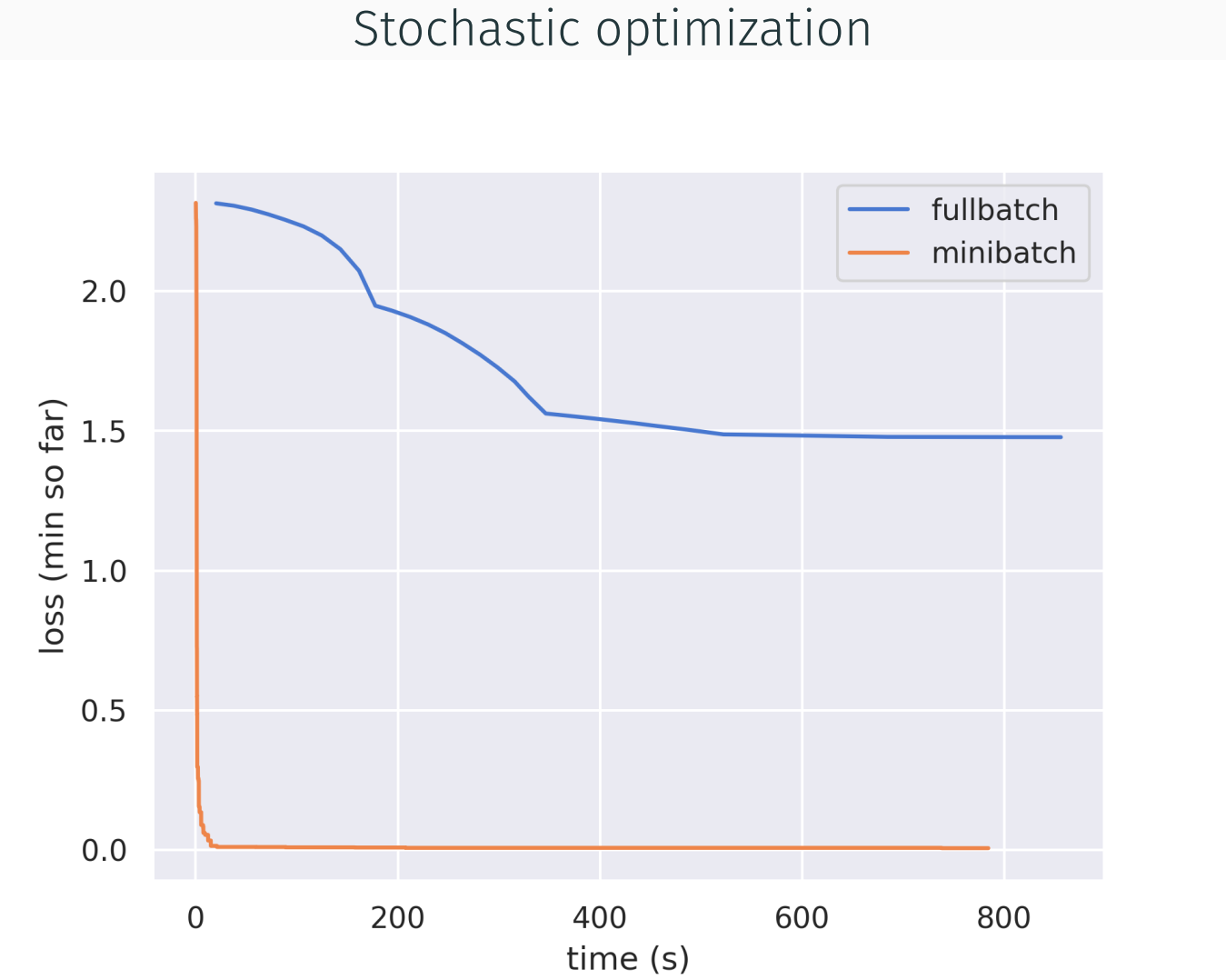
Stochastic Gradient Annealed Importance Sampling

Scott Cameron
Hans Eggers
Steve Kroon

Stellenbosch University
NITheP

Motivation


Stochastic optimization
fullbatch
minibatch
loss (min so far)
2.0
1.5
1.0
0.5
0.0
0
200
400
600
800
time (s)

Motivation

Goal: *Efficient large-scale marginal likelihood estimation using mini-batches*

Marginal Likelihood (Evidence)

Consider a Bayesian model

$$\mathcal{D} = \{y_n\}_{n=1}^N \qquad p(\mathcal{D}, \theta) = p(\theta) \prod_n p(y_n|\theta)$$

Posterior given by Bayes theorem

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

Marginal likelihood

$$\mathcal{Z} := p(\mathcal{D}) = \int p(\mathcal{D}|\theta)p(\theta)\, d\theta$$

Posterior predictive

$$p(y'|\mathcal{D}) = \int p(y'|\theta)p(\theta|\mathcal{D})\, d\theta$$

Model Comparison/Combination

Posterior over models $\mathcal{M}_1, \mathcal{M}_2, \cdots$

$$\frac{P(\mathcal{M}_1|\mathcal{D})}{P(\mathcal{M}_2|\mathcal{D})} = \frac{\mathcal{Z}_1}{\mathcal{Z}_2}\frac{p(\mathcal{M}_1)}{p(\mathcal{M}_2)}$$

$\mathcal{M}_1$ is a 'better' model than $\mathcal{M}_2$ if $\mathcal{Z}_1 \gg \mathcal{Z}_2$

Combined predictions

$$p(y'|\mathcal{D}) = \frac{\sum_i p(y'|\mathcal{D}, \mathcal{M}_i)\mathcal{Z}_i\, p(\mathcal{M}_i)}{\sum_i \mathcal{Z}_i\, p(\mathcal{M}_i)}$$

Weighs models proportionately to how well they describe data

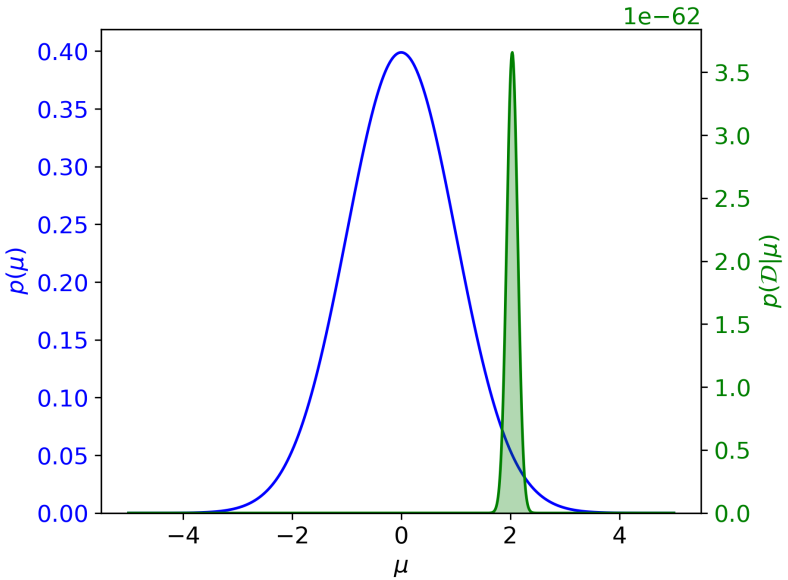
Why is this difficult?

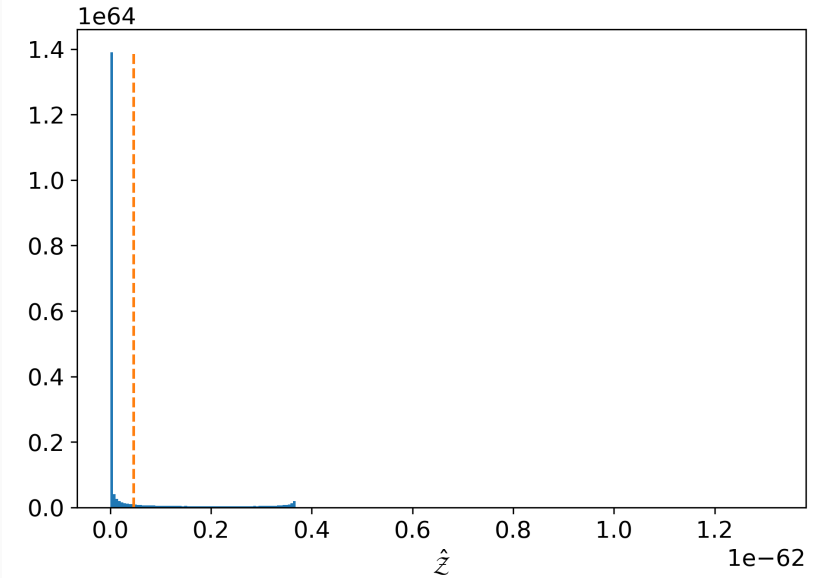
Example model

$$\mu \sim \mathcal{N}(0, 1) \qquad y_n \sim \mathcal{N}(\mu, 1)$$

Naive estimator

$$\hat{\mathcal{Z}} = \frac{1}{M} \sum_{i=1}^{M} p(\mathcal{D}|\mu_i) \qquad \mu_i \sim p(\mu)$$

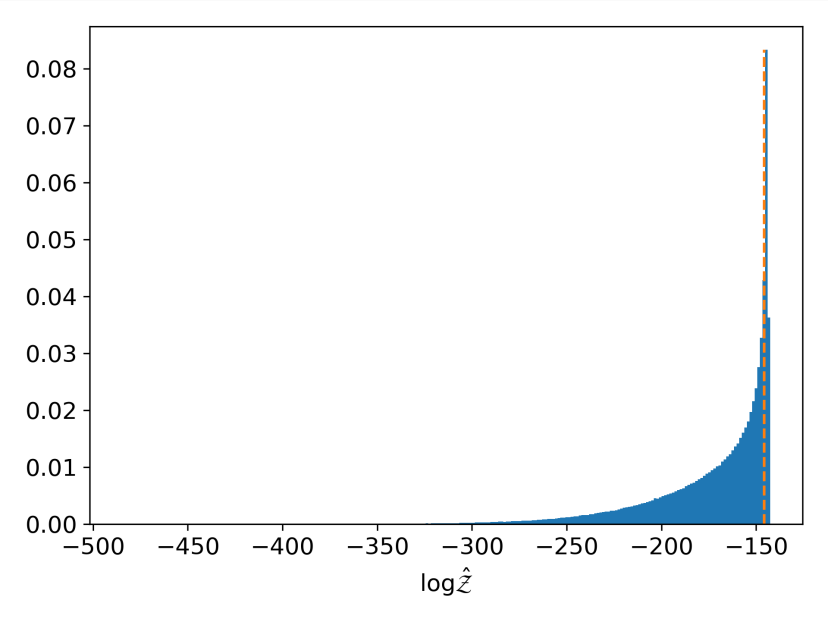




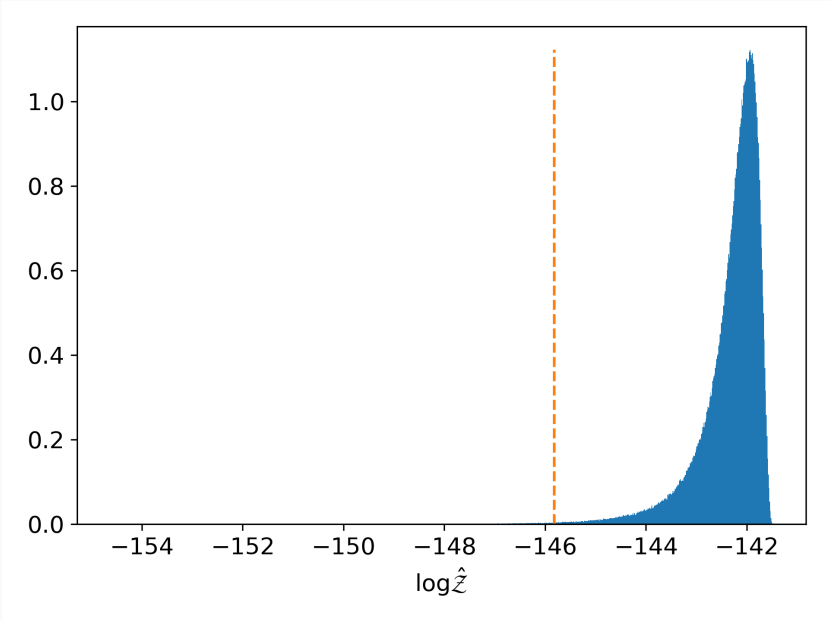
Why is this difficult?

Consistently underestimate/overestimate

Prior samping



Harmonic mean



Annealed Importance Sampling

Adiabatically decrease temperature: $0 = \lambda_0 < \cdots < \lambda_T = 1$

$$f_t(\theta) = p(\mathcal{D}|\theta)^{\lambda_t} p(\theta)$$

Update particles with HMC[1]

$$U_t(\theta) = -\lambda_t \log p(\mathcal{D}|\theta) - \log p(\theta)$$

Iterated importance sampling

$$w_i^{(t)} \leftarrow w_i^{(t-1)} \, p(\mathcal{D}|\theta_i^{(t-1)})^{\lambda_t - \lambda_{t-1}}$$

Estimator

$$\hat{\mathcal{Z}} = \frac{1}{M} \sum_{i=1}^{M} w_i^{(T)}$$

[1] Hamiltonian Monte Carlo

Annealed Importance Sampling

Adiabatically decrease temperature: $0 = \lambda_0 < \cdots < \lambda_T = 1$

$$f_t(\theta) = p(\mathcal{D}|\theta)^{\lambda_t} p(\theta)$$

Update particles with HMC[1]

$$U_t(\theta) = -\lambda_t \log p(\mathcal{D}|\theta) - \log p(\theta)$$

Iterated importance sampling

$$w_i^{(t)} \leftarrow w_i^{(t-1)} \, p(\mathcal{D}|\theta_i^{(t-1)})^{\lambda_t - \lambda_{t-1}}$$

Estimator

$$\hat{\mathcal{Z}} = \frac{1}{M} \sum_{i=1}^{M} w_i^{(T)}$$

[1] Hamiltonian Monte Carlo

Problems with Scalability

Accurate estimates require $T \propto |\mathcal{D}|$

1. HMC needs likelihood gradients, $\mathcal{O}(|\mathcal{D}|)$

2. Importance weights need likelihood, $\mathcal{O}(|\mathcal{D}|)$

More or less $\mathcal{O}\left(|\mathcal{D}|^2\right)$ complexity

Stochastic Gradient HMC

Simulate Langevin dynamics

$$\dot{\theta} = v$$
$$\dot{v} = -\nabla U(\theta) - \gamma v + \sqrt{2\gamma}\,\xi \qquad \langle \xi(t)\xi(t') \rangle = \delta(t - t')$$

Fokker–Planck equation[2]

$$\frac{\partial p}{\partial t} = \partial^T A \left\{ p\, \partial H + \partial p \right\} \qquad A = \begin{pmatrix} 0 & -I \\ I & \gamma \end{pmatrix}$$

Canonical ensemble

$$p_\infty(\theta, v) = \frac{1}{Z} e^{-H(\theta, v)}$$

[2] $H(\theta, v) = U(\theta) + \frac{1}{2} v^2$

Stochastic Gradient HMC

Euler–Maruyama discretization

$$\Delta\theta = v$$
$$\Delta v = -\eta\nabla\hat{U}(\theta) - \alpha\, v + \mathcal{N}(0, 2(\alpha - \hat{\beta})\eta)$$

Mini-batch energy estimate

$$\hat{U}(\theta) = -\frac{|\mathcal{D}|}{|B|}\sum_{y\in B} \log p(y|\theta) - \log p(\theta)$$

Time complexity $\mathcal{O}(|B|) \ll \mathcal{O}(|\mathcal{D}|)$

Stochastic Gradient HMC

Euler–Maruyama discretization

$$\Delta\theta = v$$
$$\Delta v = -\eta\nabla\hat{U}(\theta) - \alpha\, v + \mathcal{N}(0, 2(\alpha - \hat{\beta})\eta)$$

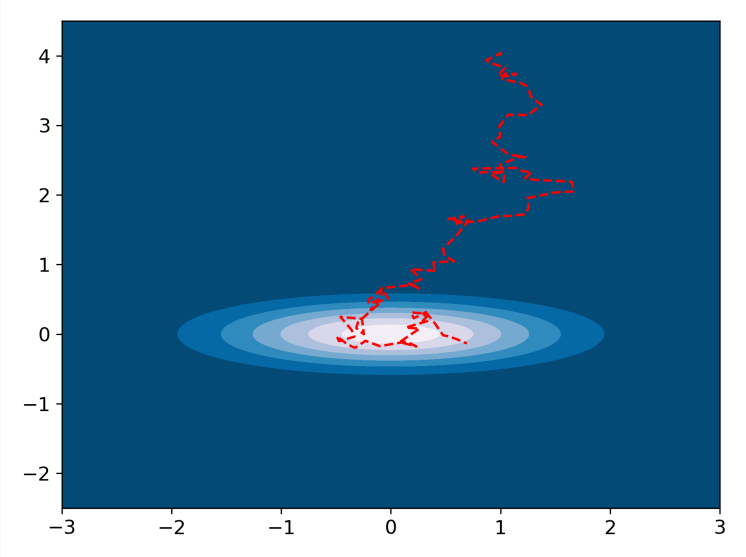
Mini-batch energy estimate

$$\hat{U}(\theta) = -\frac{|\mathcal{D}|}{|B|}\sum_{y\in B} \log p(y|\theta) - \log p(\theta)$$

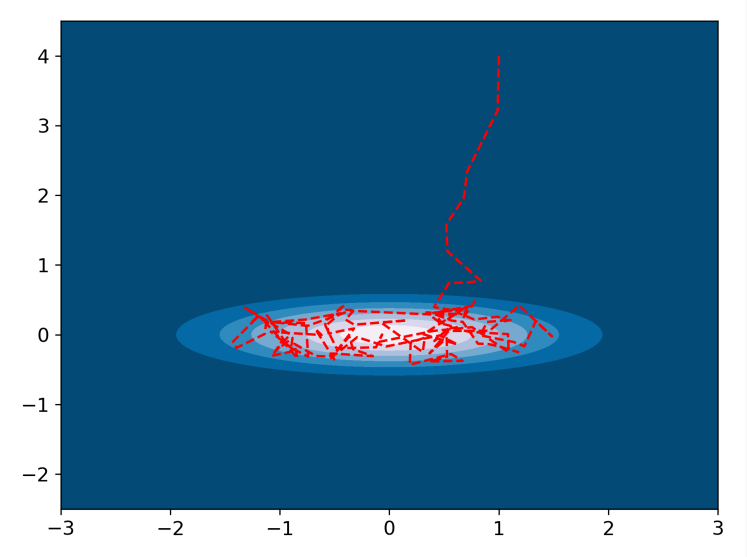
Time complexity $\mathcal{O}(|B|) \ll \mathcal{O}(|\mathcal{D}|)$
solves (1)

Comparison of MCMC Trajectories

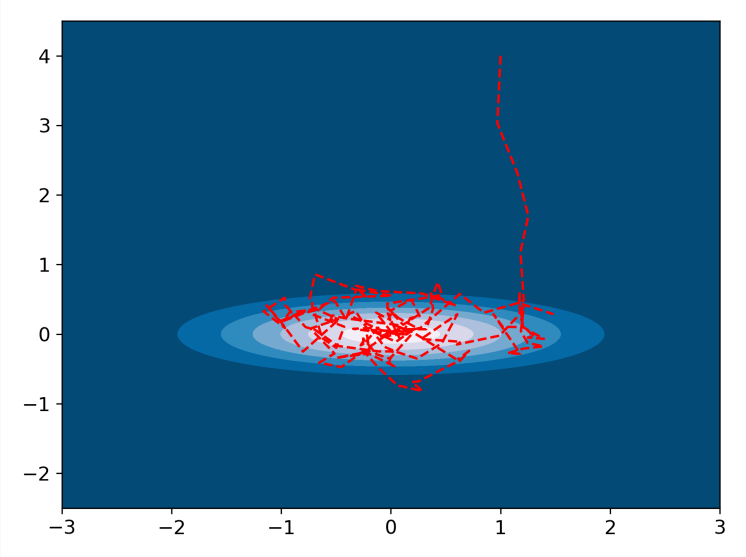
RWMH



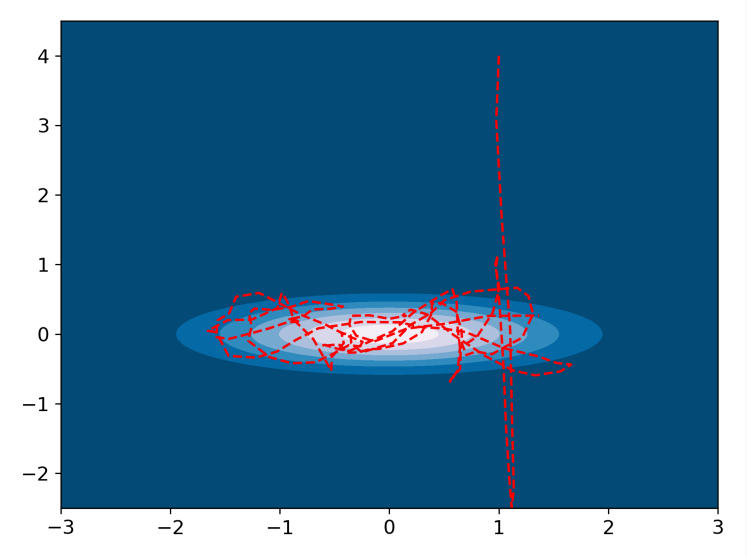
HMC



SGLD



SGHMC



Bayesian Updating/Online Estimation

Predictive distributions

$$\mathcal{Z} = \prod_n p(y_n|y_{<n}) = \prod_n \int p(y_n|\theta)p(\theta|y_{<n})$$

Estimate $p(y_n|y_{<n})$ with AIS

$$\theta_i^{(n)},\ \tilde{w}_i^{(n)} \leftarrow \text{AIS}(y_n, \theta_i^{(n-1)})$$

Marginal likelihood

$$\hat{\mathcal{Z}} = \frac{1}{M}\sum_{i=1}^{M}\prod_n \tilde{w}_i^{(n)}$$

Bayesian Updating/Online Estimation

Predictive distributions

$$\mathcal{Z} = \prod_n p(y_n|y_{<n}) = \prod_n \int p(y_n|\theta)p(\theta|y_{<n})$$

Estimate $p(y_n|y_{<n})$ with AIS

$$\theta_i^{(n)},\ \tilde{w}_i^{(n)} \leftarrow \text{AIS}(y_n, \theta_i^{(n-1)})$$

Marginal likelihood

$$\hat{\mathcal{Z}} = \frac{1}{M}\sum_{i=1}^{M}\prod_n \tilde{w}_i^{(n)}$$

solves (2)

Stochastic Gradient Annealed Importance Sampling

Intermediate distributions

$$f_n^{(\lambda)}(\theta) = p(y_n|\theta)^{\lambda} \left[\prod_{k<n} p(y_k|\theta)\right] p(\theta)$$

Update particles with SGHMC

$$\hat{U}_n^{(\lambda)}(\theta) = -\lambda \log p(y_n|\theta) - \frac{n-1}{|B|} \sum_{y \in B} \log p(y|\theta) - \log p(\theta)$$

Importance weights

$$w_i^{(t)} \leftarrow w_i^{(t-1)}\, p(y_n|\theta_i^{(t-1)})^{\lambda_t - \lambda_{t-1}}$$

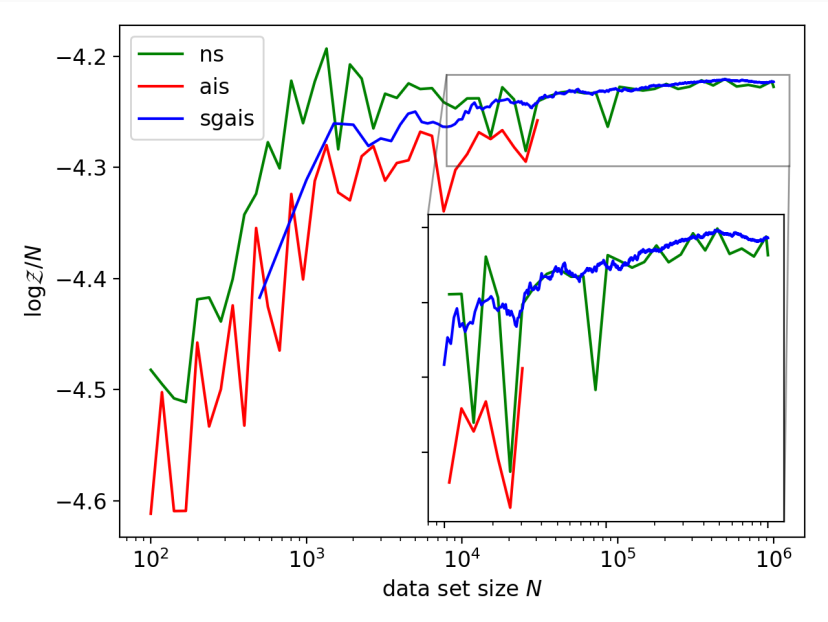
ML estimator

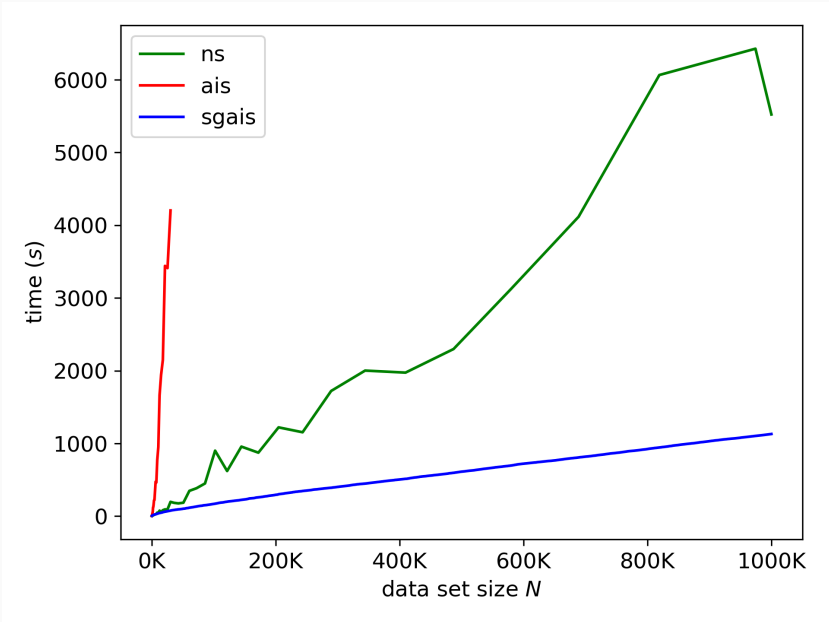
$$\hat{\mathcal{Z}} = \frac{1}{M} \sum_{i=1}^{M} w_i^{(T)}$$

Results

Gaussian mixture model

- vs nested sampling
- vs annealed importance sampling

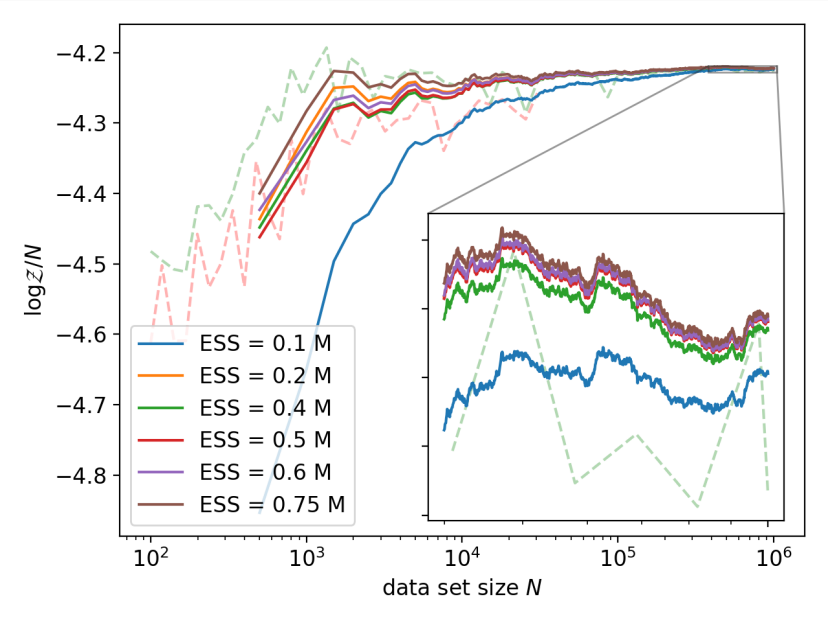


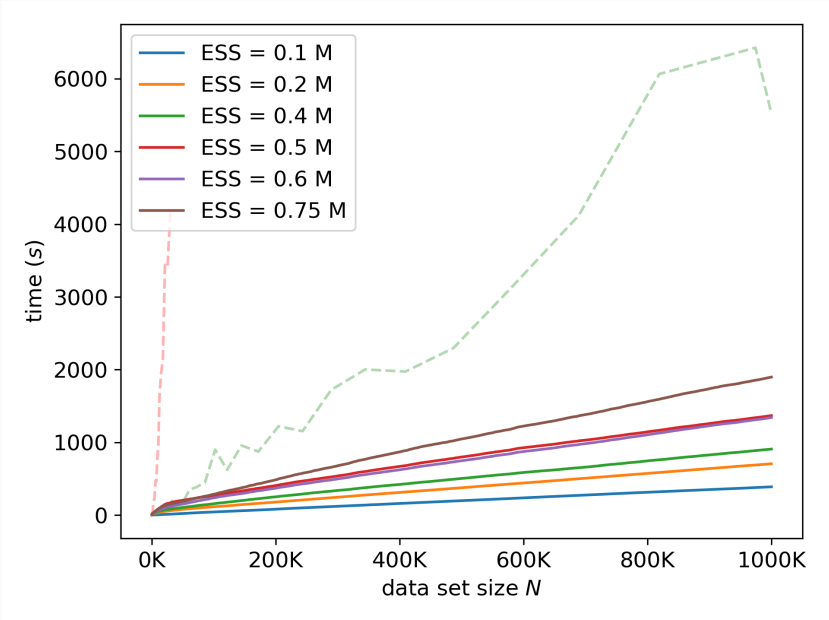


Parameter sensitivity

Adaptive annealing schedule

- Blue ≈ no annealing steps

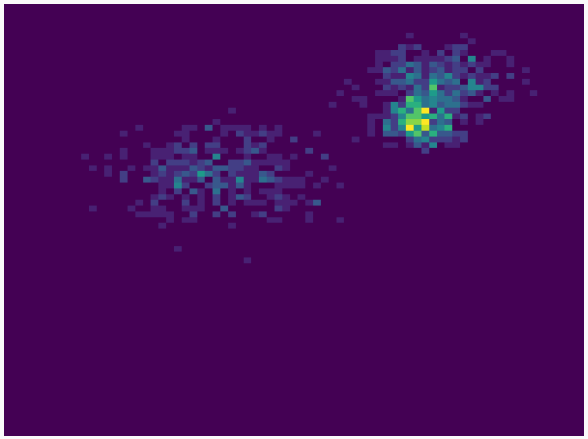




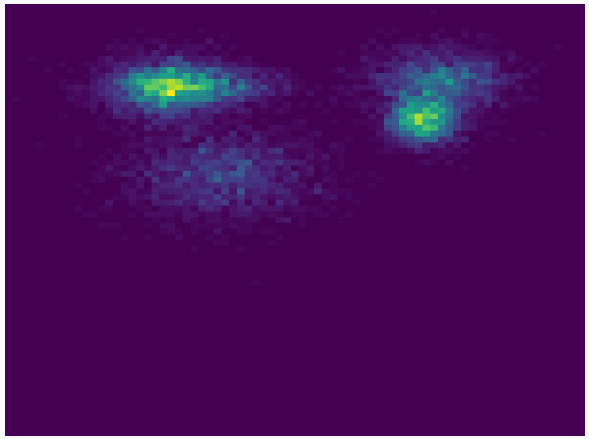
Distribution Shift

Data may change over time

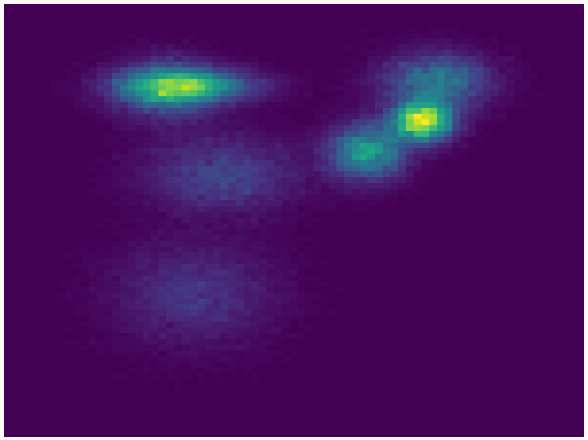
$1 \leq n \leq 10^3$



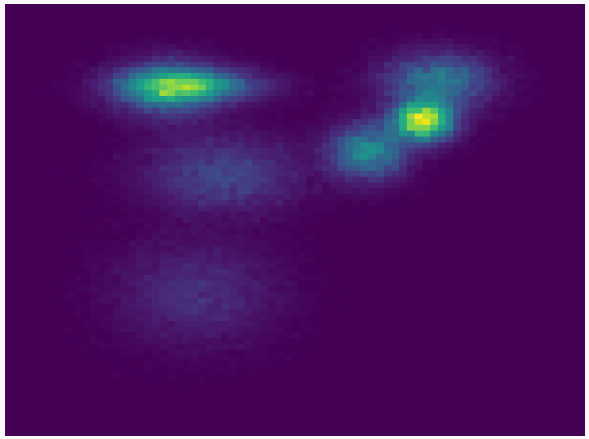
$10^3 < n \leq 10^4$



$10^4 < n \leq 10^5$

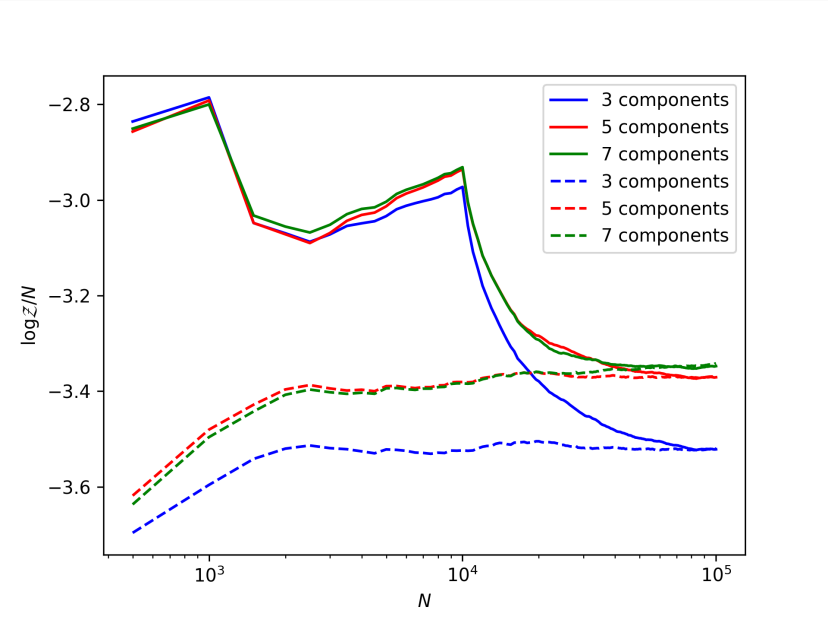


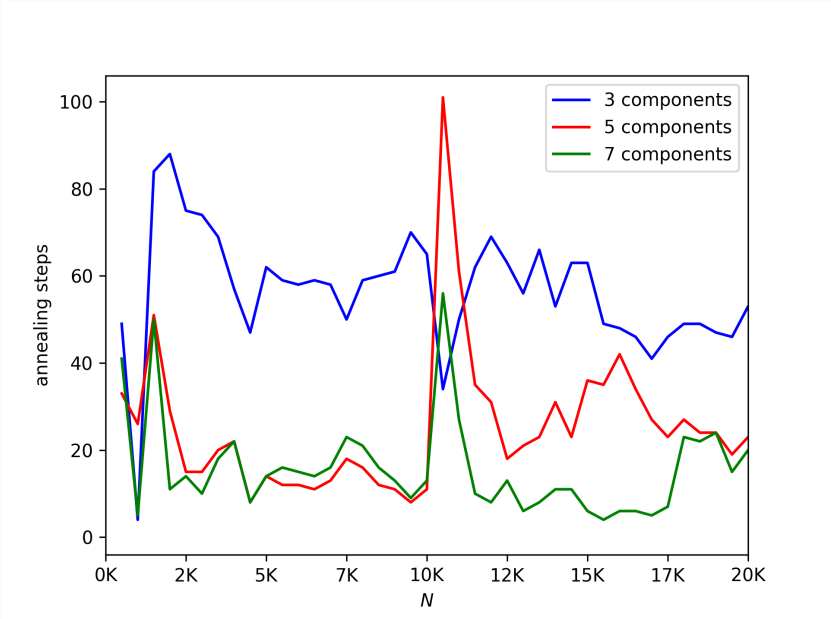
total



Distribution Shift

Dashed lines = shuffled data





Thank You!

[1] Cameron, S.A.; Eggers, H.C.; Kroon, S.
Stochastic Gradient Annealed Importance Sampling for Efficient Online Marginal Likelihood Estimation.
Entropy 21.11 (2019).

[2] Chen, T.; Fox, E.; Guestrin, C.
Stochastic Gradient Hamiltonian Monte Carlo.
ICML Proceedings vol. 5. (2014).

Funded by NIThePP[3]

Paper sponsored by MaxEnt 2019

Big thanks to Hans and Steve!

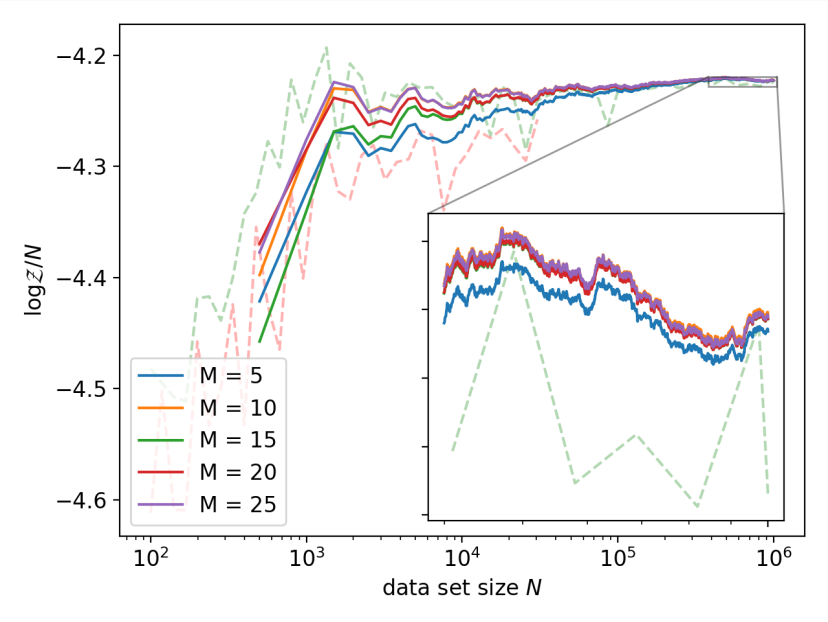
[3] National Institute of Theoretical Physics

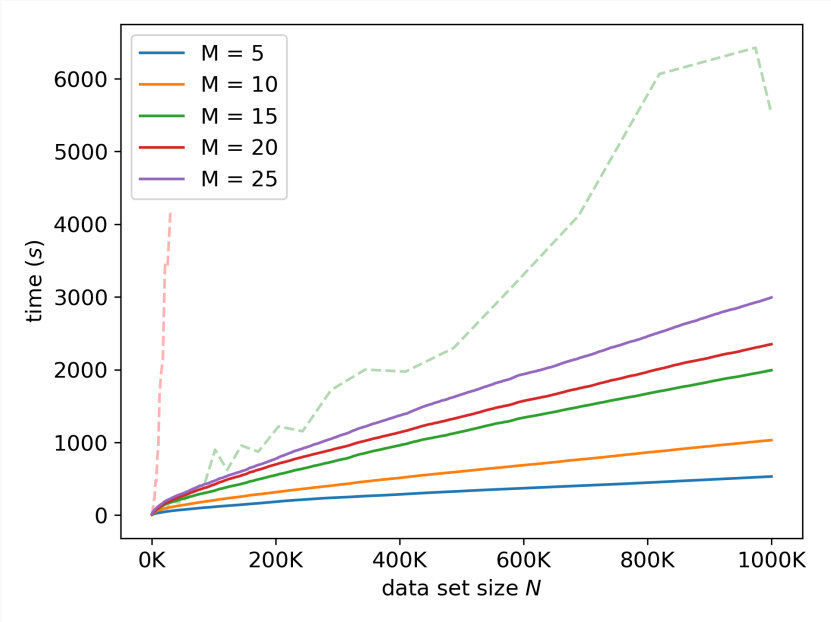
Extra Slides

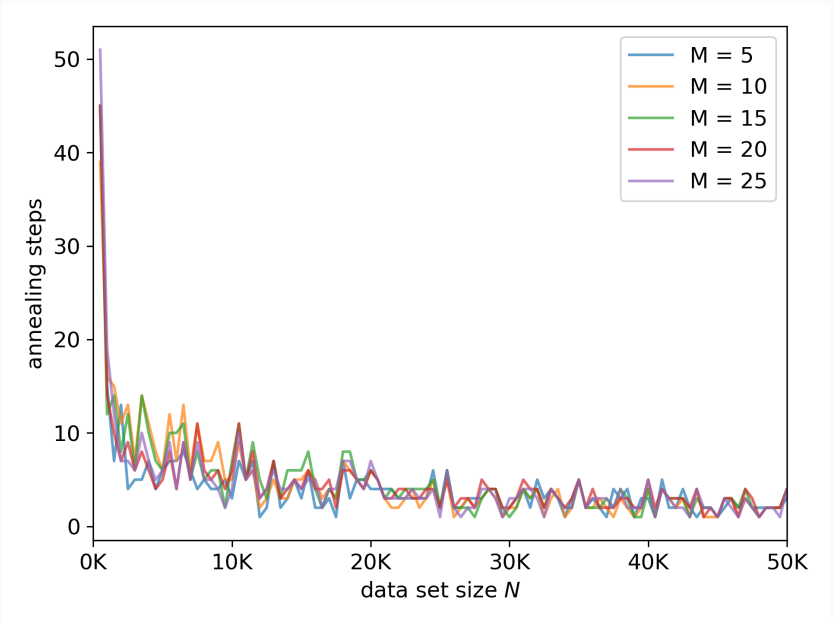
Algorithm 1 Stochastic Gradient Annealed Importance Sampling

$\forall_i$: sample $\theta_i \sim p(\theta)$
$\forall_i$: $w_i \leftarrow 1$
for $n = 1, \ldots, N$ **do**
    $\lambda \leftarrow 0$
    while $\lambda < 1$ **do**
        $\Delta \leftarrow \text{argmin}_\Delta[\text{ESS}(\Delta) - \text{ESS}^*]$
        $\lambda \leftarrow \lambda + \Delta$
        $\forall_i$: $w_i \leftarrow w_i \, p(y_n|\theta_i)^\Delta$ ▷ optionally resample particles
        $\forall_i$: $\theta_i \leftarrow \text{SGHMC}(\theta_i, \hat{U}_n^{(\lambda)})$
    end while
end for
return $\hat{\mathcal{Z}} = \frac{1}{M} \sum_i w_i$

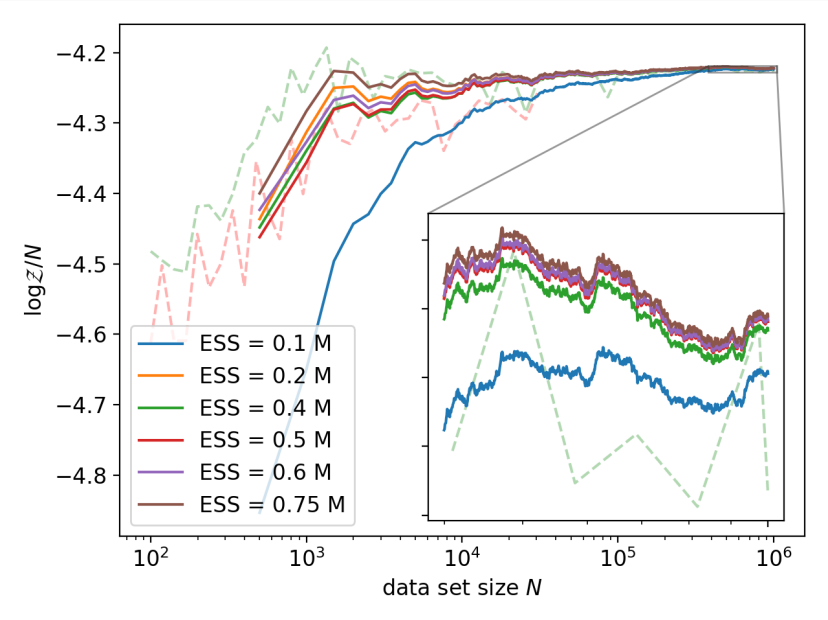
Number of Particles

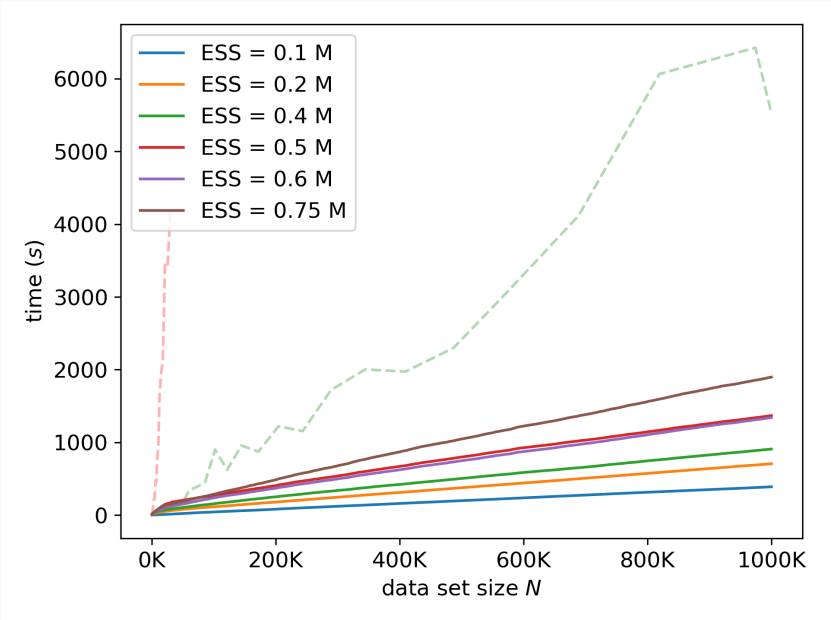


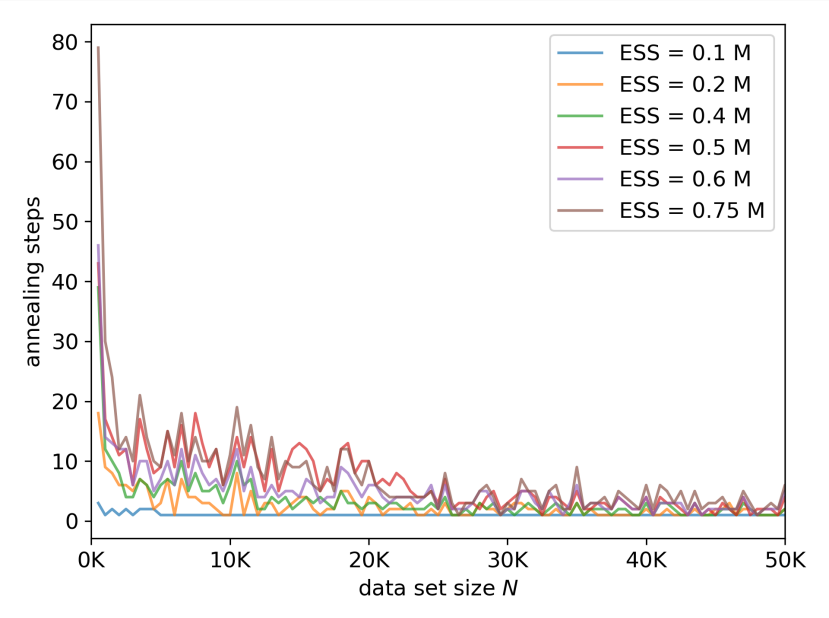




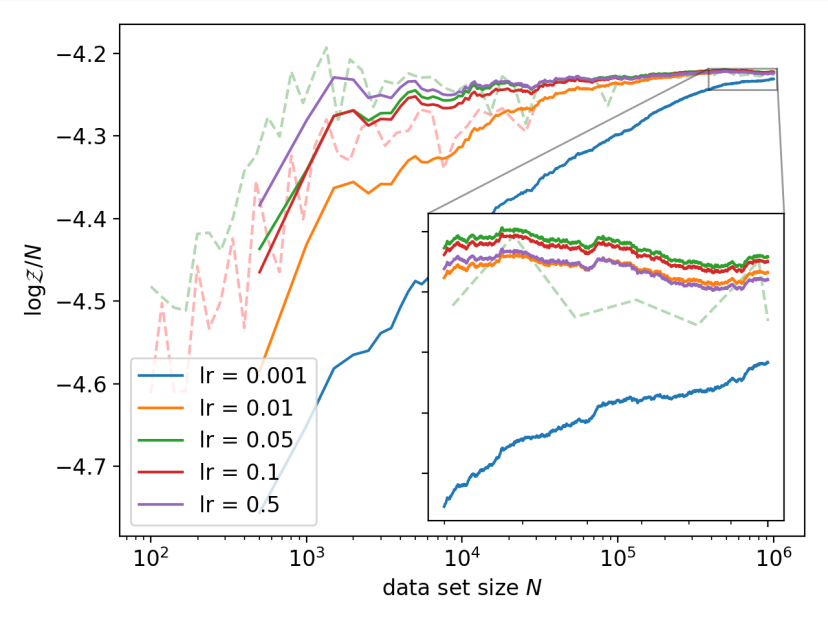
Effective Sample Size

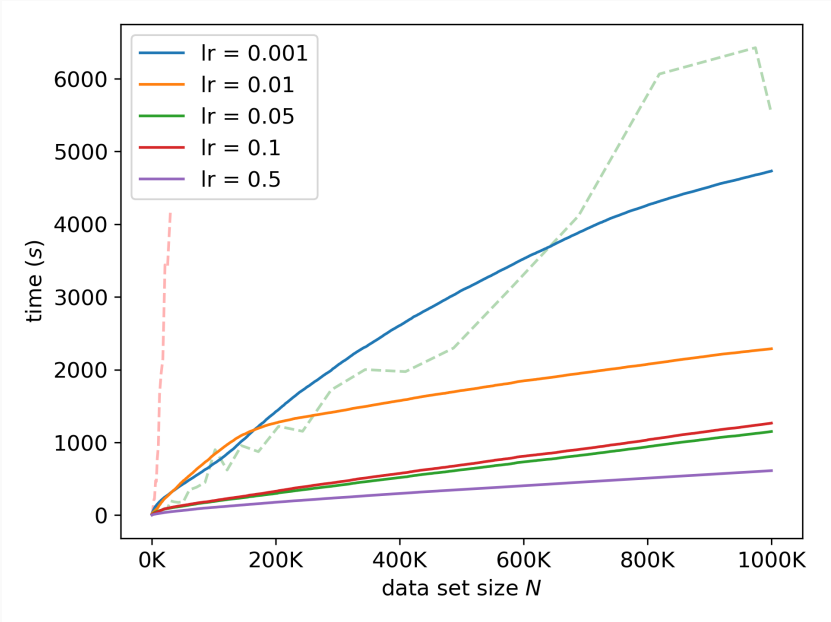


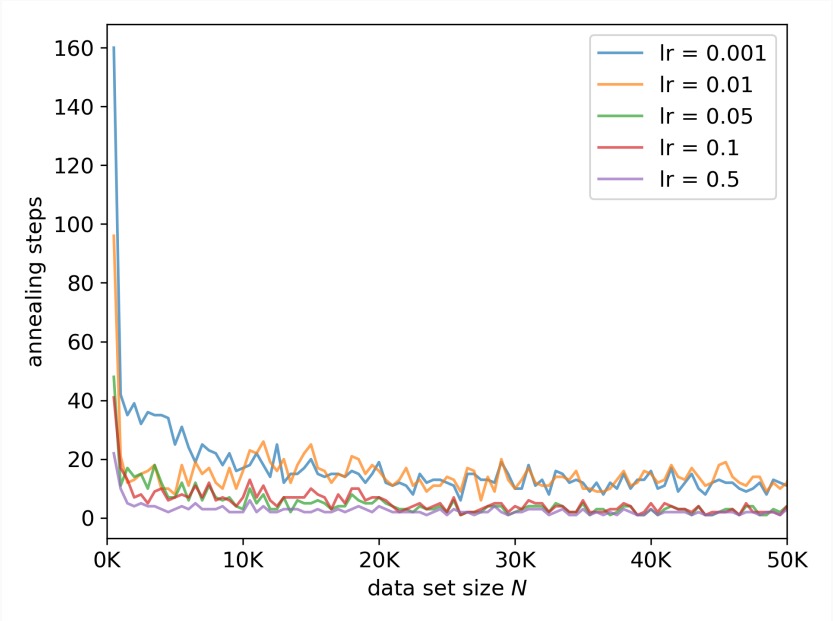




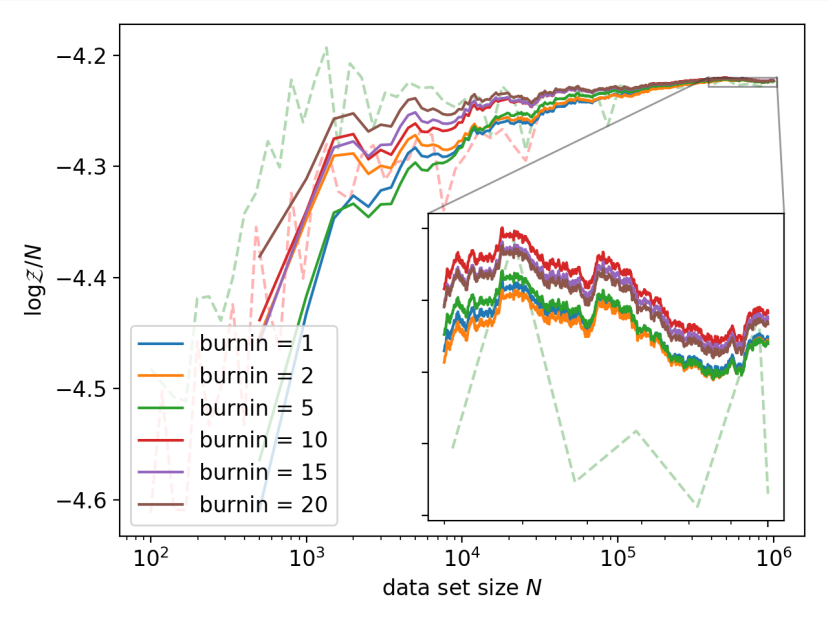
Learning Rate

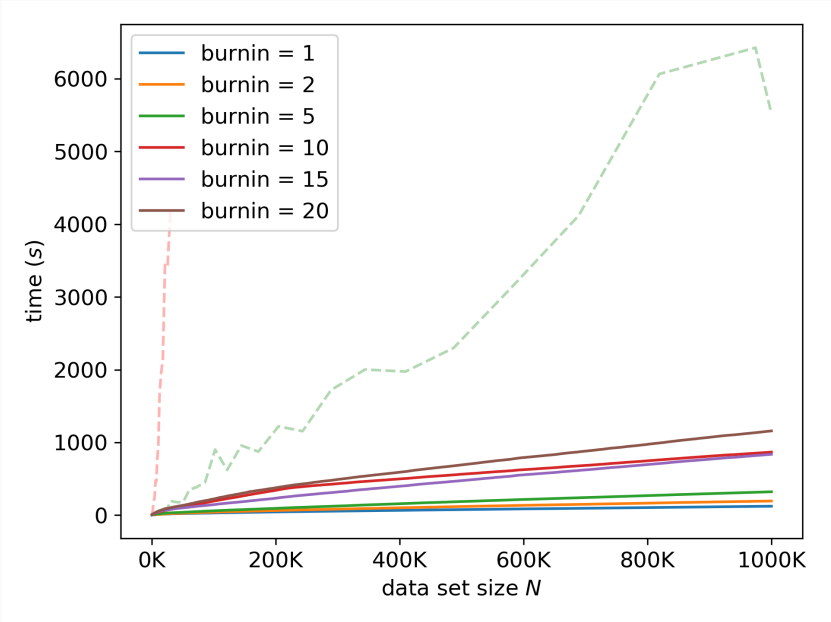


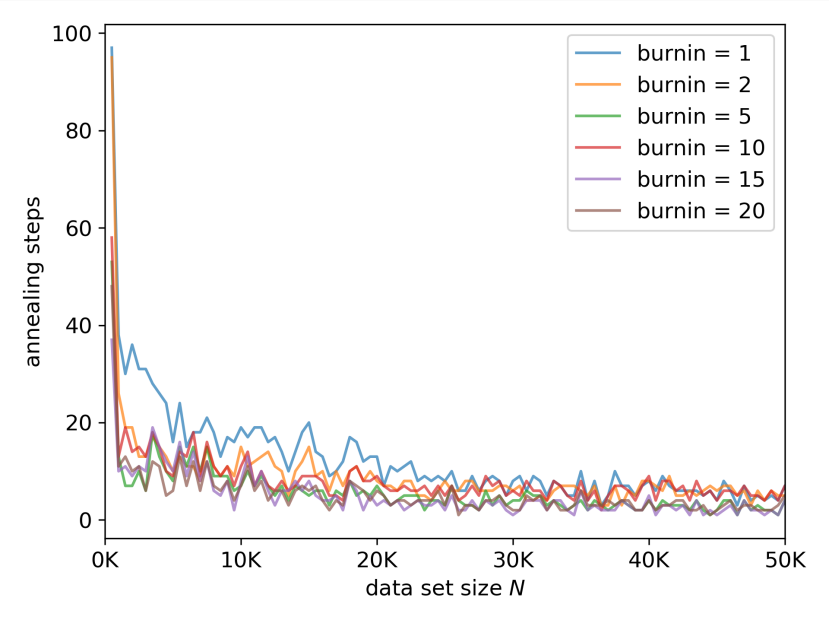




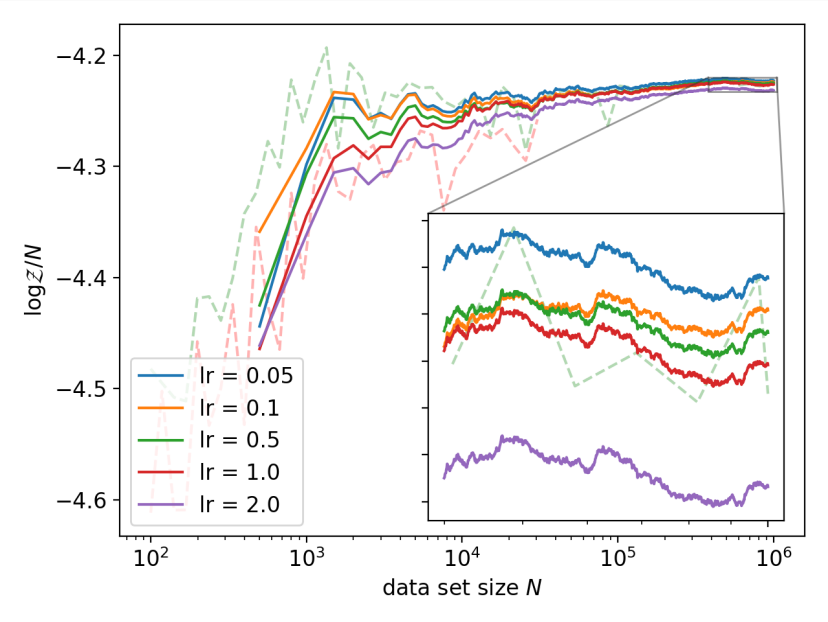
Burnin

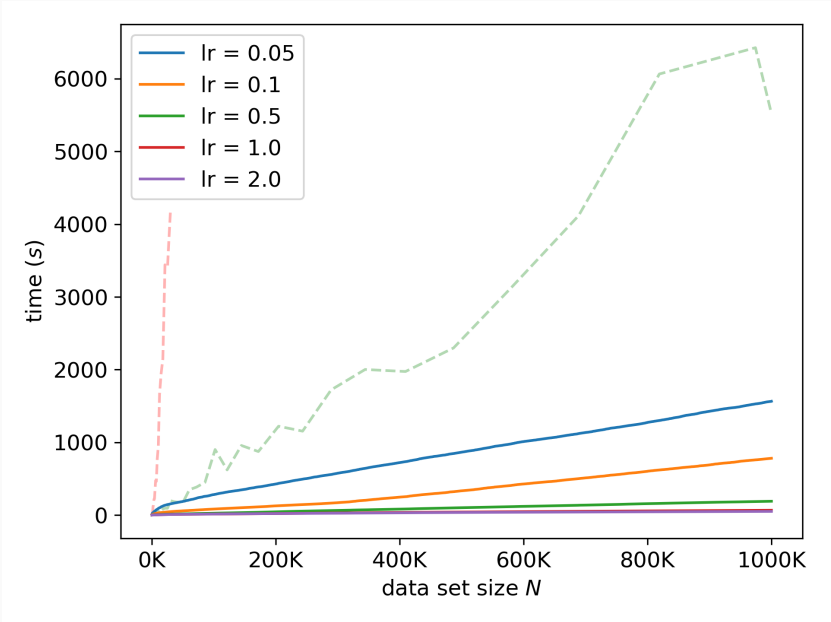

−4.2
−4.3
−4.4
−4.5
−4.6
log$\mathcal{Z}$/N
10^2
10^3
10^4
10^5
10^6
data set size N
burnin = 1
burnin = 2
burnin = 5
burnin = 10
burnin = 15
burnin = 20

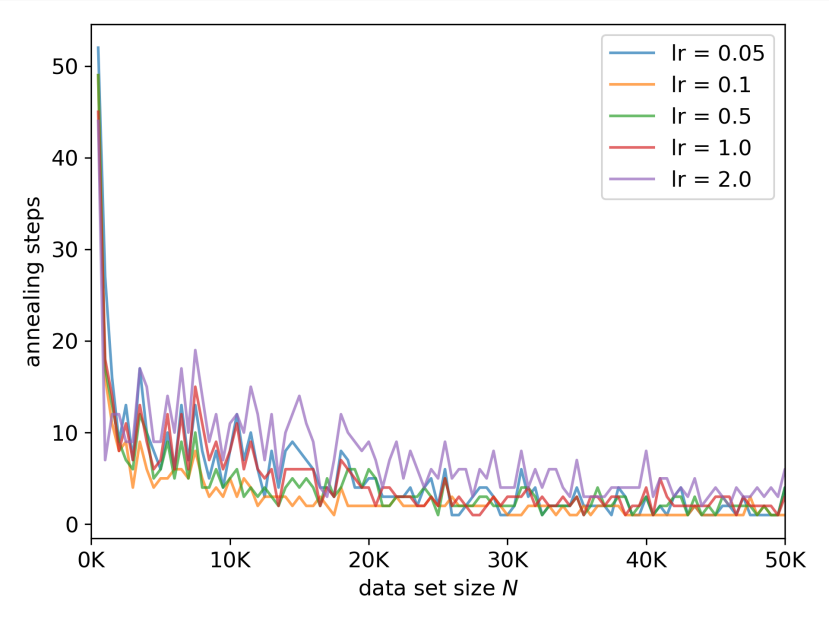

6000
5000
4000
3000
2000
1000
0
time (s)
0K
200K
400K
600K
800K
1000K
data set size N
burnin = 1
burnin = 2
burnin = 5
burnin = 10
burnin = 15
burnin = 20


100
80
60
40
20
0
annealing steps
0K
10K
20K
30K
40K
50K
data set size N
burnin = 1
burnin = 2
burnin = 5
burnin = 10
burnin = 15
burnin = 20

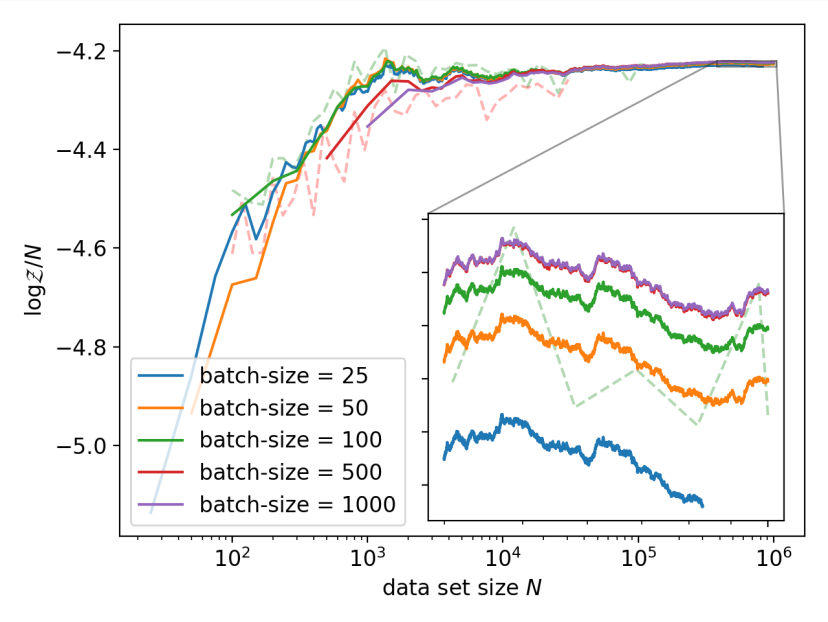
Learning Rate $\times$ Burnin

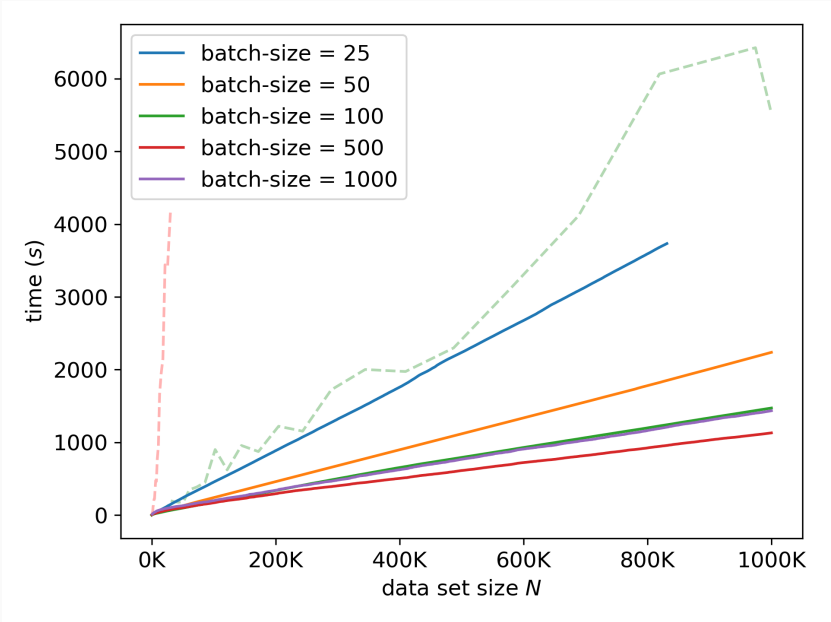


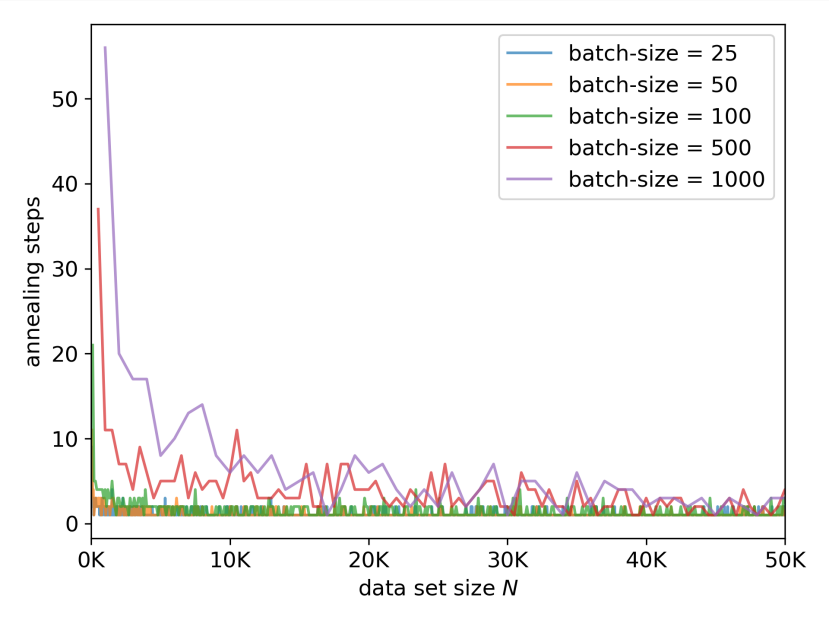




Batch Size


−4.2
−4.4
−4.6
−4.8
−5.0
log Z/N
batch-size = 25
batch-size = 50
batch-size = 100
batch-size = 500
batch-size = 1000
10^2
10^3
10^4
10^5
10^6
data set size N


batch-size = 25
batch-size = 50
batch-size = 100
batch-size = 500
batch-size = 1000
6000
5000
4000
3000
2000
1000
0
time (s)
0K
200K
400K
600K
800K
1000K
data set size N


batch-size = 25
batch-size = 50
batch-size = 100
batch-size = 500
batch-size = 1000
50
40
30
20
10
0
annealing steps
0K
10K
20K
30K
40K
50K
data set size N