# Scalable and Reliable Inference for Probabilistic Modeling

Ruqi Zhang

UT Austin/Purdue

Joint work with Chris De Sa, A. Feder Cooper, Changyou Chen, Chunyuan Li, Andrew Wilson, Jianyi Zhang

# Many Areas are Revolutionized by Data

Society

Science

Technology

How to learn from big and complex data?

Polic

Computer vision

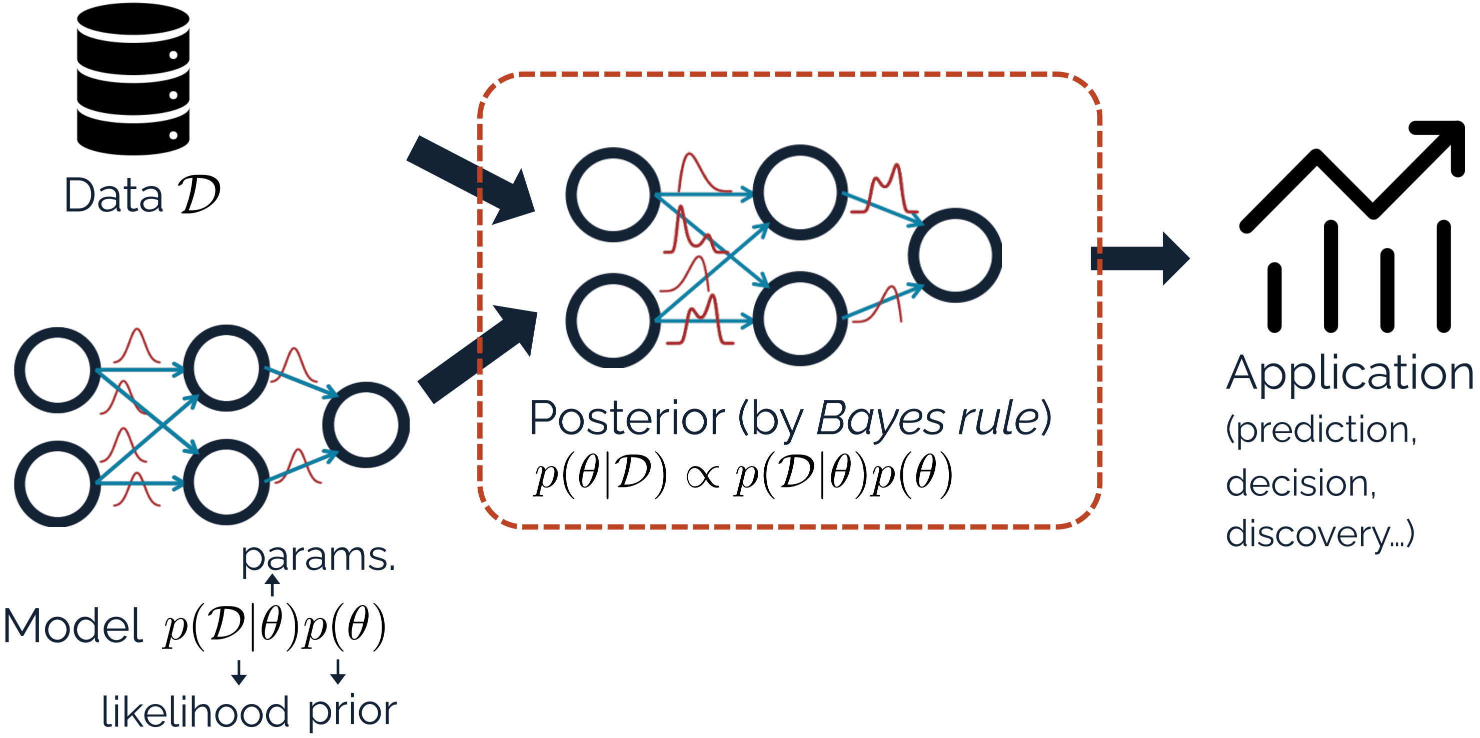
Economics analysis

Scientific discovery

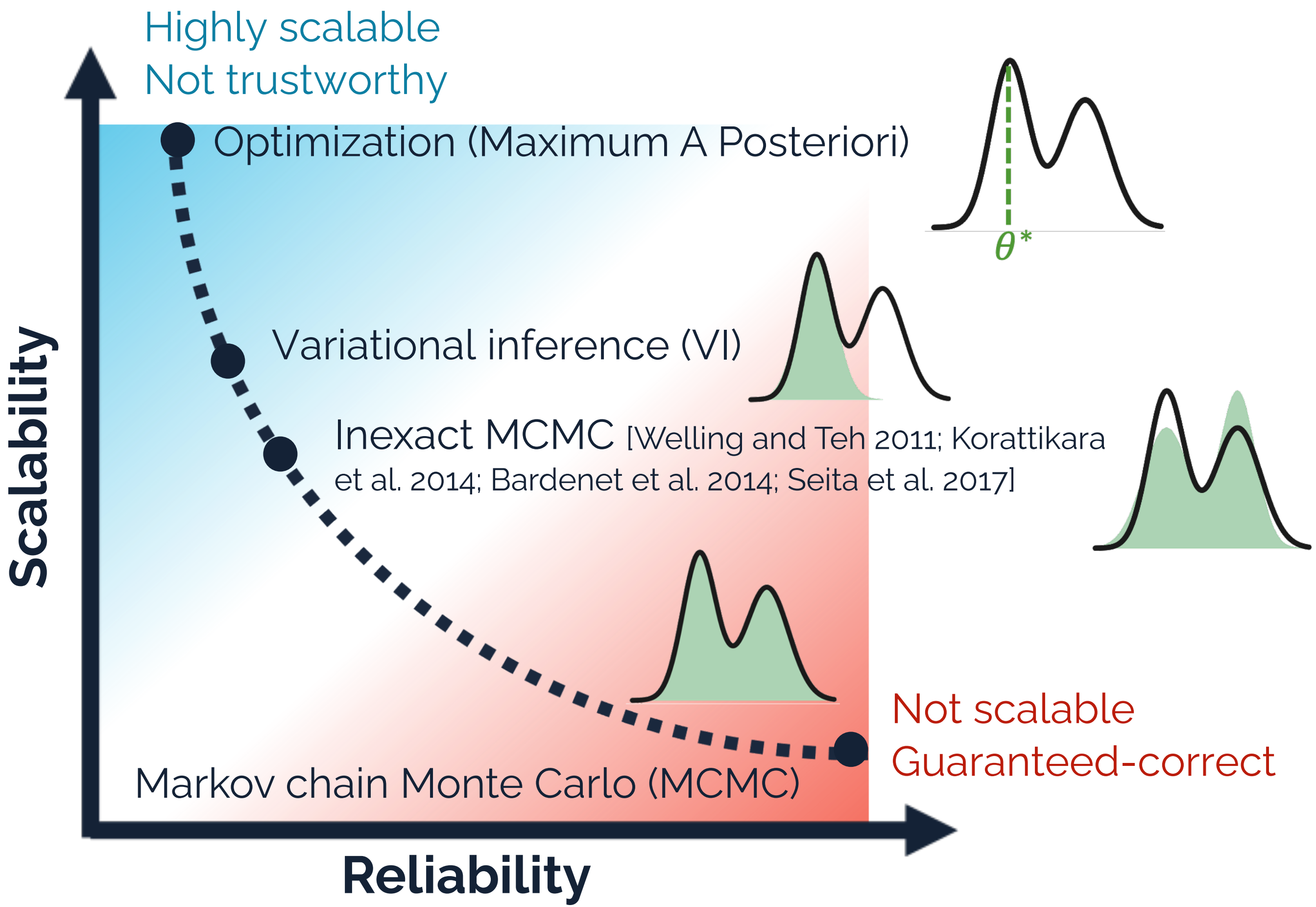
Speech recognition

# Probabilistic Modeling Pipeline

**Key algorithmic problem**: how to infer parameters from data?



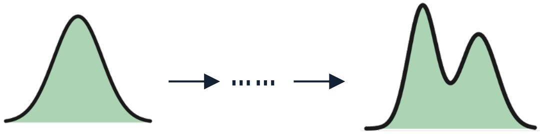
# Trade-Off in Current Inference Methods



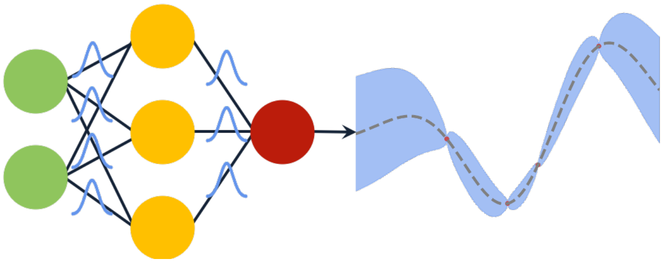
# Scalable and Reliable Inference

## Theoretically-Guaranteed Inference

minibatch ≈ dataset



## Efficient Inference for Reliable Deep Learning



TunaMH. Zhang et al. NeurIPS'20. Spotlight

Poisson-Gibbs. Zhang et al. NeurIPS'19. Spotlight

AMAGOLD. Zhang et al. AISTATS'20

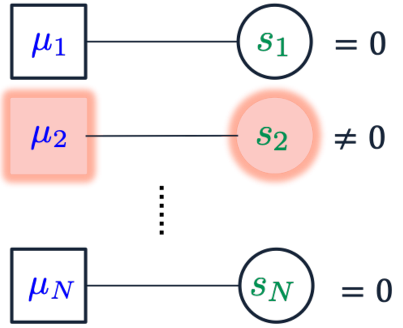
cSG-MCMC. Zhang et al. ICLR'20. Oral

Meta-VI. Zhang et al. AISTATS'21

# Talk Outline

## Poisson-Minibatching

A general framework to make inference scalable and reliable

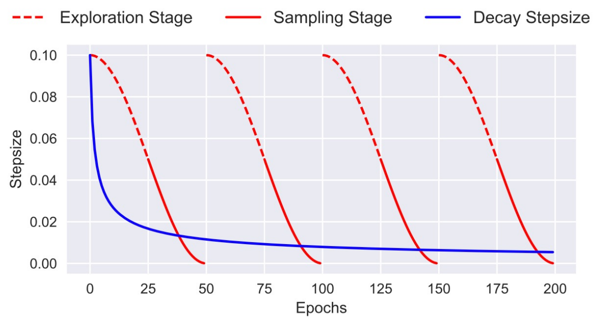



TunaMH. Zhang et al. NeurIPS'20. Spotlight

Poisson-Gibbs. Zhang et al. NeurIPS'19. Spotlight

## Cyclical SG-MCMC

An efficient MCMC for inference in deep learning




cSG-MCMC. Zhang et al. ICLR'20. Oral

# Metropolis–Hastings (MH)

- One of the most fundamental inference methods (Metropolis et al. 1953, Hastings 1970)
- One of top ten most influential algorithms

from *SIAM News*, Volume 33, Number 4

**The Best of the 20th Century: Editors Name Top 10 Algorithms**

*By Barry A. Cipra*

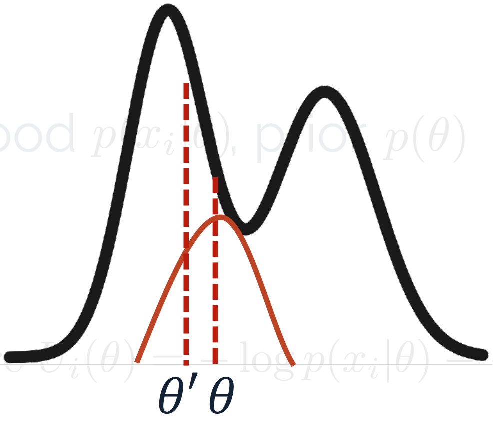
- Workhorse for many other MCMC methods

# Metropolis–Hastings (MH)

Given: dataset $\{x_i\}_{i=1}^N$, model: likelihood $p(x_i|\theta)$, prior $p(\theta)$

Goal: estimate the posterior

$$p(\theta|\{x_i\}_{i=1}^N) \propto \exp\left(-\sum_{i=1}^N U_i(\theta)\right), \text{ where } U_i(\theta) = -\log p(x_i|\theta) - \frac{1}{N}\log p(\theta)$$

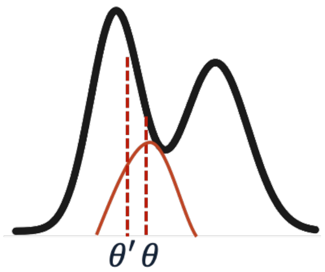



Algorithm

- Generate a proposal $\theta' \sim q(\theta'|\theta)$
- Accept it with probability $a(\theta,\theta') = \min\left(1, \exp\left(\sum_{i=1}^N (U_i(\theta) - U_i(\theta'))\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$

sum over entire dataset

# Metropolis-Hastings (MH)




Given: dataset $\{x_i\}_{i=1}^N$, model: likelihood $p(x_i|\theta)$, prior $p(\theta)$

Goal: estimate the posterior

$$p(\theta|\{x_i\}_{i=1}^N) \propto \exp\left(-\sum_{i=1}^N U_i(\theta)\right), \text{where } U_i(\theta) = -\log p(x_i|\theta) - \frac{1}{N}\log p(\theta)$$

Algorithm

Approximate by a minibatch

- Generate a proposal $\theta' \sim q(\theta'|\theta)$
- Accept it with probability $a(\theta,\theta') = \min\left(1, \exp\left(\sum_{i=1}^N (U_i(\theta) - U_i(\theta'))\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$

Challenge: the accept/reject step is costly when dataset is large!

# Minibatch to scale MH

Inexact methods

- Pros: mild assumptions
- Cons: asymptotic bias

[Korattikara et al. 2014; Bardenet et al. 2014; Seita et al. 2017...... ]

NO ← **Preserve Posterior?** → YES

Exact methods

- Pros: no bias
- Cons: strong assumptions, low scalability

[Maclaurin et al. 2015; Cornish et al. 2019; Zhang et al. 2019 ...... ]

Which to use?
Better trade-off?

# Q: Is it important to be exact?
# A: Yes. Inexact methods are unreliable

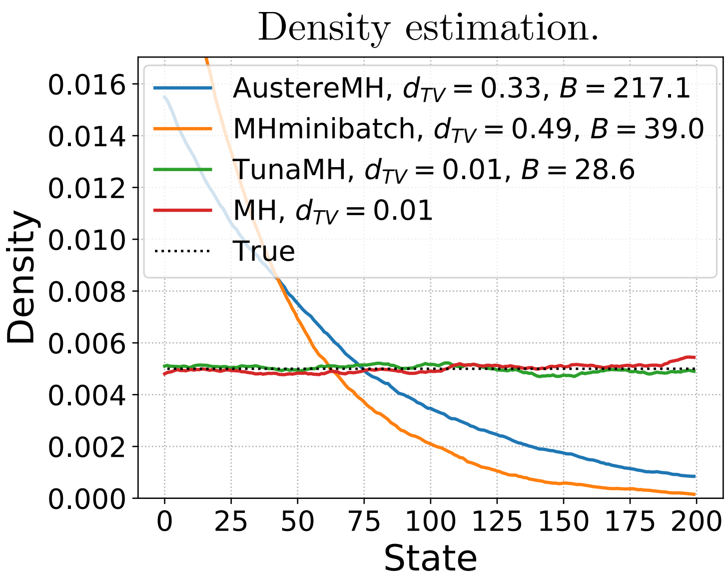
**Theorem** (informal): *the stationary distribution of any inexact method can be arbitrarily far from the posterior (in terms of total variation distance and KL divergence)*

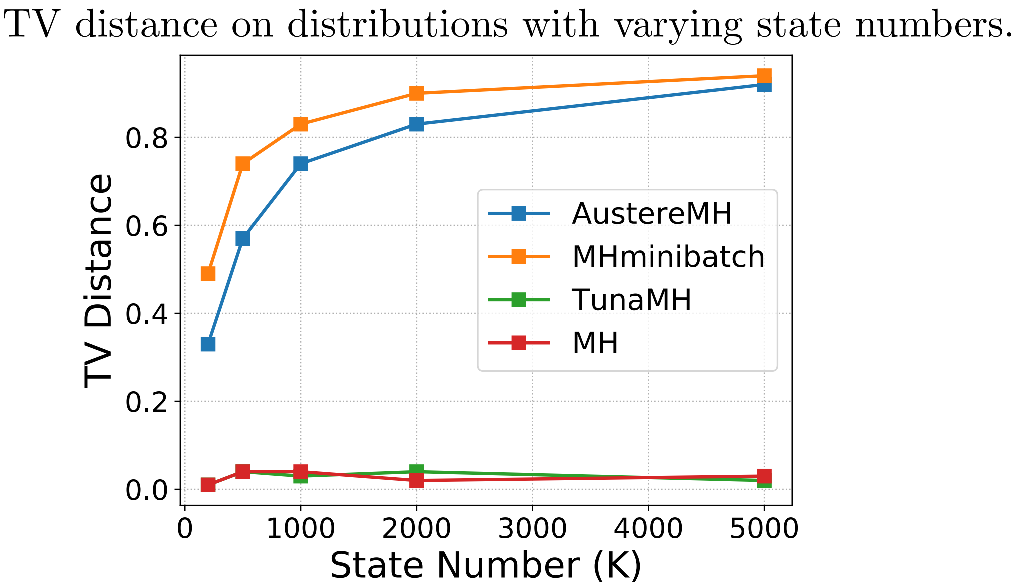
**Takeaway**

- Any inexact minibatch MH can be arbitrarily wrong
- We should use exact methods

Asymptotically Optimal Exact Minibatch Metropolis-Hastings, R. Zhang, A. Cooper, C. De Sa, NeurIPS'20. Spotlight

# Empirical Verification



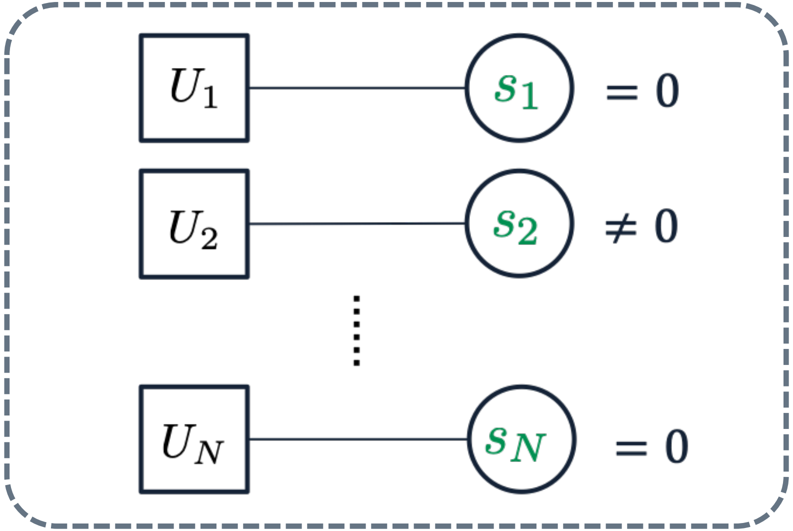




- The stationary distributions of inexact methods (AustereMH and MHminibatch) diverge significantly from the true distribution
- Divergence can be arbitrarily large

# Q: How to make exact methods scalable?
# A: Poisson-Minibatching

Acceptance probability

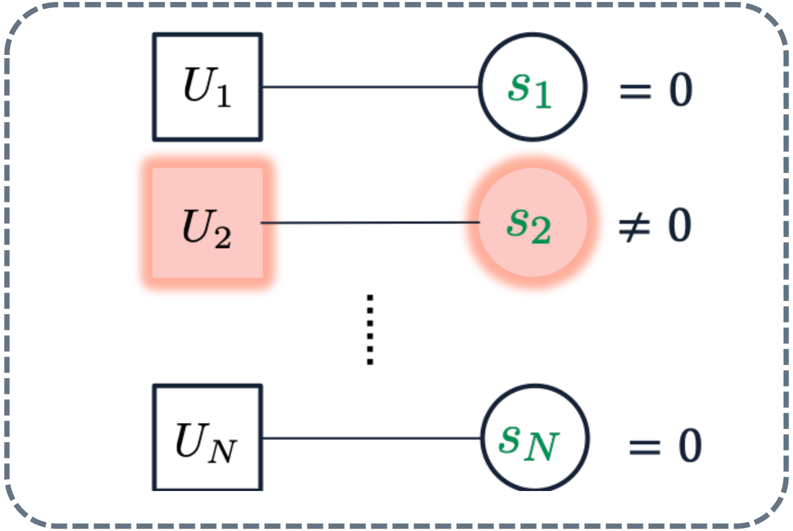
$$a(\theta, \theta') = \min\left(1, \exp\left(\sum_{i=1}^{N} s_i (U_i(\theta) - U_i(\theta'))\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$$



# Q: How to make exact methods scalable?
# A: Poisson-Minibatching

Acceptance probability

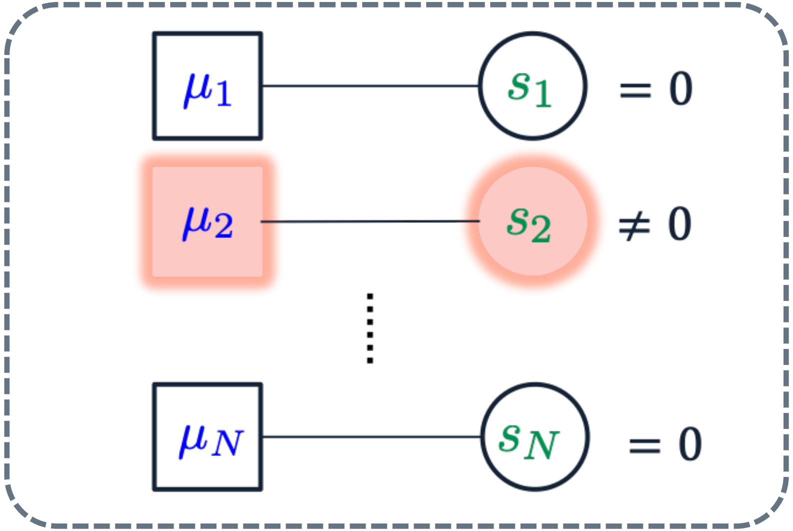
$$a(\theta, \theta') = \min\left(1, \exp\left(\sum_{i \in \{j | s_j \neq 0\}} s_i (U_i(\theta) - U_i(\theta'))\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$$



# Q: How to make exact methods scalable?
# A: Poisson-Minibatching

Acceptance probability

$$a(\theta, \theta') = \min\left(1, \exp\left(\sum_{i \in \{j \mid s_j \neq 0\}} s_i \mu_i(\theta, \theta')\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$$



# Guaranteed Exactness

$$a(\theta, \theta') = \min\left(1, \exp\left(\sum_{i \in \{j | s_j \neq 0\}} s_i \mu_i(\theta, \theta')\right) \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}\right)$$

- How to sample $\{s_i\}_{i=1}^N$ quickly?
- Poisson variables! $B = \sum_{i=1}^N s_i \sim \text{Pois}(\Lambda),\ \{s_i\}_{i=1}^N \sim \text{Multinomial}(B, \{p_i\}_{i=1}^N)$
  $s_i \sim \text{Pois}(\lambda_i(\theta, \theta'))$

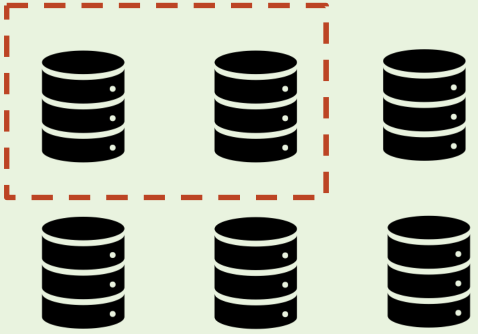
- How to define $\lambda_i(\theta, \theta')$ and $\mu_i(\theta, \theta')$ ?
- Define to ensure exactness (detailed balance): $\pi(\theta)T(\theta, \theta') = \pi(\theta')T(\theta', \theta)$

- We call this algorithm *TunaMH*

# Guaranteed Scalability

Overall cost = cost per step × # of steps

Batch size ≪ $N$



**Theorem** (informal): *TunaMH is at most a constant factor slower than standard MH*

The first such bound for minibatch MH

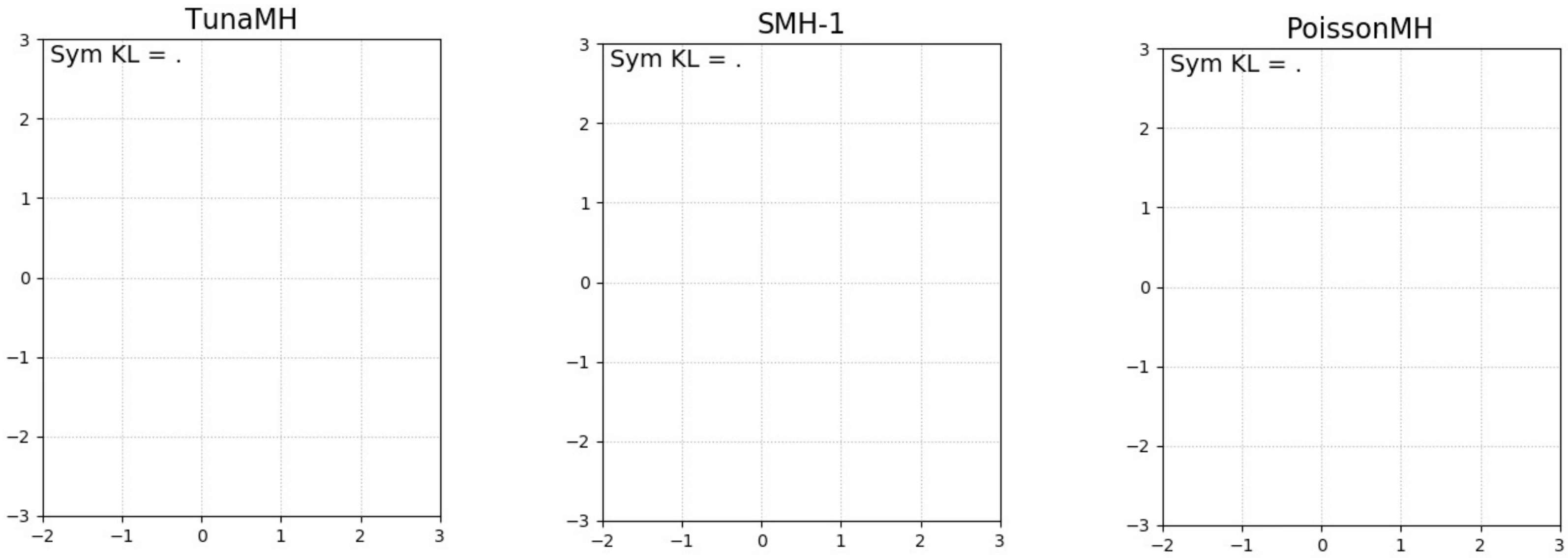
# Q: Is it possible to develop a better exact method?
# A: No. TunaMH is asymptotically optimal

**Theorem** (informal): *given a target convergence rate, we prove a lower bound on the required batch size for any exact minibatch MH*

**Takeaway**

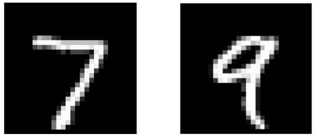
- The first theorem to provide a ceiling for the performance of exact minibatch MH
- TunaMH is asymptotically optimal in the batch size

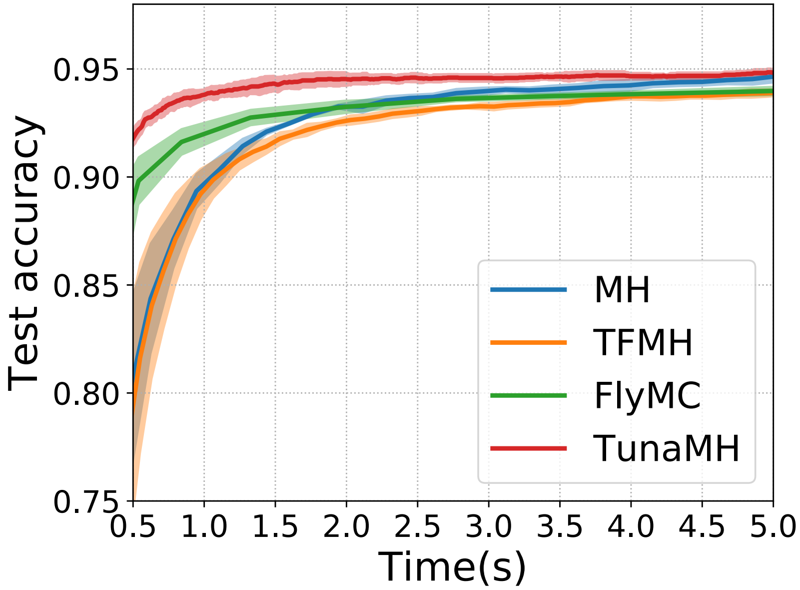
# Gaussian Mixture




- Compared to SOTA exact methods, TunaMH is the fastest to converge

# Logistic Regression on MNIST





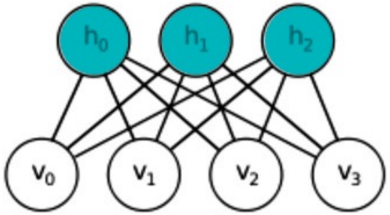

- TunaMH has the highest test accuracy given time

# What about other inference methods?

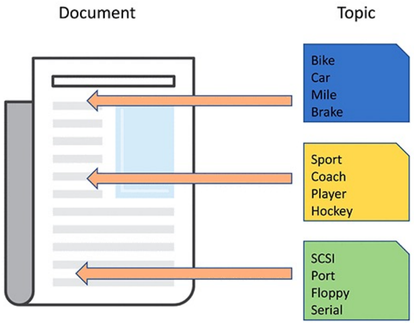
**Poisson-minibatching offers a general recipe for scalable exact inference**

# Gibbs sampling (Geman et al. 1984)

- De facto inference method for graphical models
- Used in many applications

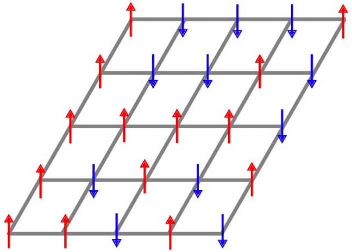



Restricted Boltzmann machine (RBM)




Topic modeling

snorkel



Physical modeling

DeepDive

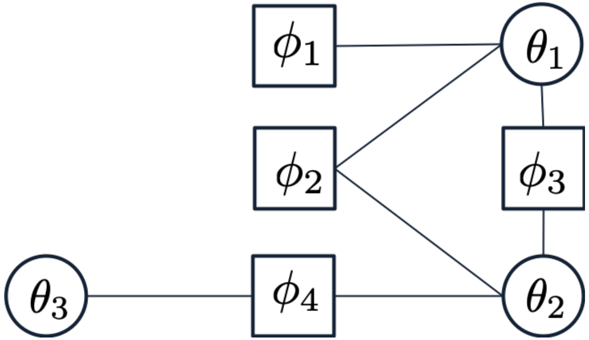
# Inference on Graphical Models

- Consider factor graphs

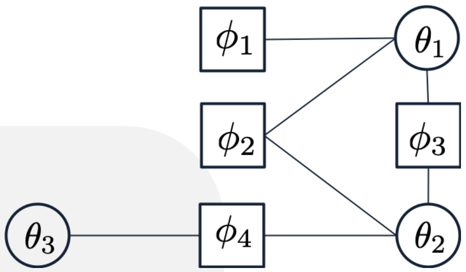
$$\pi(\theta_{1:d}) \propto \exp\left(\sum_{i=1}^{N} \phi_i(\theta_{1:d})\right)$$

$$A[j] = \{i | \phi_i \text{ depends on variable } j\}$$

e.g. $A[1] = \{1, 2, 3\}$



# Gibbs sampling




**Loop**

1. Select a variable $\theta_j$ to sample at random
2. Compute the conditional distribution of $\theta_j$

$$\rho \propto \exp\left(\sum_{i \in A[j]} \phi_i(\theta_{1:d})\right)$$

3. Resample variable $\theta_j$ from the conditional distribution

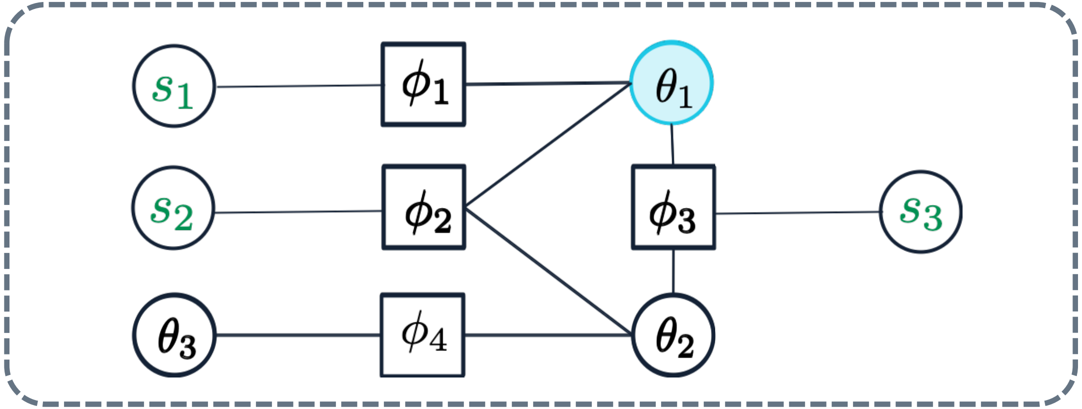
**End loop**

Expensive proposal distribution!

- Very expensive when the factor set is large!
- Can we subsample factors to compute conditional distributions?

# Poisson-Minibatching for Gibbs Sampling

$$\rho \propto \exp\left(\sum_{i \in A[j]} s_i \phi_i(\theta_{1:d})\right)$$

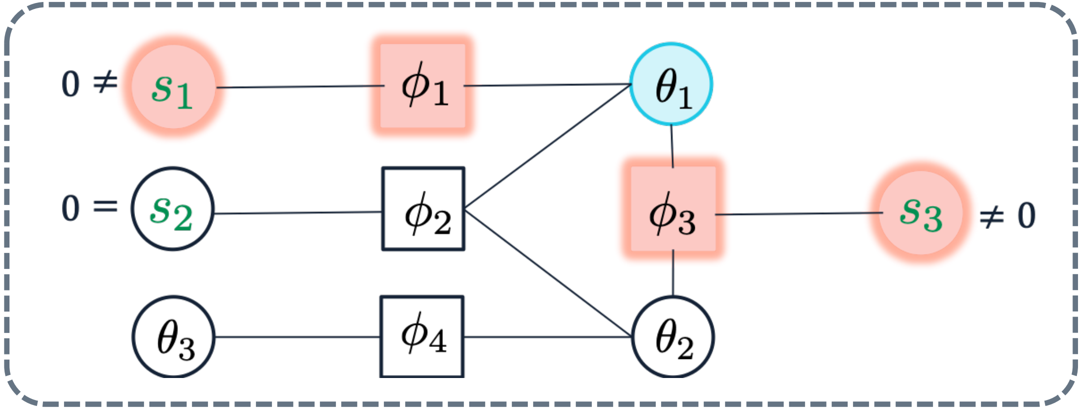



*Poisson-Gibbs*: guaranteed exactness and scalability

Poisson-Minibatching for Gibbs Sampling with Convergence Rate Guarantees, R. Zhang and C. De Sa, NeurIPS'19. Spotlight

# Poisson-Minibatching for Gibbs Sampling

$$\rho \propto \exp\left(\sum_{i \in \{k \mid s_k \neq 0, k \in A[j]\}} s_i \phi_i(\theta_{1:d})\right)$$

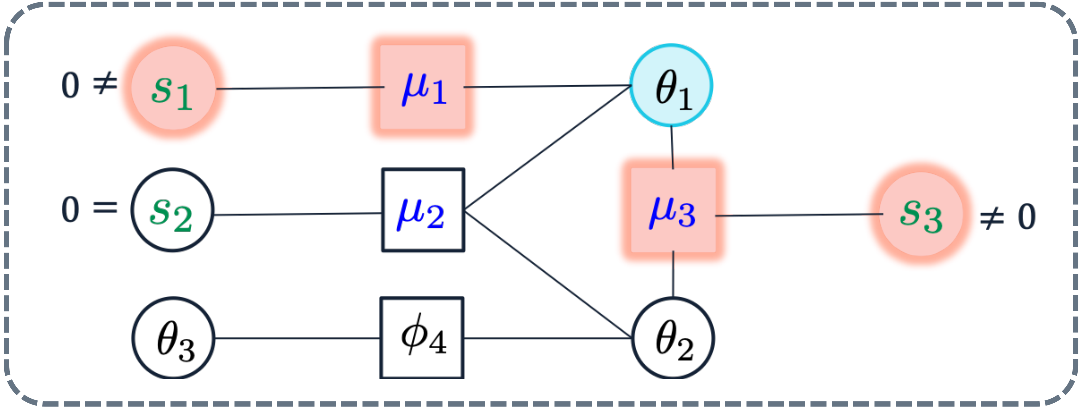



*Poisson-Gibbs*: guaranteed exactness and scalability

Poisson-Minibatching for Gibbs Sampling with Convergence Rate Guarantees, R. Zhang and C. De Sa, NeurIPS'19. Spotlight

# Poisson-Minibatching for Gibbs Sampling

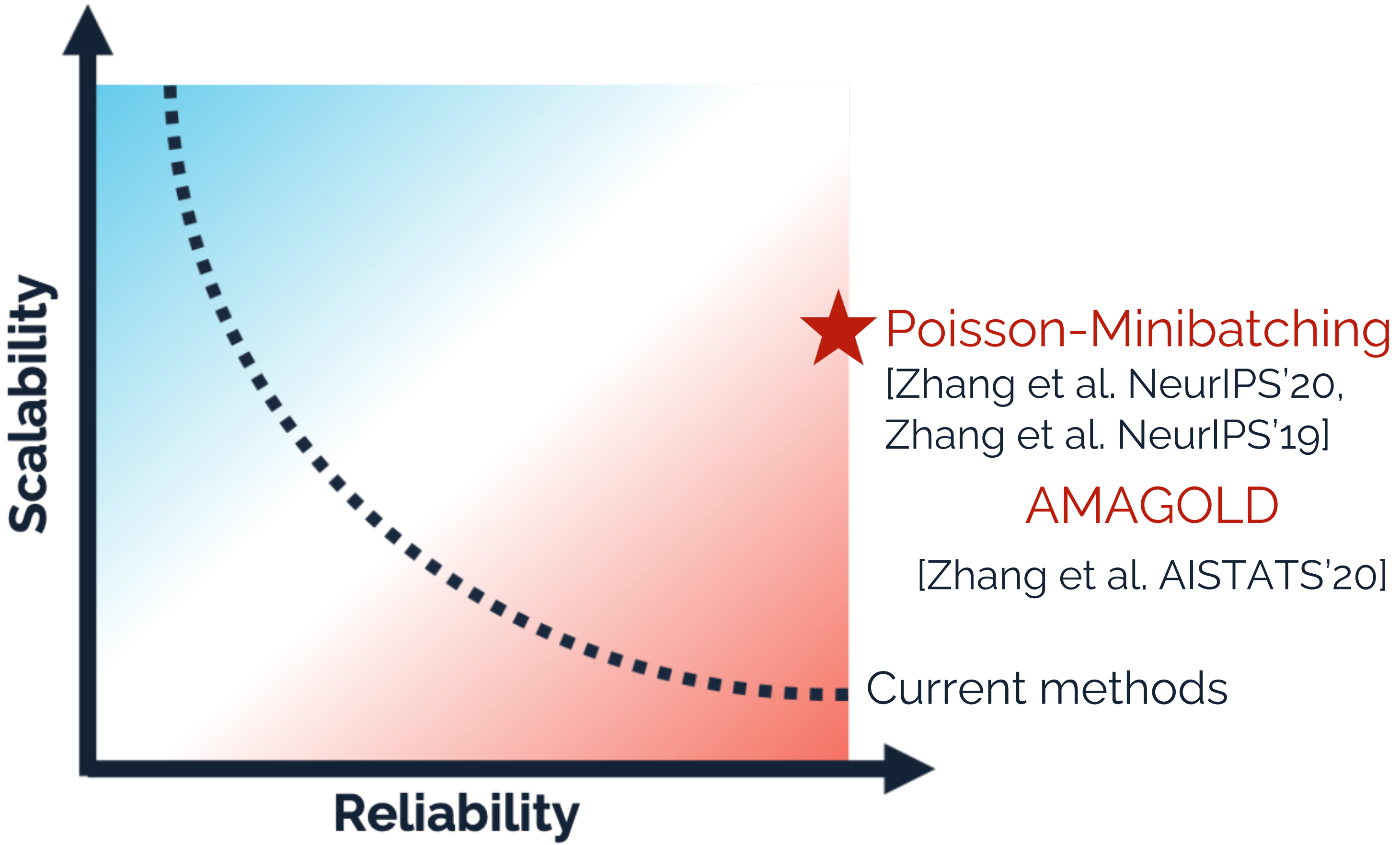
$$\rho \propto \exp\left( \sum_{i \in \{k \mid s_k \neq 0, k \in A[j]\}} s_i \mu_i(\theta_{1:d}) \right)$$




*Poisson-Gibbs*: guaranteed exactness and scalability

Poisson-Minibatching for Gibbs Sampling with Convergence Rate Guarantees, R. Zhang and C. De Sa, NeurIPS'19. Spotlight

# Theoretically-Guaranteed Inference



# Talk Outline

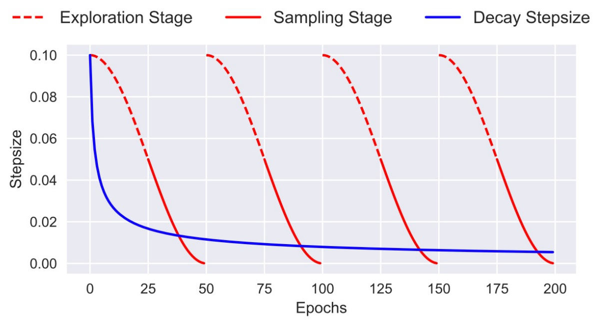
## Poisson-Minibatching

A general framework to make inference scalable and reliable




## Cyclical SG-MCMC

An efficient MCMC for inference in deep learning

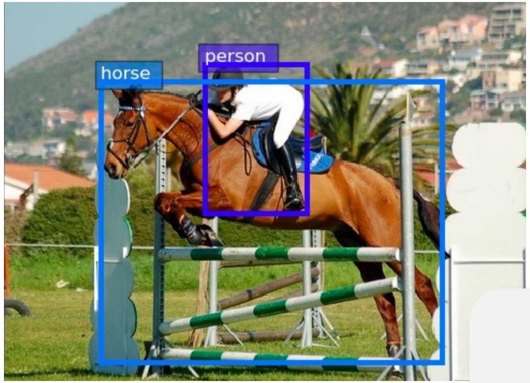



TunaMH. Zhang et al. NeurIPS'20. Spotlight

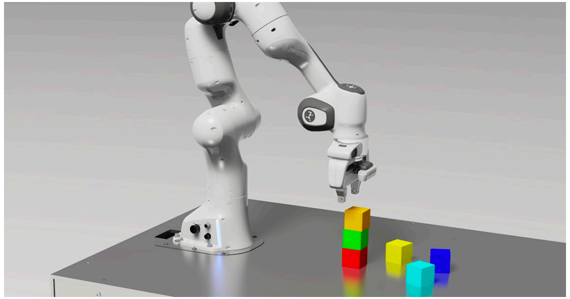
Poisson-Gibbs. Zhang et al. NeurIPS'19. Spotlight

cSG-MCMC. Zhang et al. ICLR'20. Oral

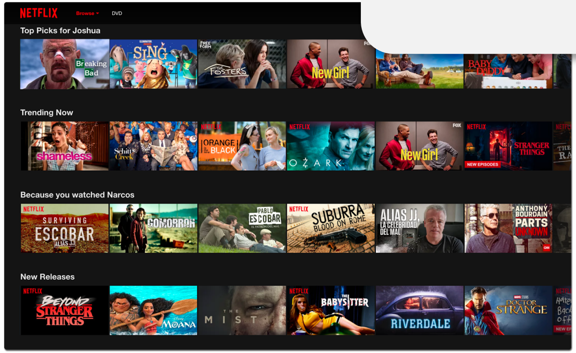
# Deep Learning







But…is it reliable?




# Question 1

Imagine that you travel to Seattle and want to know more about this city. Where will you go?

# Question 1

Imagine that you travel to Seattle and want to know more about this city. Where will you go?

# Question 1

Imagine that you travel to Seattle and want to know more about this city. Where will you go?

# Question 1

Imagine that you travel to Seattle and want to know more about this city. Where will you go?

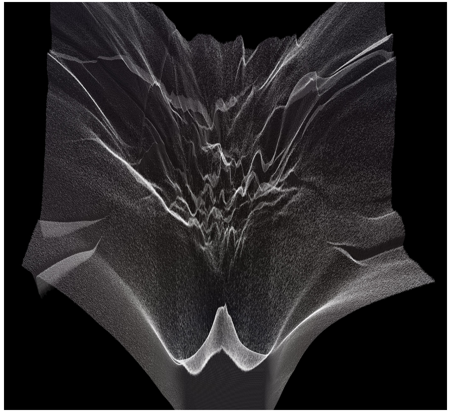
# Question 1

Imagine that you travel to Seattle and want to know more about this city. Where will you go?


**Answer: explore as many places as you can**

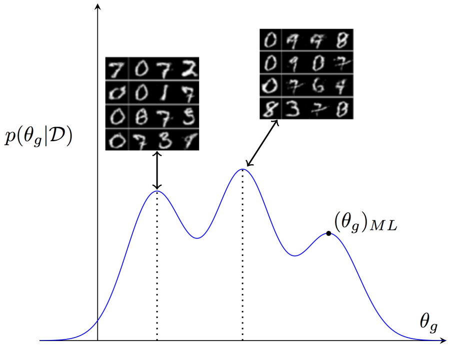
# Why Deep Learning Needs Reliable Inference

Posterior is complex and multimodal

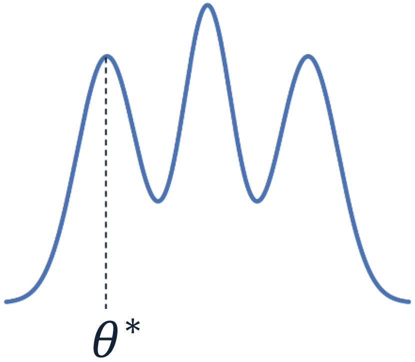


Loss surface in deep learning (credit: losslandscape.com)

Modes provide complementary explanations of data




[Saatchi and Wilson, 2017]



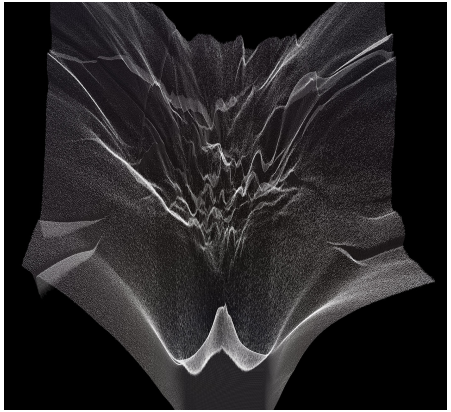

Overconfident

Easily fooled

Cyclical stochastic gradient MCMC for Bayesian deep learning, R. Zhang, C. Li, J. Zhang, C. Chen, A.G.Wilson, ICLR'20. Oral

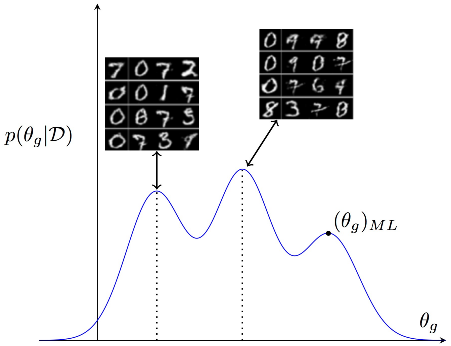
# Why Deep Learning Needs Reliable Inference

Posterior is complex and multimodal

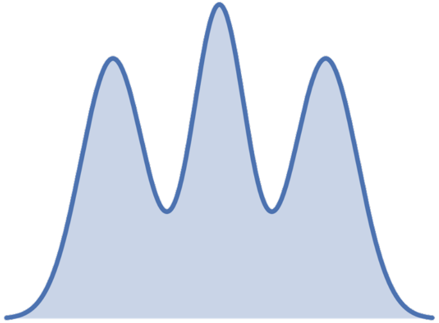


Loss surface in deep learning
(credit: losslandscape.com)

Modes provide complementary explanations of data




[Saatchi and Wilson, 2017]



$p(\theta|\mathcal{D})$

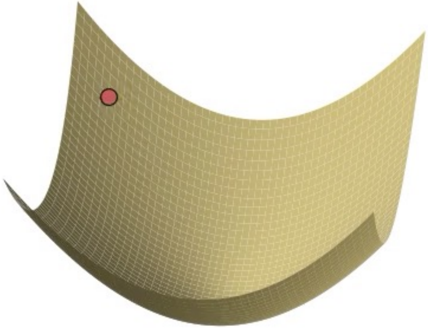
How to get it?

Cyclical stochastic gradient MCMC for Bayesian deep learning, R. Zhang, C. Li, J. Zhang, C. Chen, A.G.Wilson, ICLR'20. Oral

# Stochastic gradient MCMC

Stochastic gradient decent (SGD)

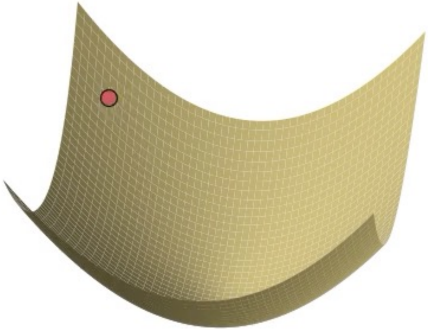
$$\theta_{k+1} = \theta_k - \alpha_k \nabla \tilde{U}(\theta_k)$$



Stochastic gradient Markov chain Monte Carlo (SG-MCMC) [Welling and Teh, 2011]

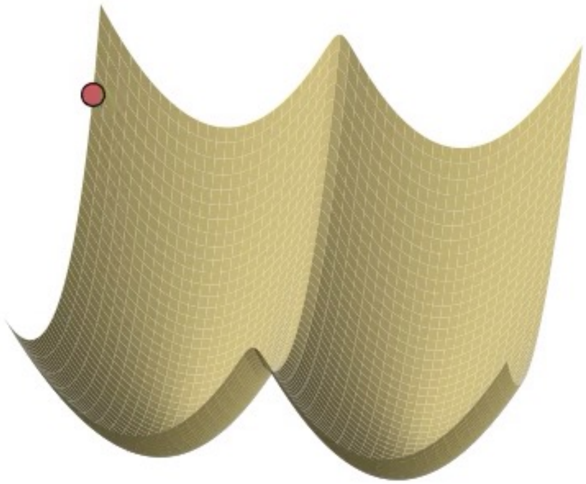
$$\theta_{k+1} = \theta_k - \alpha_k \nabla \tilde{U}(\theta_k) + \sqrt{2\alpha_k}\epsilon$$

where, $\epsilon \sim \mathcal{N}(0, I)$



# Improvements for SG-MCMC

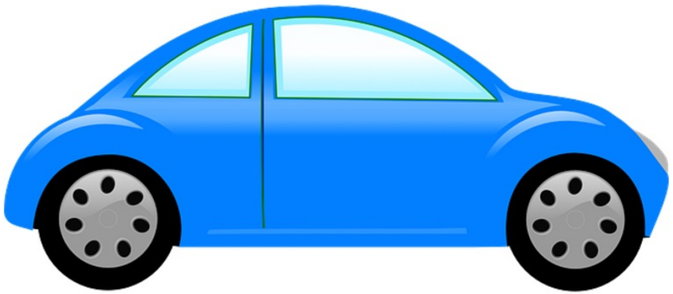
Introduce momentum variables [Chen et al.,2014], preconditioners [Ma et al. 2015, Li et al. 2016], variance reduction [Dubey et al. 2016, Baker et al. 2019]



Slow mixing: not efficient to explore multimodal distributions of DNNs

# Question 2

How do you efficiently explore the city? By car or on foot?



# Problem Analysis

Stepsize is the key!

- SG-MCMC requires a decaying stepsize to control error
- A small stepsize leads to slow mixing

Stepsize controls SG-MCMC's behavior in two ways:

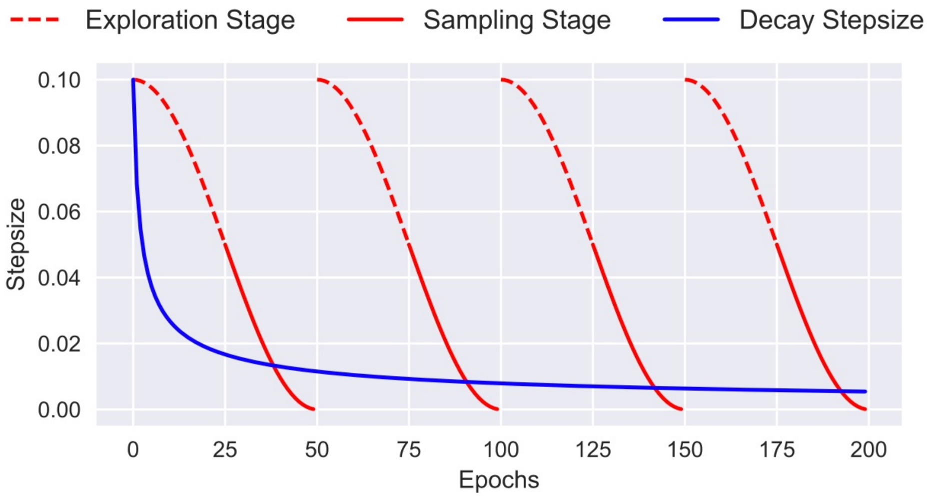
- magnitude to drift towards high density regions
- the level of injecting noise

$$\theta_{k+1} = \theta_k - \alpha_k \nabla \tilde{U}(\theta) + \sqrt{2\alpha_k}\epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, I)$$

A small stepsize reduces both abilities

# Our solution

## Cyclical stepsize schedule




Two stages of cSG-MCMC:

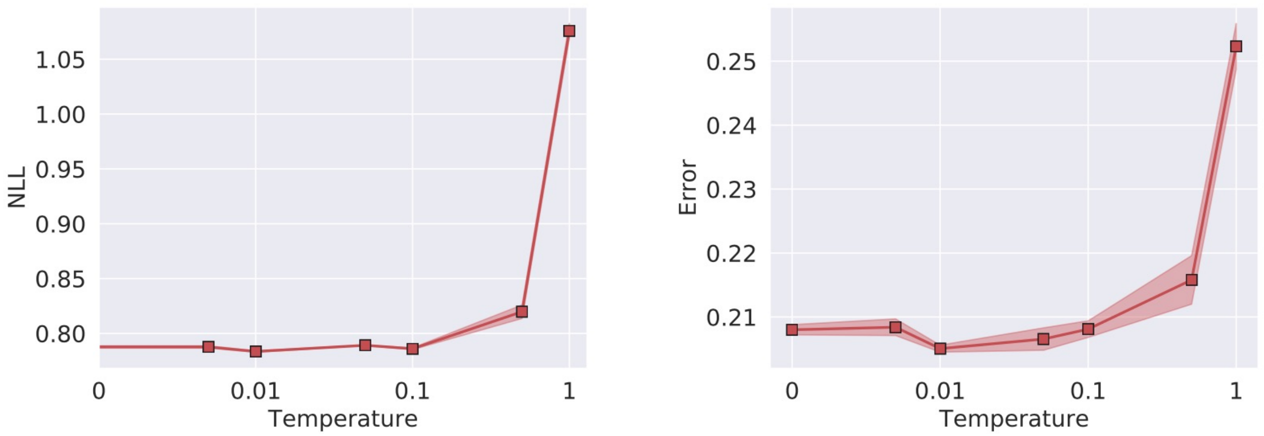
- Exploration: explore the parameter space with large stepsizes
- Sampling: characterize the fine-scale local density with small stepsizes

# cSG-MCMC Details

Introduce a system temperature $T$ to control the sampler's behavior

$$\theta_{k+1} = \theta_k - \alpha_k \nabla \tilde{U}(\theta) + \sqrt{2T\alpha_k}\epsilon$$

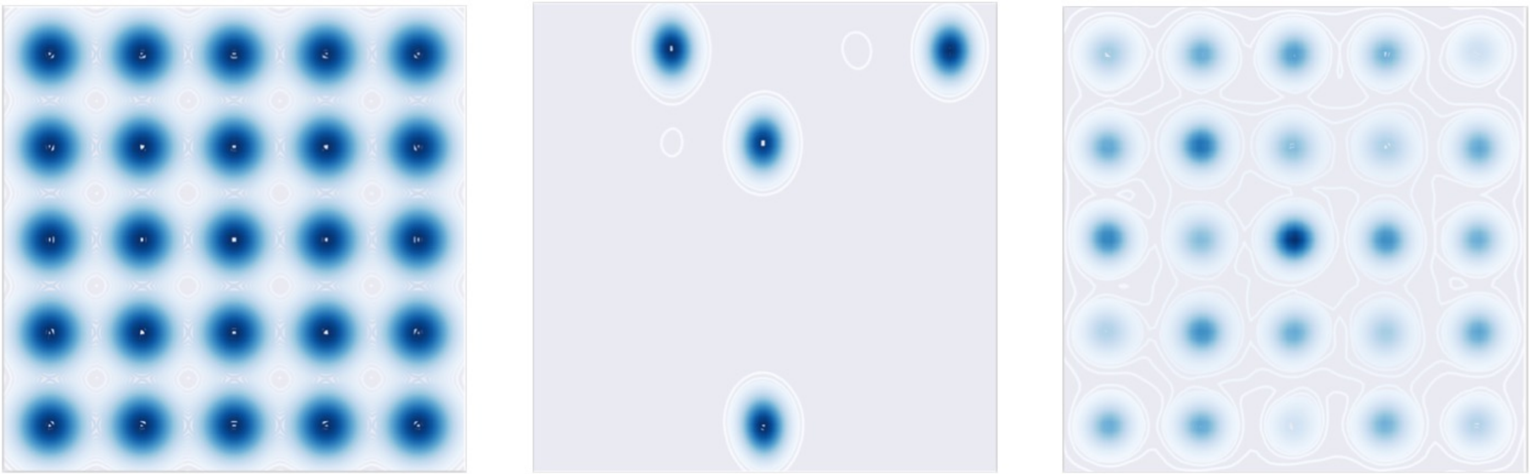
- Exploration: use $T = 0$ to converge quickly
- Sampling: use $0 < T < 1$ to improve performance



# Convergence Guarantees

**Theorem** (informal): *cSG-MCMC converges weakly and converges under the Wasserstein distance to the target distribution*

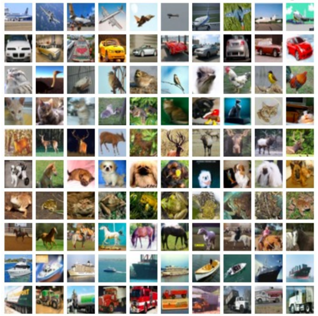
# Mixture of 25 Gaussians



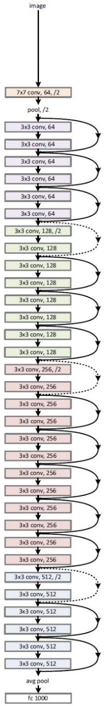
(a) Target (b) SGLD (c) cSGLD (ours)

- Whereas SGLD gets trapped in some local modes, cSGLD is able to find and characterize all modes

# Bayesian Neural Networks



CIFAR: 50k images



ResNet: ~ 11 million params.

| | CIFAR-10 | CIFAR-100 |
|---|---|---|
| SGD | 5.29±0.15 | 23.61±0.09 |
| SGDM | 5.17±0.09 | 22.98±0.27 |
| Snapshot-SGD | 4.46±0.04 | 20.83±0.01 |
| Snapshot-SGDM | 4.39±0.01 | 20.81±0.10 |
| SGLD | 5.20±0.06 | 23.23±0.01 |
| cSGLD (ours) | 4.29±0.06 | 20.55±0.06 |
| SGHMC | 4.93±0.1 | 22.60±0.17 |
| cSGHMC (ours) | **4.27**±0.03 | **20.50**±0.11 |

**Table 1:** Comparison of test error (%).

- cSG-MCMC outperforms SG-MCMC and optimization methods.

# ImageNet


~ 14 million images

| | NLL ↓ | Top1 ↑ | Top5 ↑ |
|---|---|---|---|
| SGDM | 0.9595 | 76.046 | 92.776 |
| Snapshot-SGDM | 0.8941 | 77.142 | 93.344 |
| SGHMC | 0.9308 | 76.274 | 92.994 |
| cSGHMC | **0.8882** | 77.114 | 93.524 |

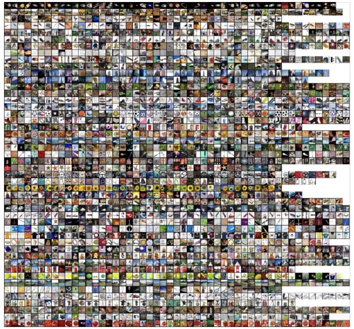
- cSG-MCMC gives the best testing NLL and Top5 accuracy
- One of the first work making MCMC work on ImageNet

# Impact of cSG-MCMC

**How Good is the Bayes Posterior in Deep Neural Networks Really?**

**Florian W** **Stephan** **Bayesian Neural Network Priors Revisited** **Linh Tran** [5] + **an Nowozin** [7] +

**What Are Bayesian Neural Network Posteriors Really Like?**

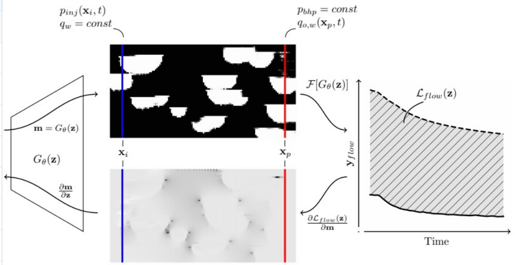


Large-scale image classification
[Heek et al. 2020]

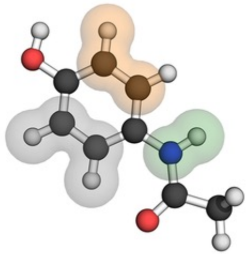
Uncertainty Baselines Team and znado Project import generated by Copybara. ... 643c2e4 10 days ago 259 commits

| | | |
|---|---|---|
| baselines | Project import generated by Copybara. | 10 days ago |
| experimental | Project import generated by Copybara. | 10 days ago |
| uncertainty_baselines | Project import generated by Copybara. | 10 days ago |
| .gitignore | Project import generated by Copybara. | 9 months ago |
| .travis.yml | Clean up travis.yml (and ed2's setup.py) across Ed2, Baselines, Metri... | 8 months ago |
| CONTRIBUTING.md | Project import generated by Copybara. | 9 months ago |
| LICENSE | Project import generated by Copybara. | 9 months ago |
| README.md | Moving references to references.md, updating contributors, in READ... | 4 months ago |
| pylintrc | Fixing Travis lint errors. | 8 months ago |
| setup.py | Project import generated by Copybara. | 10 days ago |

Uncertainty Baselines library

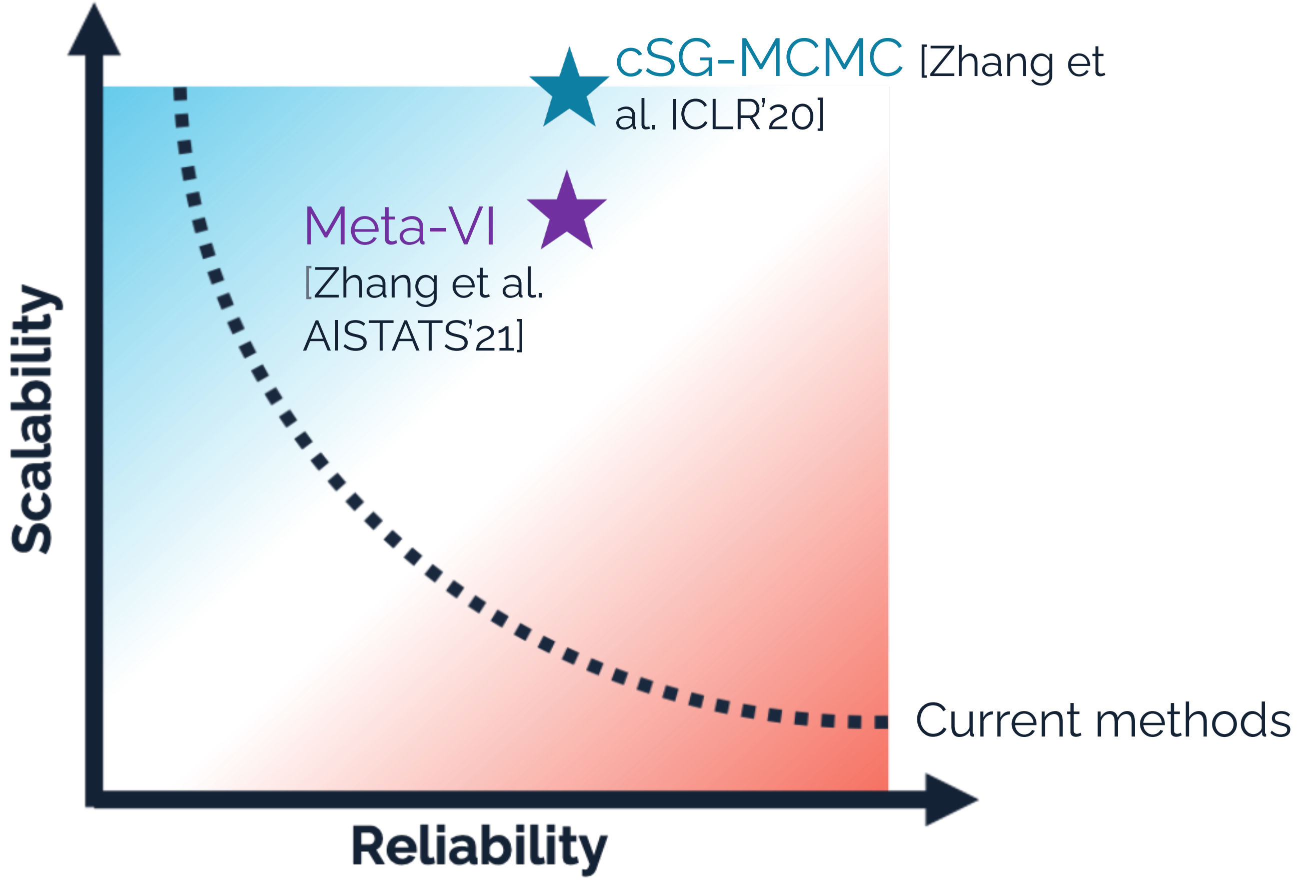



Earth's subsurface calibration
[Mosser et al. 2019]



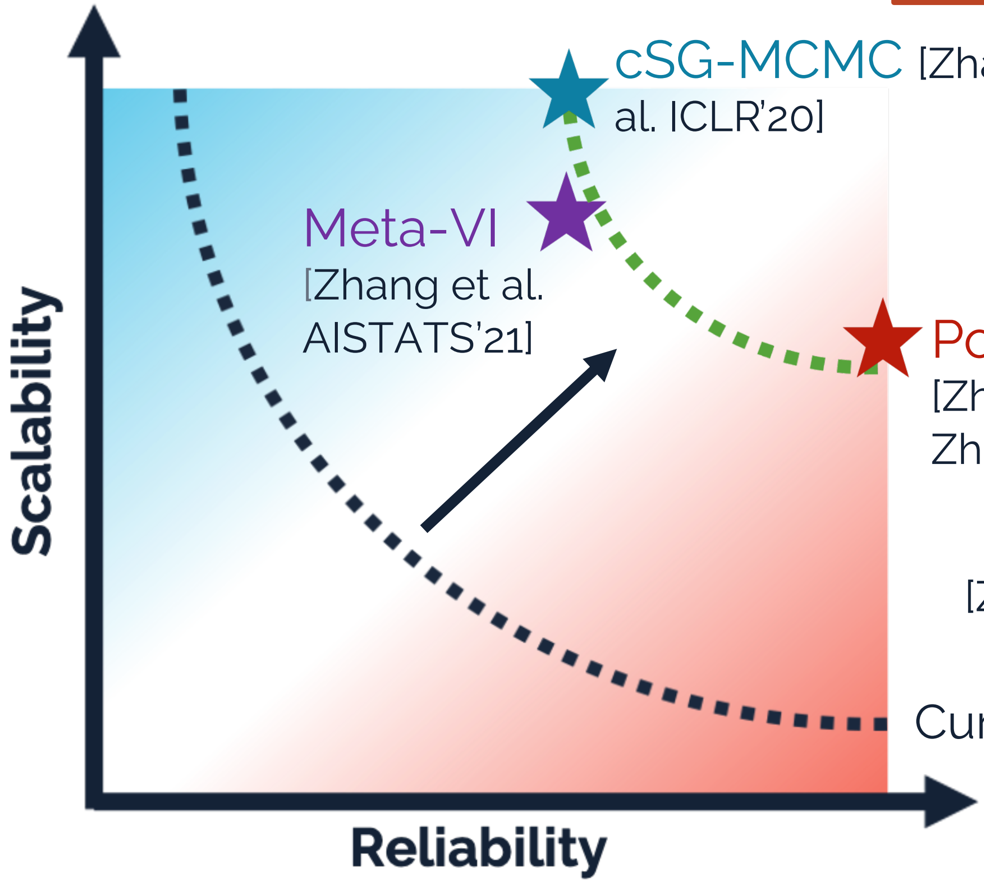
Molecular property prediction [Lamb et al. 2020]

# Efficient Inference for Reliable Deep Learning



# **Push the Frontier**

Open-source:
https://github.com/ruqizhang/




Poisson-Minibatching
[Zhang et al. NeurIPS'20, Zhang et al. NeurIPS'19]

AMAGOLD

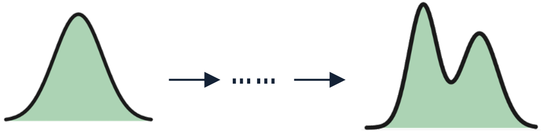
[Zhang et al., AISTATS'20]

Current methods

# Thank you!

Theoretically-Guaranteed Inference

minibatch ≈ dataset



Efficient Inference for Reliable Deep Learning