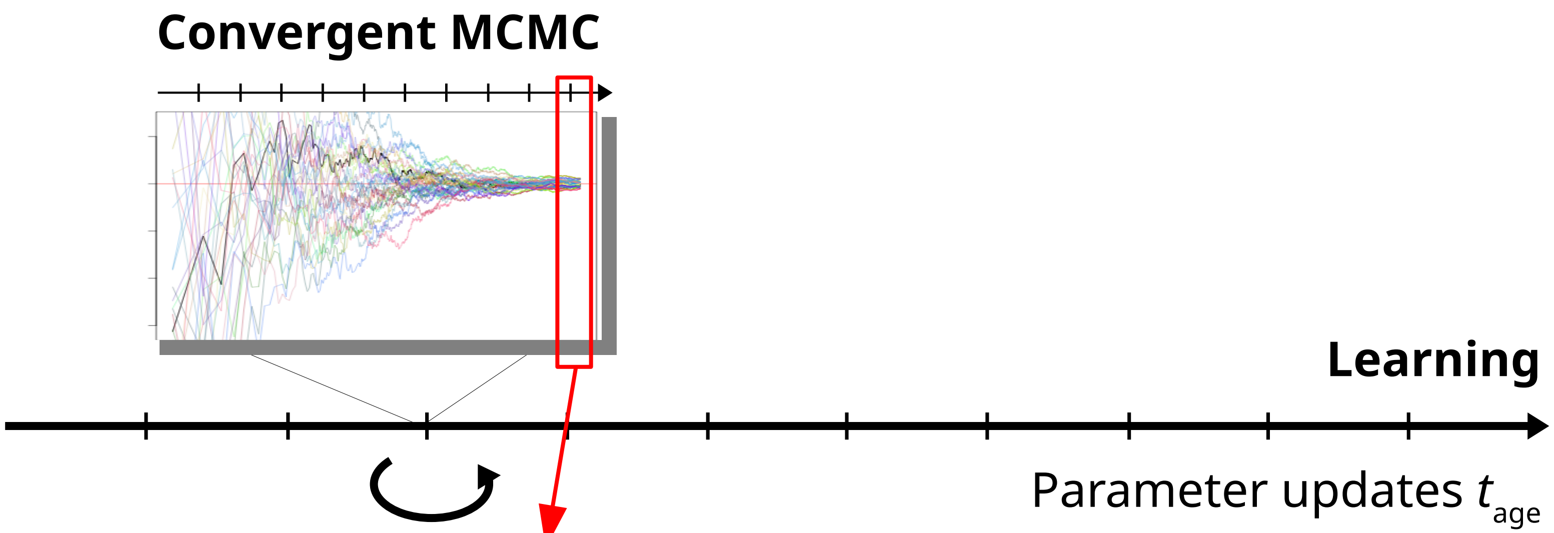


Deep Convnets: Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI 2020. **Boltzmann Machines:** Muntoni, Pagnani, Martin Weigt, Zamponi BMC (2021)

Explanation effects

Agoritsas, Catania, Decelle, Seoane ICML (2023)

Gibbs sampling + learning



$$\boldsymbol{\theta}(t+t) \longleftarrow \boldsymbol{\theta}(t) + \gamma \boldsymbol{\nabla}\mathcal{L}(t)$$

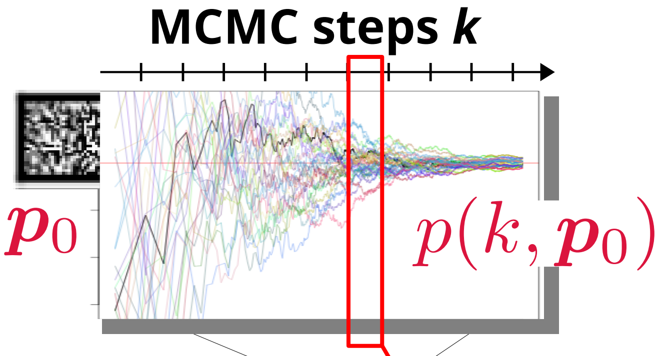
$$\boldsymbol{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E \rangle_{p_{\text{data}}} - \langle -\boldsymbol{\nabla} E \rangle_{p_{\boldsymbol{\theta}}}$$

Gibbs sampling + learning

$\boldsymbol{\mathcal{U}}_{\boldsymbol{\theta}}$: stochastic matrix

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \boldsymbol{\mathcal{U}}_{\boldsymbol{\theta}}^k \, \boldsymbol{p}_0$$

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \boldsymbol{p}_{\boldsymbol{\theta}} + \sum_{\alpha > 0} c_\alpha(\boldsymbol{p}_0) \, e^{-k/\kappa_\alpha} \boldsymbol{u}_\alpha$$



$$\boldsymbol{\theta}(t + t) \longleftarrow \boldsymbol{\theta}(t) + \gamma \boldsymbol{\nabla} \mathcal{L}(t)$$

$$\boldsymbol{\nabla} \mathcal{L} = \langle -\boldsymbol{\nabla} E \rangle_{p_{\mathrm{data}}} - \langle -\boldsymbol{\nabla} E \rangle_{p_{\boldsymbol{\theta}}}$$

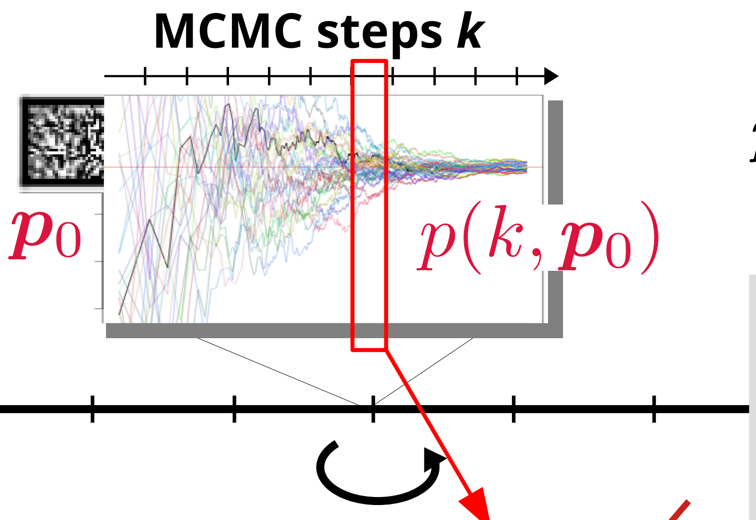
-Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI (2020)
-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Gibbs sampling + learning

$\mathcal{U}_{\boldsymbol{\theta}}$: stochastic matrix

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \mathcal{U}_{\boldsymbol{\theta}}^k \boldsymbol{p}_0$$

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \boldsymbol{p}_{\boldsymbol{\theta}} + \sum_{\alpha>0} c_\alpha(\boldsymbol{p}_0)\, e^{-k/\kappa_\alpha} \boldsymbol{u}_\alpha$$



$$\boldsymbol{\theta}(t+t) \longleftarrow \boldsymbol{\theta}(t) + \gamma \boldsymbol{\nabla} \mathcal{L}(t)$$

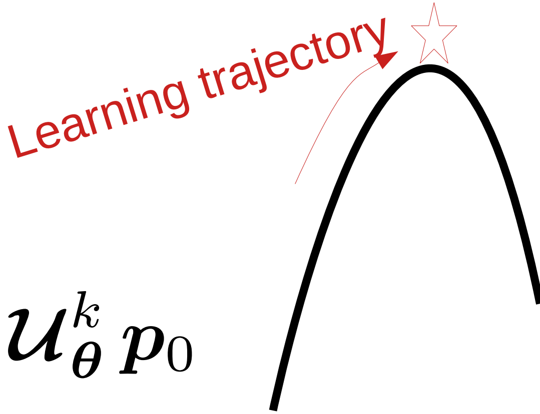
$$\boldsymbol{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E \rangle_{p_{\rm data}} - \langle -\boldsymbol{\nabla} E \rangle_{p_{\boldsymbol{\theta}}}$$

$$\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} \neq \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p(k, \boldsymbol{p}_0)}$$

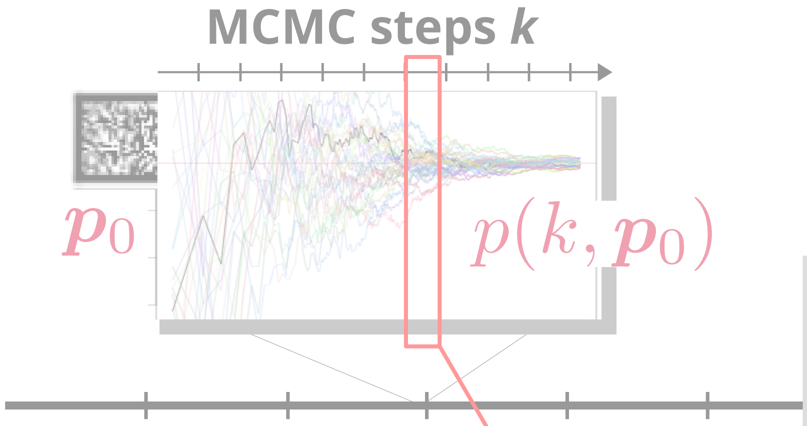
$$\tilde{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\rm data}} - \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{\boldsymbol{p}(k, \boldsymbol{p}_0)}$$

-Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI (2020)
-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Gibbs sampling + learning



$$p(k, \boldsymbol{p}_0) = \boldsymbol{\mathcal{U}}_{\boldsymbol{\theta}}^{k}\, \boldsymbol{p}_0$$



$$\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} \neq \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p(k, \boldsymbol{p}_0)}$$

$$\tilde{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\rm data}} - \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p(k, \boldsymbol{p}_0)}$$

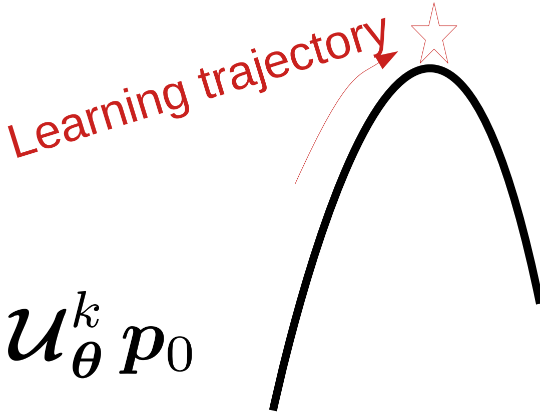
Fixed point : $\nabla \mathcal{L}_{\theta^\star} = 0$

$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\rm data}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\boldsymbol{\theta}^\star}}$$

$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\rm data}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k, \boldsymbol{p}_0)}$$

-Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI (2020)
-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Gibbs sampling + learning



$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\sim \boldsymbol{p}_{\text{data}}(x)$$

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \boldsymbol{\mathcal{U}}_{\boldsymbol{\theta}}^{k}\, \boldsymbol{p}_0$$

Trainings with non-convergent MCMC fit **bad models** for the data

$$\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} \neq \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p(k,\boldsymbol{p}_0)}$$

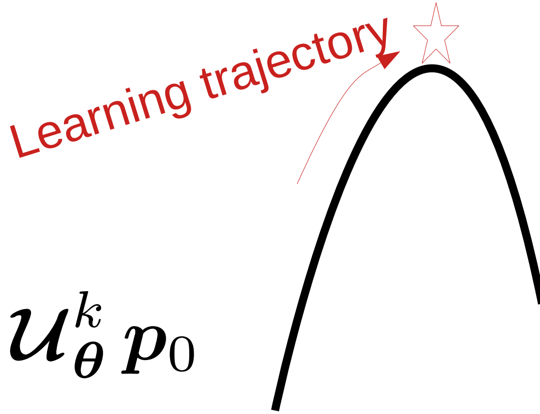
$$\tilde{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\text{data}}} - \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{\boldsymbol{p}(k,\boldsymbol{p}_0)}$$

~~$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\text{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\boldsymbol{\theta}^\star}}$$~~

$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\text{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$$

-Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI (2020)
-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Gibbs sampling + learning



Similarly:

$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\approx \boldsymbol{p}_{\mathrm{data}}(x)$$

$$\boldsymbol{p}(k, \boldsymbol{p}_0) = \boldsymbol{\mathcal{U}}_{\boldsymbol{\theta}}^k \boldsymbol{p}_0$$

Trainings with non-convergent MCMC fit **bad models** for the data

$$\langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} \neq \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p(k, \boldsymbol{p}_0)}$$

$$\tilde{\nabla}\mathcal{L} = \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{p_{\mathrm{data}}} - \langle -\boldsymbol{\nabla} E_{\boldsymbol{\theta}} \rangle_{\boldsymbol{p}(k, \boldsymbol{p}_0)}$$

But they still can be used as **very good generative models**

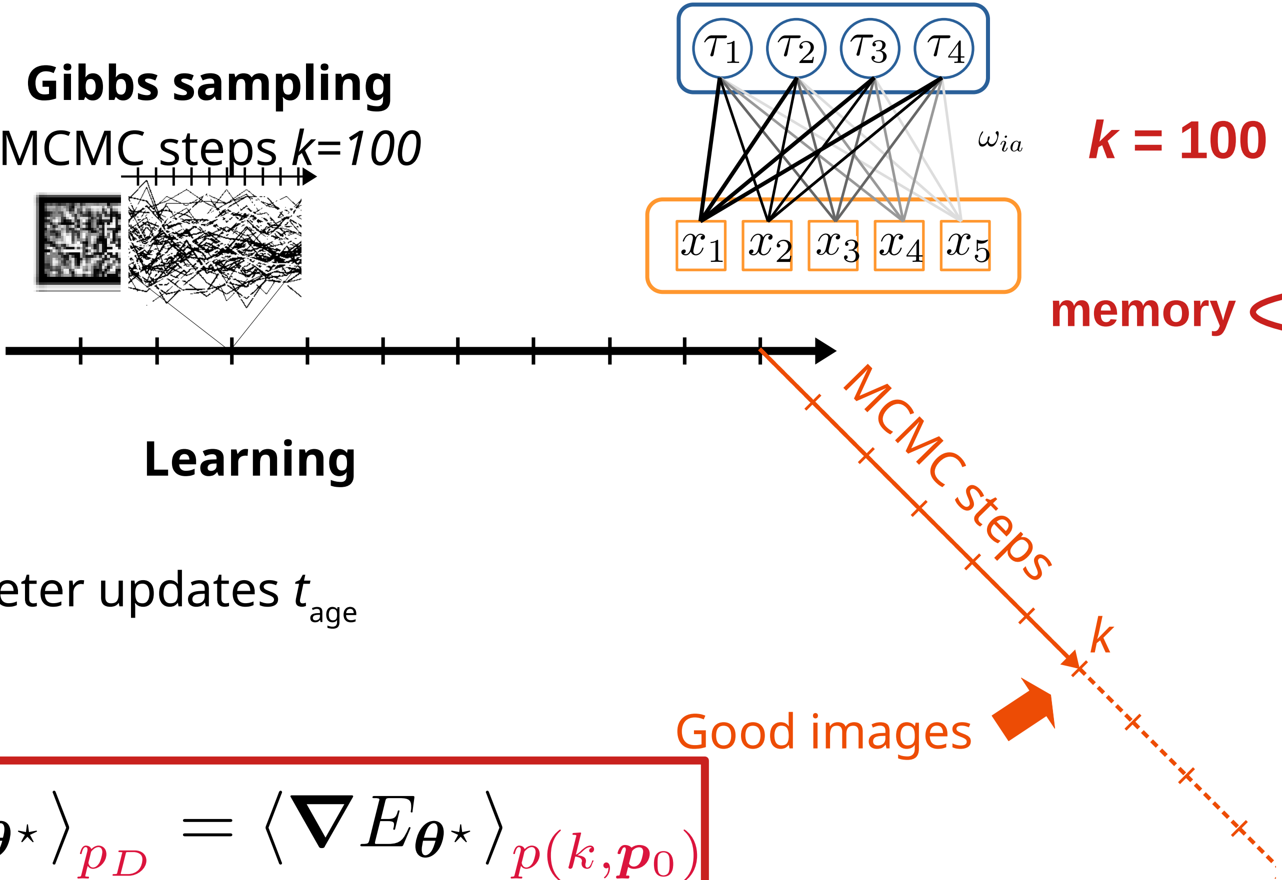
$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\mathrm{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k, \boldsymbol{p}_0)}$$

-Nijkamp, Hill, Han, Wu, Zhu. NeurIPS 2019, AAAI (2020)
-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Non-equilibrium regime

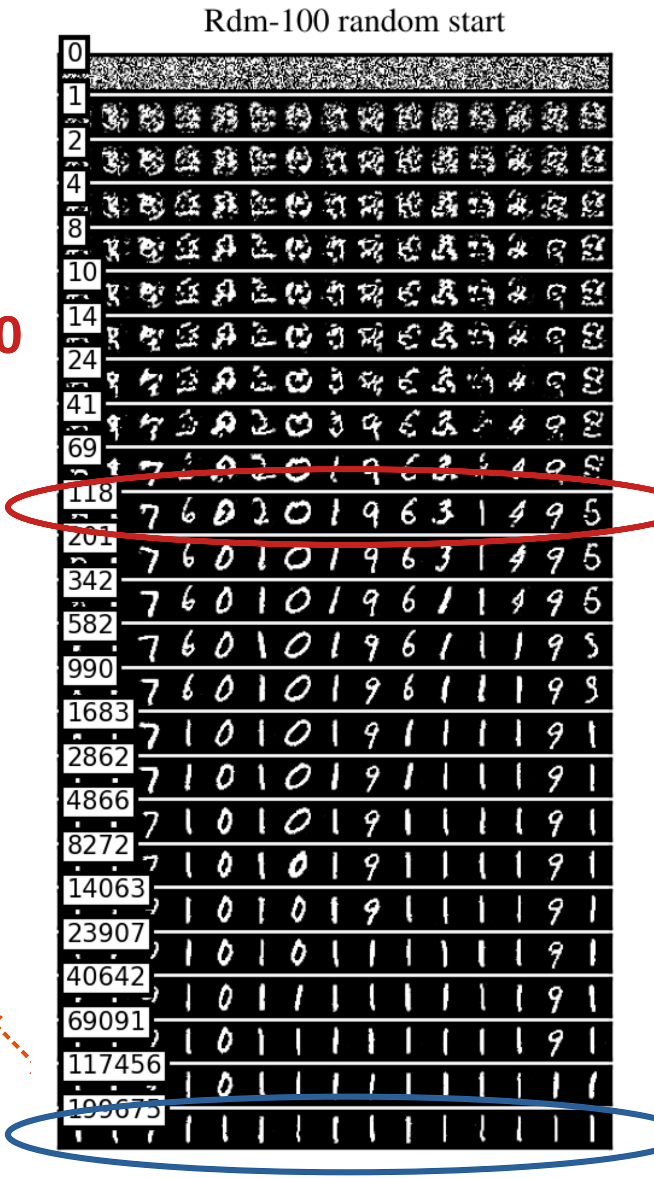
Gibbs sampling

MCMC steps *k=100*



Parameter updates t_{age}

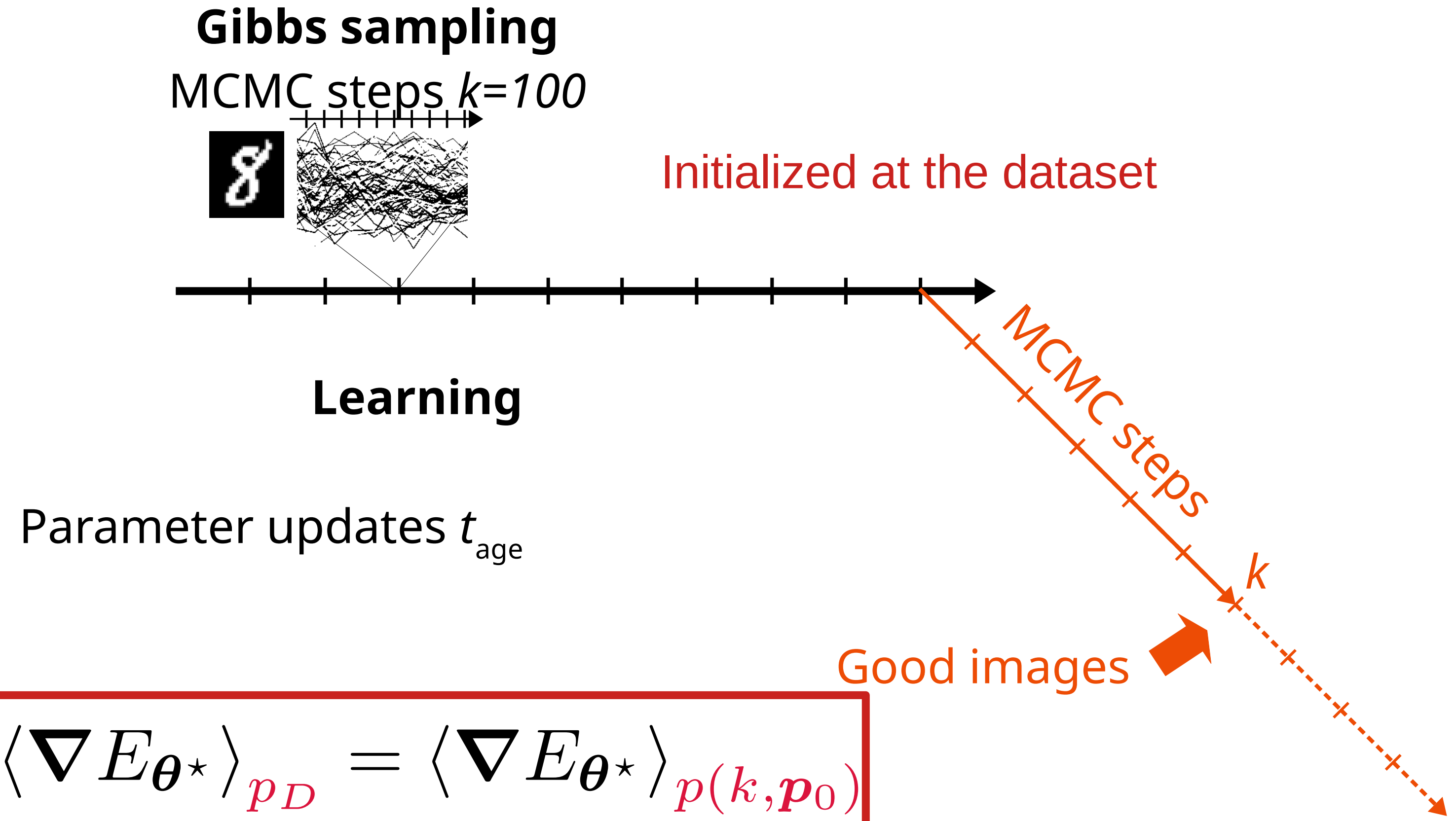
k = 100

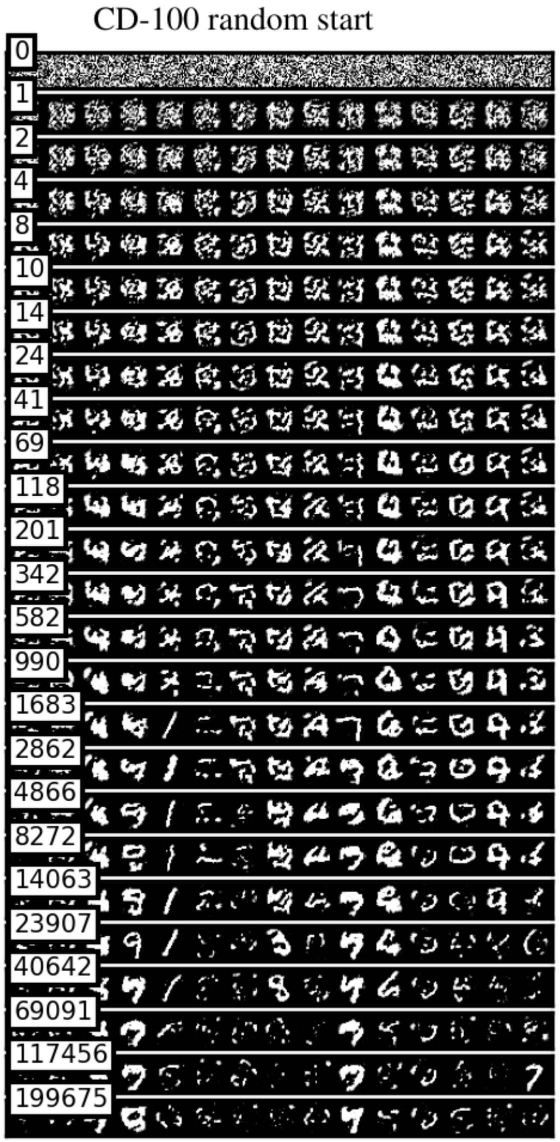


$$\langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p(k, \boldsymbol{p}_0)}$$

Decelle, Furtlehner, Seoane ArXiv:2105.13889 NeurIPS (2021)

Change initial conditions



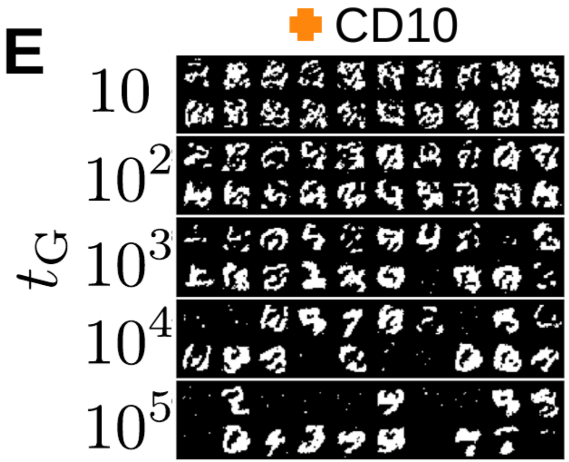


$$\langle \nabla E_{\theta^\star} \rangle_{p_D} = \langle \nabla E_{\theta^\star} \rangle_{p(k,\mathbf{p}_0)}$$

Decelle, Furtlehner, Seoane ArXiv:2105.13889 NeurIPS (2021)

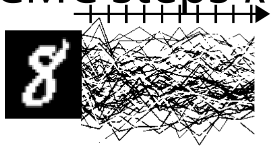
Changing the sampling process...

Change the initial conditions :

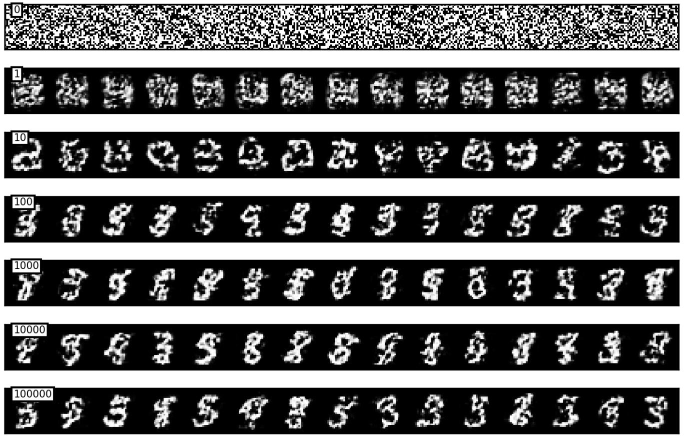


Gibbs sampling

MCMC steps *k=10*



Change the update rules:



RBM trained using

Gabrié, M., Tramel, E. W., & Krzakala, F. (2015). NeurIPS

Out-of-equilibrium regime :
best for sample generation

Generative Convnets:
Nijkamp, Hill, Han, Wu, Zhu.
NeurIPS 2019, AAAI 2020.

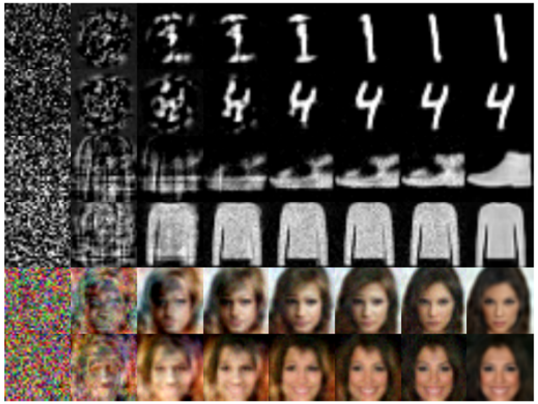


Figure 2: Intermediate samples from Gibbs-Langevin sampling.

GAUSSIAN-BERNOULLI RBMS WITHOUT TEARS

Renjie Liao*[1], **Simon Kornblith**[2], **Mengye Ren**[3], **David J. Fleet**[2,4,5], **Geoffrey Hinton**[2,4,5]
University of British Columbia[1], Google Research, Brain Team[2],
New York University[3], University of Toronto[4], Vector Institute[5]
rjliao@ece.ubc.ca, mengye@cs.nyu.edu
{skornblith, davidfleet, geoffhinton}@google.com

ArXiv: 2210.10318 (2022)

Published as a conference paper at ICLR 2022

A TALE OF TWO FLOWS: COOPERATIVE LEARNING OF LANGEVIN FLOW AND NORMALIZING FLOW TOWARD ENERGY-BASED MODEL

Jianwen Xie, Yaxuan Zhu, Jun Li, Ping Li

Consequences
(theorem)

Fixed point

$$\langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$$

[Agoritsas, Catania, Decelle, Seoane ICML (2023)]

1) Very good quality generation

Fixed point

EQ training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\boldsymbol{\theta}}}$

OOE training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$

↑

We reproduce the same set of statistics!

1) Very good quality generation

Fixed point

EQ training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\boldsymbol{\theta}}}$

OOE training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k, \boldsymbol{p}_0)}$

We reproduce the same set of statistics!

Samples generated using the OOE strategy should be as good as if the EBM was trained in EQ !

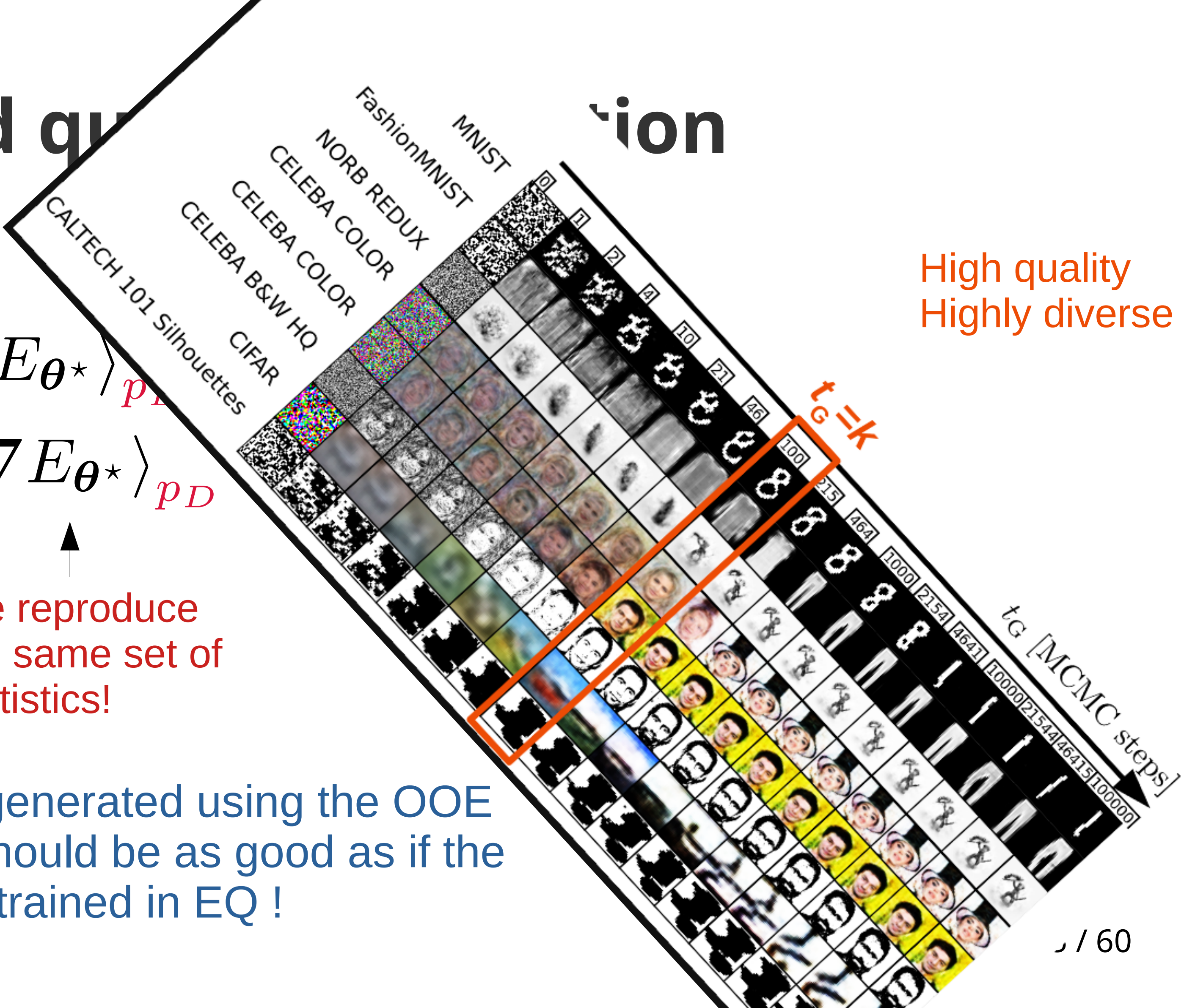
1) Very good qu … ion

Fixed point

EQ training $\langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p_D}$

OOE training $\langle \nabla E_{\boldsymbol{\theta}^\star} \rangle_{p_D}$

We reproduce the same set of statistics!



High quality
Highly diverse

Samples generated using the OOE strategy should be as good as if the EBM was trained in EQ !

2) Fast generators

Fixed point

We need to thermalize first !

EQ training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\boldsymbol{\theta}}}$

Typically very long...

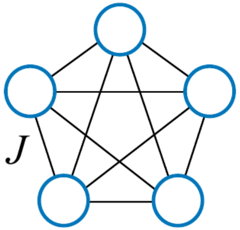
OOE training $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_D} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$

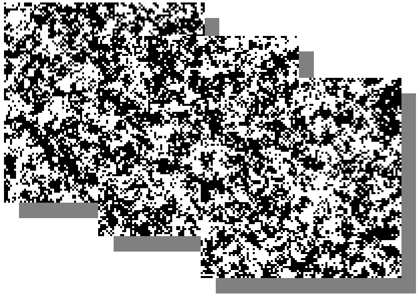
We just need k steps !

The quality does not depend on

- *k*,
- p_0,
- the dynamic rules used for sampling

2) Fast generators : Inverse Ising





Training set : Equilibrium samples of the Ising model

$$\{s_i^{(m)}\}_{m=1,\dots,M} \sim p(\boldsymbol{s}) = \frac{1}{Z(\beta)} e^{\beta \sum_{i<j} J_{ij} s_i s_j + \beta \sum_i h_i s_i}$$

Infer J by training a Boltzmann Machine (Ising model)

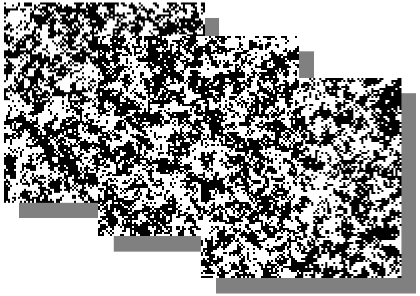
$$E(\boldsymbol{x}) = -\sum_{ij} J_{ij} x_i x_j - \sum_i h_i x_i$$

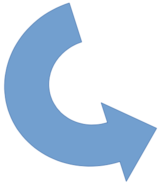
$$\langle S_i S_j \rangle_{p_{\mathrm{data}}} = \langle S_i S_j \rangle_{p(k,\boldsymbol{p}_0)}$$

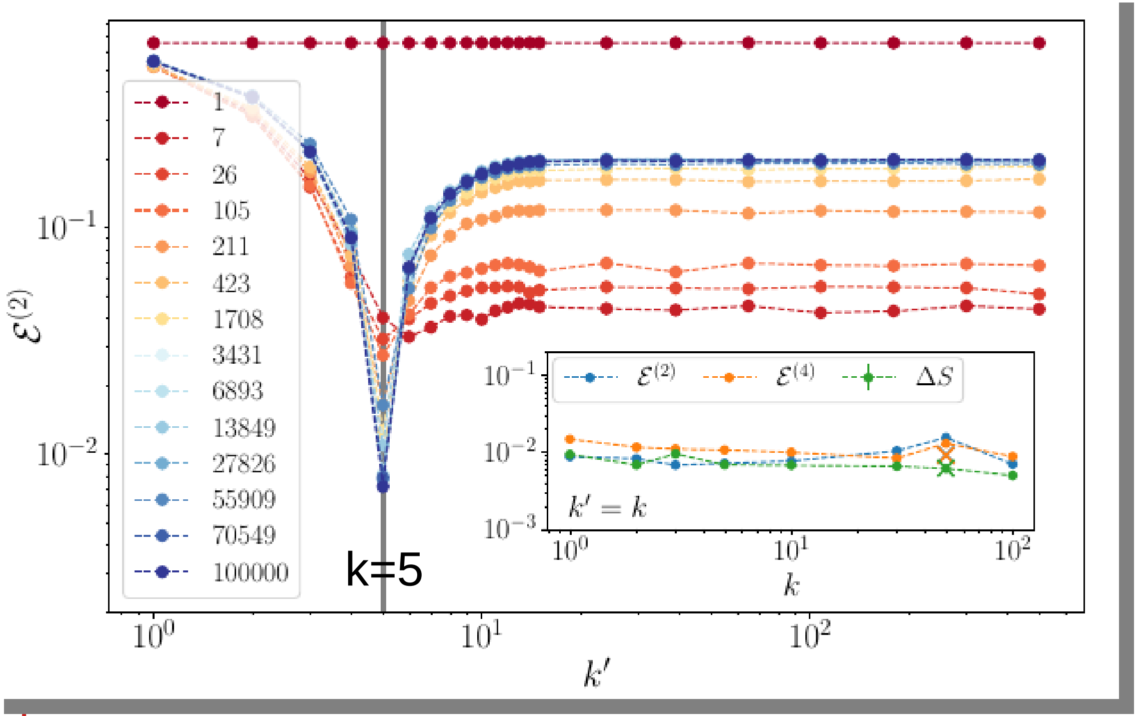
$$\langle S_i \rangle_{p_{\mathrm{data}}} = \langle S_i \rangle_{p(k,\boldsymbol{p}_0)}$$

$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\mathrm{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$$

2) Fast generators : Inverse Ising





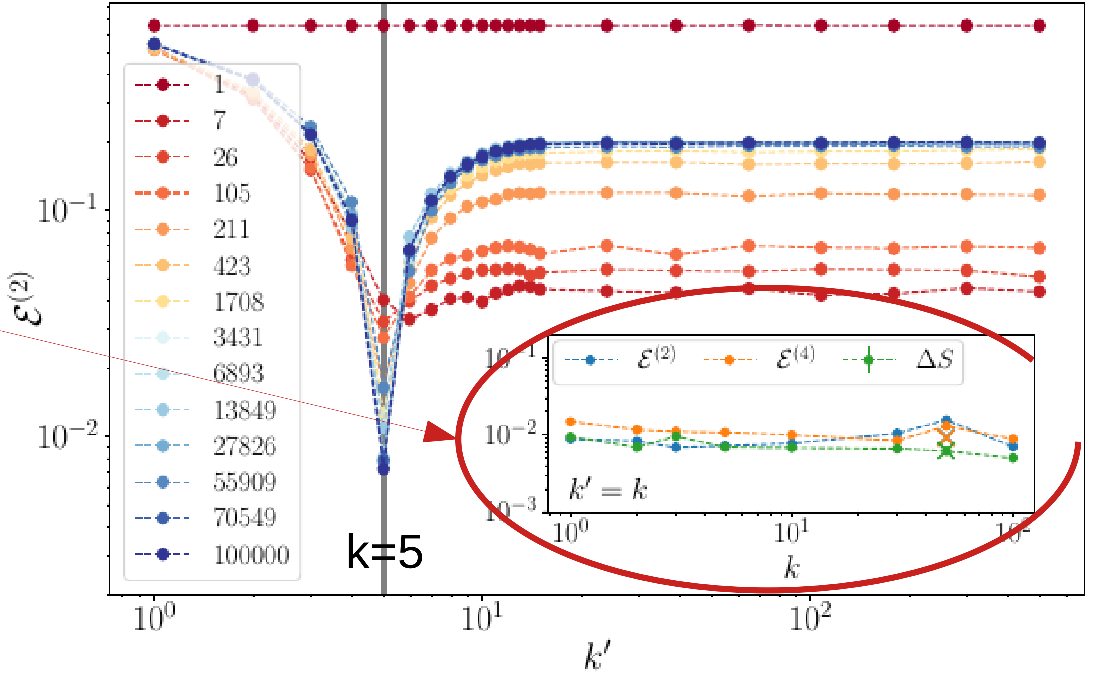


$$\langle S_i S_j \rangle_{p_{\text{data}}} = \langle S_i S_j \rangle_{p(k, \boldsymbol{p}_0)}$$

$$\langle S_i \rangle_{p_{\text{data}}} = \langle S_i \rangle_{p(k, \boldsymbol{p}_0)}$$

$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\text{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k, \boldsymbol{p}_0)}$$

2) Fast generators : Inverse Ising



Even at k=1 !

$$\langle S_i S_j \rangle_{p_{\text{data}}} = \langle S_i S_j \rangle_{p(k,\boldsymbol{p}_0)}$$

$$\langle S_i \rangle_{p_{\text{data}}} = \langle S_i \rangle_{p(k,\boldsymbol{p}_0)}$$

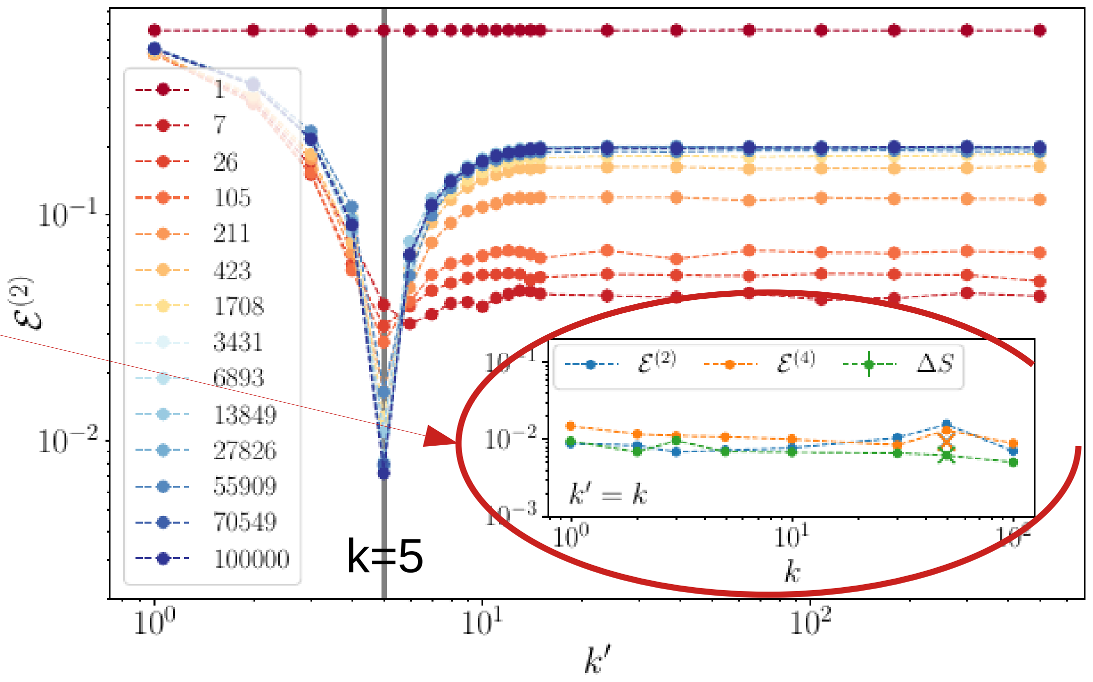
$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\text{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$$

2) Fast generators : Inverse Ising

Describe these curves analytically at high temperatures or in the Gaussian model...

-Agoritsas, Catania, Decelle, Seoane ICML (2023)

Even at k=1 !

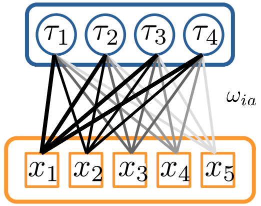


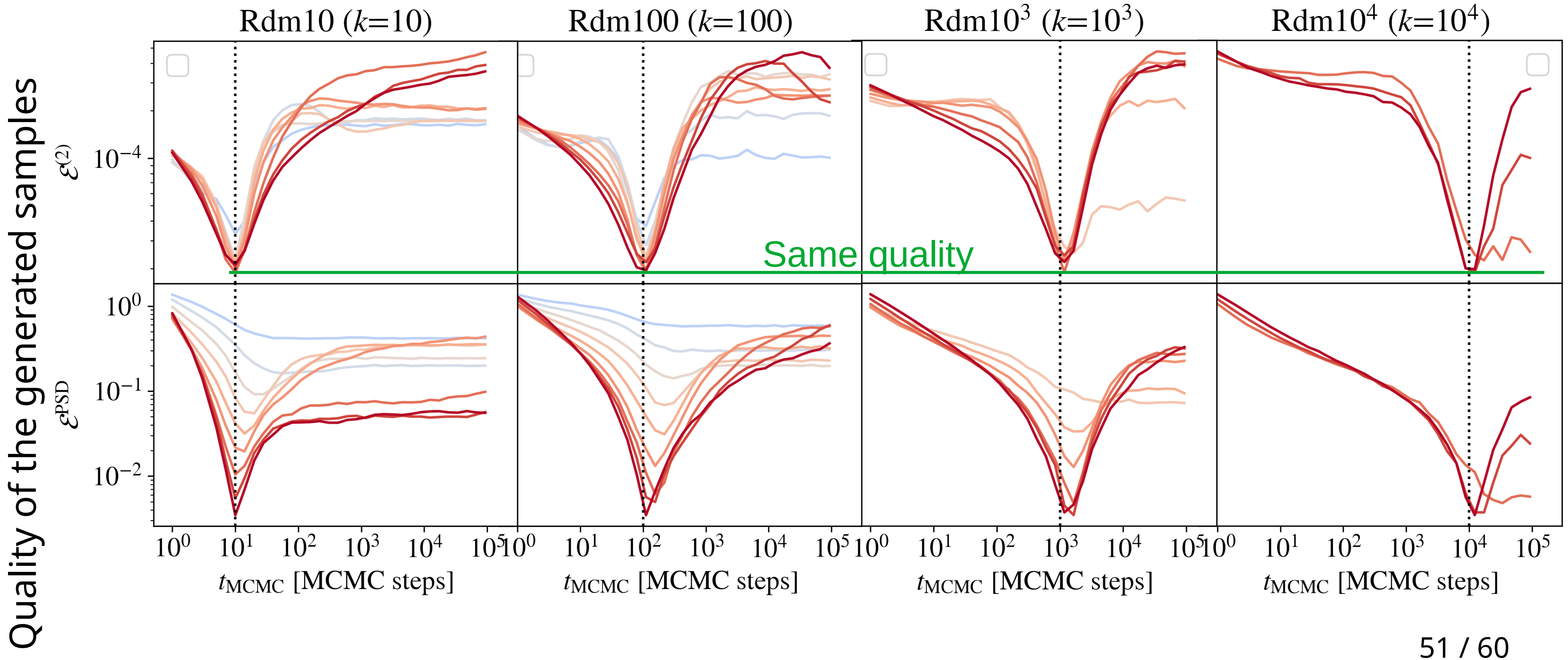
$$\langle S_i S_j \rangle_{p_{\mathrm{data}}} = \langle S_i S_j \rangle_{p(k,\boldsymbol{p}_0)}$$

$$\langle S_i \rangle_{p_{\mathrm{data}}} = \langle S_i \rangle_{p(k,\boldsymbol{p}_0)}$$

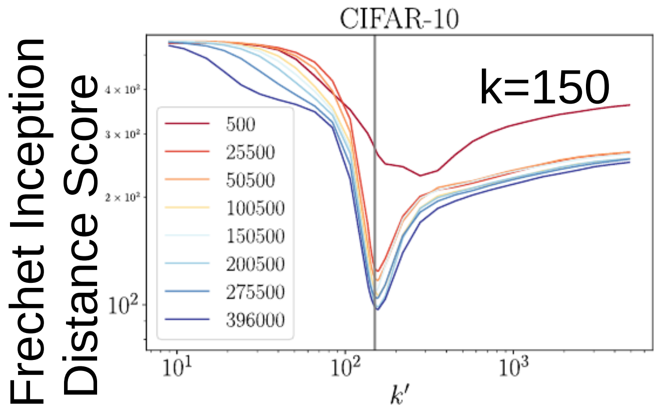
$$\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p_{\mathrm{data}}} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\star} \rangle_{p(k,\boldsymbol{p}_0)}$$

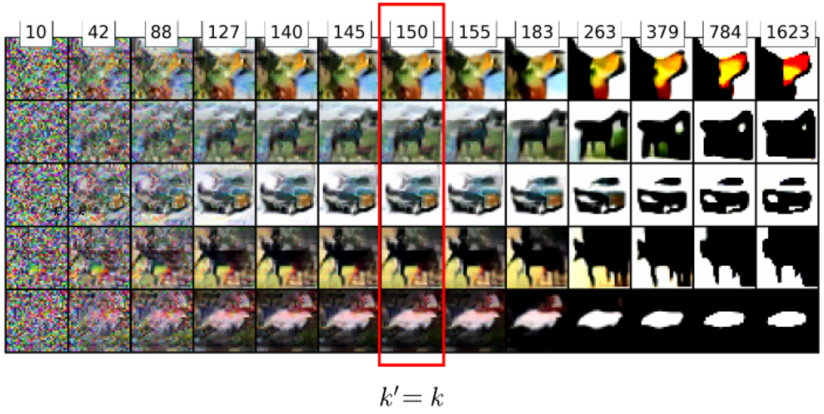
2) Fast generators : The RBM

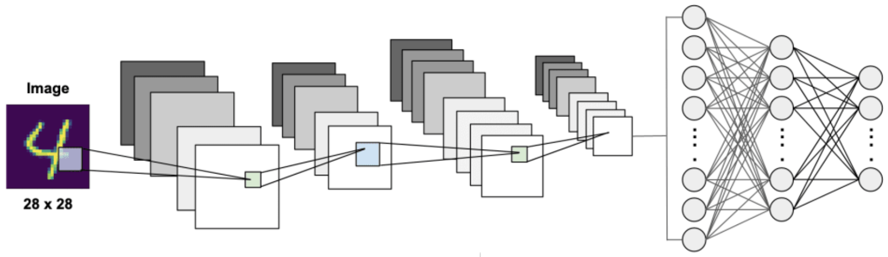




3) This effect is present in any EBM



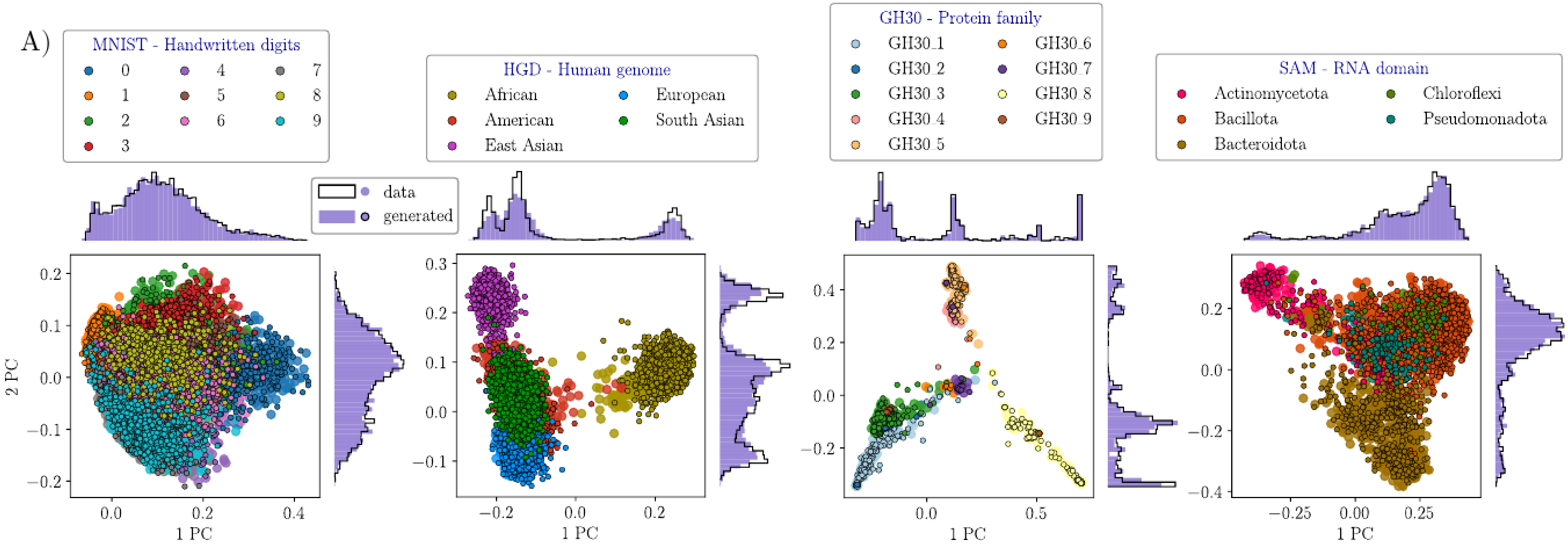




CIFAR-10

ConvNet EBM

4) and any dataset: Multimodal distributions



Fast generation of multimodal data : k=10

[Carbone, Decelle, Seoane, Rosset, arXiv:2307.06797]

Very similar to diffusion !

Sohl-Dickstein, Weiss, Maheswaranathan, Ganguli, PMLR (2015)

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

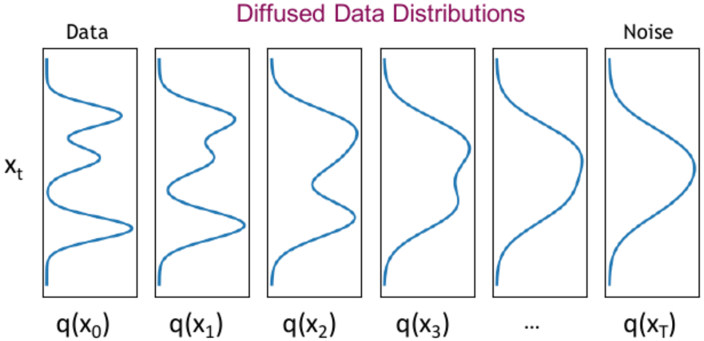
Encoder (Gaussian model)

Data

Noise

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$$

Decoder



- K. Kreis, R. Gao, and A. Vahdat. *Denoising diffusion-based generative modeling: foundations and applications.* CVPR Tutorial. 2022.
- H. Cao, C. Tan, Z. Gao, G. Chen, P.-A. Heng, and S. Z. Li. *"A Survey on Generative Diffusion Model".* ArXiv: 2209.02646 (2022).
- T. Karras, M. Aittala, T. Aila, and S. Laine. *"Elucidating the Design Space of Diffusion-Based Generative Models".* In: NeurIPS (2022)
- Gao, R., Song, Y., Poole, B., Wu, Y. N., & Kingma, D. P. ***Learning energy-based models by diffusion recovery likelihood.*** ICLR (2021)

Summary

- If the goal is to generate samples statistically similar to the data:

No point of trying to equilibrate (long training, long generation)

Go for out-of-equilibrium training !!

- But...

Summary

- If the goal is to generate samples statistically similar to the data:

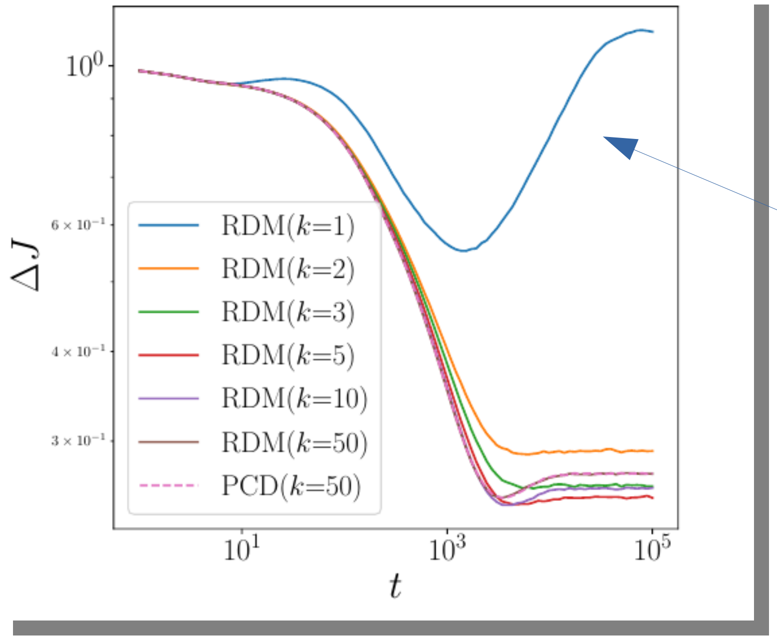
No point of trying to equilibrate (long training, long generation)

Go for out-of-equilibrium training !!

- But...

$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\sim \boldsymbol{p}_{\text{data}}(x)$$

Is this OOE energy function useful for interpretability ?

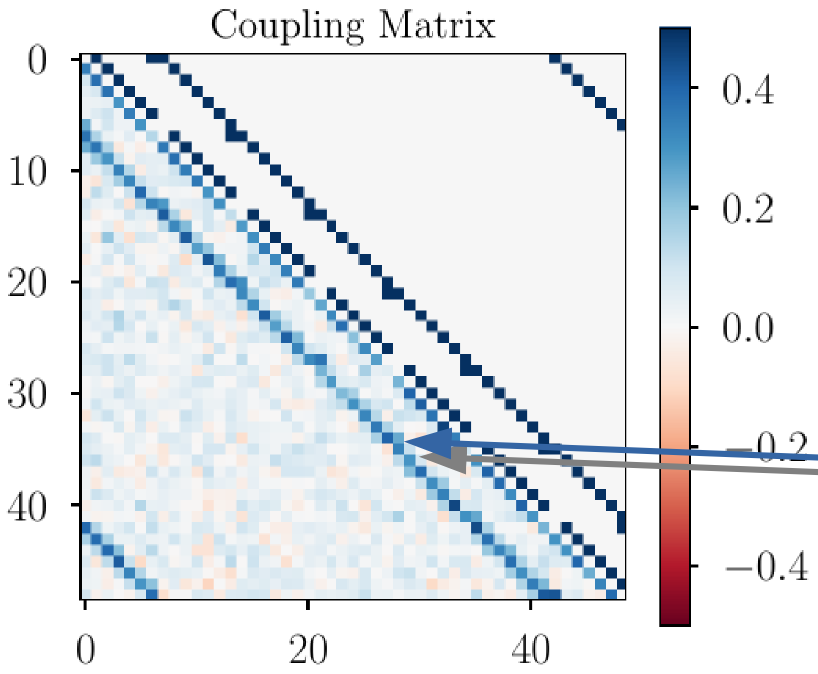


Inverse Ising case : $(J_{\text{learned}} - J_{\text{true}})^2$

Very bad model

$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\approx \boldsymbol{p}_{\text{data}}(x)$$

Is this OOE energy function useful for interpretability ?

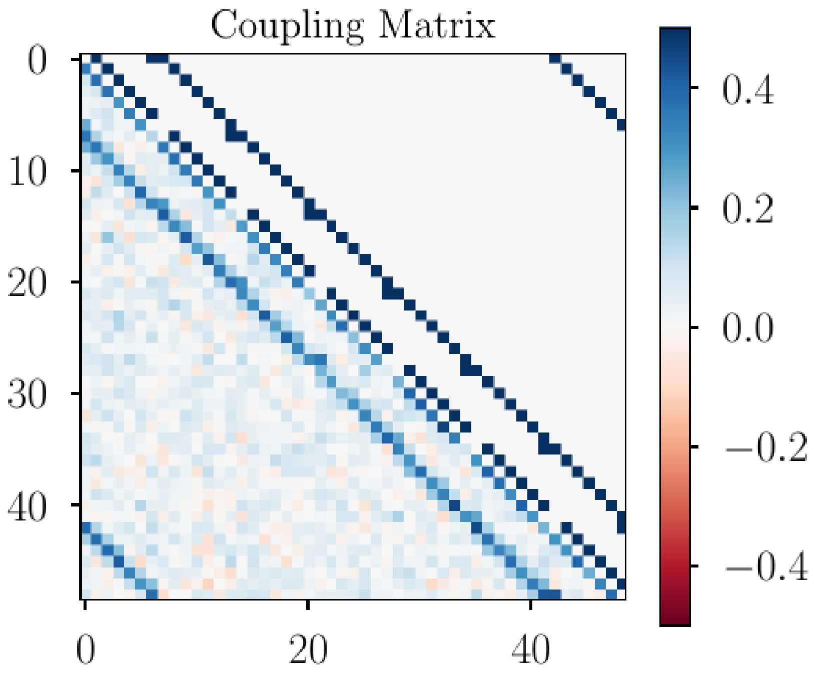


RBM Effective model : $(J_{learned} - J_{true})^2$

More interactions that what it should !

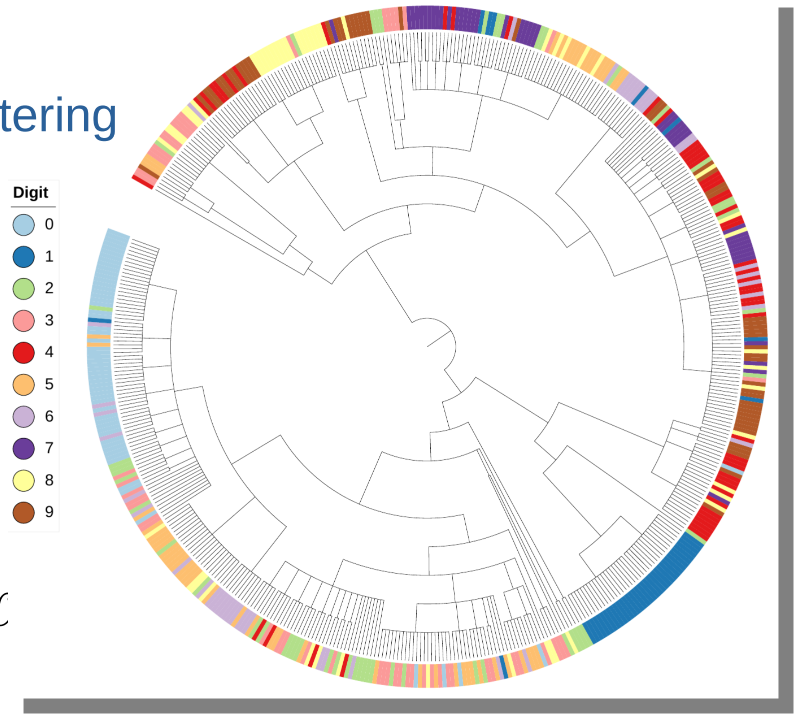
$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\sim \boldsymbol{p}_{\mathrm{data}}(x)$$

Is this OOE energy function useful for interpretability ?



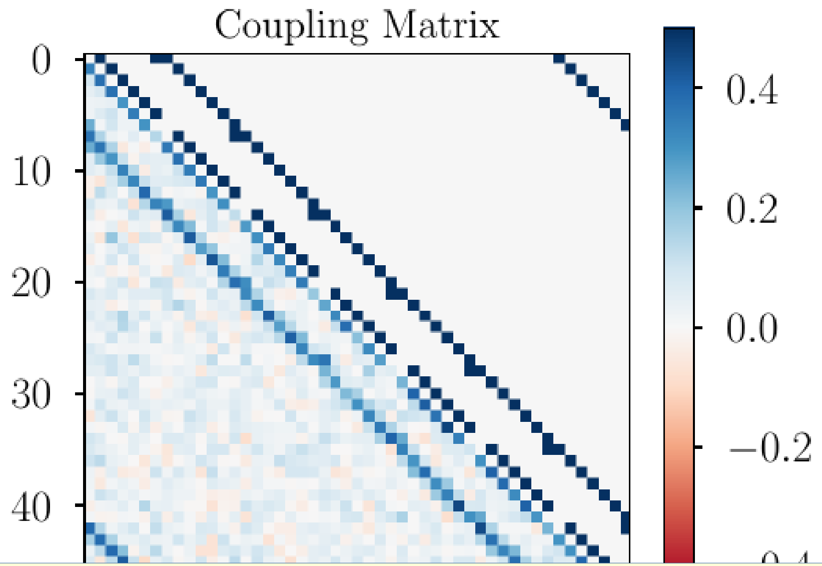
RBM Effective model : $(J_{learned} - J_{true})^2$

Poor clustering



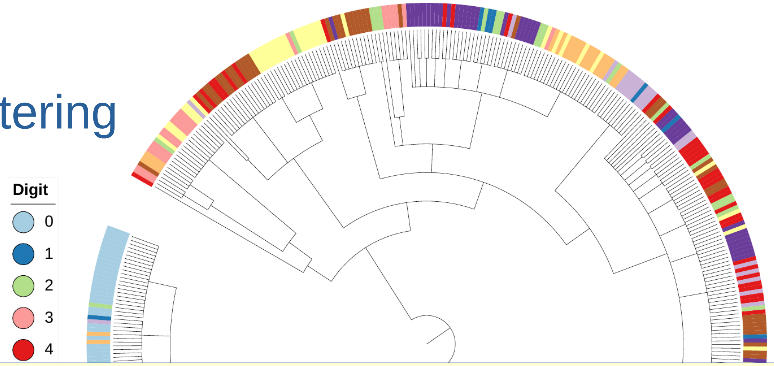
$$p_{\boldsymbol{\theta}^\star}(x) = \frac{e^{-E_{\boldsymbol{\theta}^\star}(x)}}{Z_{\boldsymbol{\theta}^\star}} \not\sim \boldsymbol{p}_{\mathrm{data}}(x$$

Is this OOE energy function useful for interpretability ?



RBM Effective model : $(J_{learned} - J_{true})^2$

Poor clustering



If we want to get good models for our data…

<u>we need to equilibrate the chains</u> all along training

Very challenging with multimodal data

The good news is that we know how to do it ! (arriving soon)