Gradient Importance Sampling

Ingmar Schuster
Université Paris Dauphine

March 10, 2016

Section 1

Outline

Section 2

Introduction

Monte Carlo techniques

Monte Carlo techniques

- approximate integral/expected value H of some function h wrt density π
- full uncertainty information about H can be estimated
- straight forward when we can draw samples from π directly
- else
  - Markov Chain Monte Carlo (MCMC): construct Markov Chain to sample approximately from π after chain reaches stationary regime
  - Importance Sampling and Sequential Monte Carlo (IS/SMC): draw from an auxiliary distribution q instead of π and reweight the samples to account for change of distribution

Importance Sampling estimators

- reweighting in Importance Sampling justified by

$$
\begin{aligned}
H = & \int \pi(x) h(x) \mathrm{d}x = \int \frac{\pi(x)}{q(x)} h(x) q(x) \mathrm{d}x \\
= & \mathbb{E}_q \left(\frac{\pi(x)}{q(x)} h(x) \right) = \mathbb{E}_q \left(w(x) h(x) \right)
\end{aligned}
$$

for $w(x) = \frac{\pi(x)}{q(x)}$

Standard Importance Sampling

- from law of large numbers obtain the *standard Importance Sampling* estimator for $\mathbb{E}_q\left(w(X)h(X)\right)$:

$$\mathfrak{I}(\mathbf{X}) = \frac{1}{|\mathbf{X}|}\sum_{X\in\mathbf{X}} w(X)h(X)$$

  where $\mathbf{X}$ iid from q and $w(X) = \pi(X)/q(X)$ is the importance weight.
- unbiased and consistent by standard results (Robert and Casella, 2004)
- as rule of thumb to ensure finite variance, q should have heavier tails than π and whenever $\pi(x) \neq 0$ also $q(x) \neq 0$

Self-Normalized Importance Sampling

- in case π is unnormalized, the *self-normalized Importance Sampling* estimator can be used:

$$\mathfrak{I}_{\mathrm{n}}(\mathbf{X}) = \frac{1}{w_{\Sigma}(\mathbf{X})} \sum_{X \in \mathbf{X}} w_u(X) h(X)$$

  where $\mathbf{X}$ iid from q, $w_u(X) = \pi(X)/q(X)$ is the unnormalized importance weight and $w_{\Sigma}(\mathbf{X}) = \sum_{X \in \mathbf{X}} w_u(X)$.
- consistent by standard results, unbiased asymptotically in $|\mathbf{X}|$, often has lower variance than $\mathfrak{I}$ (Robert and Casella, 2004)
- again: to ensure finite variance, q should have heavier tails than π
- Choice of q vital to performance!

Importance Resampling

- *Importance Resampling* We can get rid of importance weights by resampling samples with replacement where probability of sample X being resampled is $p(X) = w_u(X)/w_\Sigma(\mathbf{X})$
- We obtain $\mathbf{X}'$ from this procedure, now we can approximate the integral of interest by

$$\frac{1}{|\mathbf{X}'|} \sum_{X \in \mathbf{X}'} h(X)$$

- Main use of resampling is in sequential Importance Sampling schemes to counter degeneracy

Section 3

Adaptive Monte Carlo

Adaptive MCMC

- simplest case: fit covariance matrix C to previous samples, used in Gaussian proposal for next point (Haario et al., 2001)
- adaptive proposals in MCMC will be ergodic wrt target π in the general case as long as (Roberts and Rosenthal, 2007)
  - all Markov kernels used are ergodic wrt π (i.e. draw approximately from π)
  - diminishing adaptation, e.g. adaptation probability p_t at iteration t satisfies $p_t \to 0$ as $t \to \infty$
- we still can have $\sum_{t=1}^{N} p_t \to \infty$ as $N \to \infty$, i.e. infinite overall adaptation (cf. Andrieu and Thoms, 2008)
- conditions are sufficient but not necessary (other schemes might still be ergodic)

Adaptive Importance Sampling: Population Monte Carlo

- Population Monte Carlo (PMC; Cappé et al., 2004)
  - special case of SMC which is very simple coming from an MCMC background
  - improves proposal distributions q_t over iterations indexed by t by adapting to samples from previous iterations
  - resulting estimate is consistent by the law of large numbers as

$$\begin{aligned}
\mathbb{E}\left[w_t(X)h(X)\right] &= \iint \frac{\pi(x)}{q_t(x)} h(x)\ \mathrm{d}q_t(x)\ \mathrm{d}g(q_t) \\
&= \iint \pi(x)h(x)\ \mathrm{d}x\ \mathrm{d}g(q_t) \\
&= \int H\ \mathrm{d}g(q_t) \\
&= H
\end{aligned}$$

Population Monte Carlo Cappé et al. (2004)

Input: initial proposal density q_0, unnormalized density π, population size p, sample size m
Output: lists P, W of m samples and weights
Initialize $P = List()$
Initialize $W = List()$
while $len(P) \leq m$ **do**
  construct proposal distribution q_t
  generate set of p samples $\mathbf{X}_t$ from q_t and append it to P
    for all $X \in \mathbf{X}_t$ append weights $\pi(X)/q_t(X)$ to W
end while

- proposals q_t must not degenerate to distributions with thinner tails than π
- diminishing adaptation not a requirement, unlike in the adaptive MCMC theorems for the general case
- can adapt any which way we like without need to proof ergodicity
- estimate of evidence of our model (marginal likelihood, normalizing constant) at any time
- using randomized Low Discrepancy point sets, we can improve convergence rates more easily than in Metropolis-Hastings

Section 4

Gradient Importance Sampling

- Gradient IS updates a covariance fit C_{t+1} with posterior samples from iteration t
- resample weighted $\mathbf{X}_t$ to get unweighted $\mathbf{X}'_t$
- Proposal distribution at iteration $t+1$ is mixture

$$q_t(\cdot) = \sum_{X \in \mathbf{X}'_t} \mathcal{N}\left(\cdot | X + \delta C_t \nabla \log \pi(X), C_t\right)$$

  from which we draw $p = |\mathbf{X}'_t| = |\mathbf{X}_t| = \cdots = |\mathbf{X}_1|$ samples
- where $0 \leq \delta \leq 0.5$ is a parameter
- mixture components are closely related to discretized Langevin diffusion
- strong variance reductions through stratification in resampling and sampling from q_t

- Covariance C_t updated similar to Haario et al. (2001)
- define initial covariance matrix C_0 and iteration threshold t_0 at which adaptation starts, then

$$C_t = \begin{cases} C_0 & t \leq t_0 \\ s_d(\mathrm{Cov}(X_0, \ldots, X_{t-1}) + \epsilon I) & t > t_0 \end{cases}$$

- and the online fitting is achieved by the recursion

$$C_{t+1} = \frac{t-1}{t} C_t + \frac{s_d}{t} \left(t\bar{X}_{t-1}\bar{X}_{t-1}^T - (t+1)\bar{X}_t\bar{X}_t^T + X_t X_t^T + \epsilon I \right)$$

- where $\bar{X}_{t'}$ is the mean at time t'

Section 5

Evaluation

Algorithms

- Gradient IS experiments, a population size of $p = 10$ is used
- Compare to
  - HMC (parameters tuned to give optimal acceptance rate of 0.65)
  - adaptive Random Walk (adapt. RW; Haario et al., 2001)
  - adaptive Metropolis adjusted Langevin with truncation (adaptive MALTA; Atchadé, 2006)
- for covariance fit $s_d = 1$, $\epsilon = 2$ and $t_0 = 10$ used for all adaptive algorithms
- Drift factor δ tuned to give lowest variance across simulations for adaptive algorithms

Remarks on adaptive MCMC

- proposal density for Adaptive Metropolis is

$$\mathcal{N}(\cdot|X', C_t)$$

  with MH correction instead of importance weighting (AM; Haario et al., 2001)
- adaptive MALTA closest to our algorithm, but using

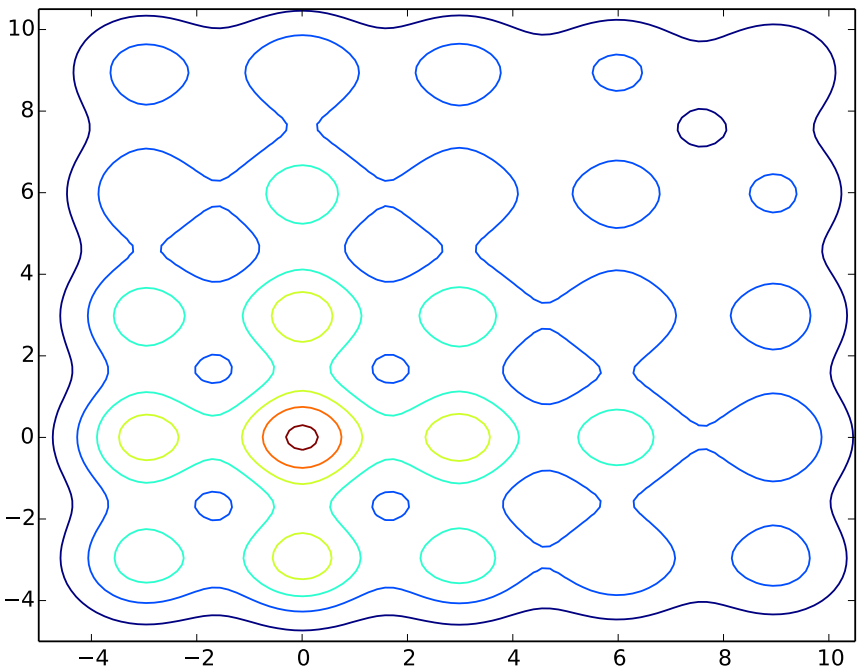
$$\mathcal{N}\left(\cdot|X + \delta C_t \nabla \log \pi(X), C_t\right)$$

  with MH correction and a different fitting procedure for C_t (which also adapts s_d)

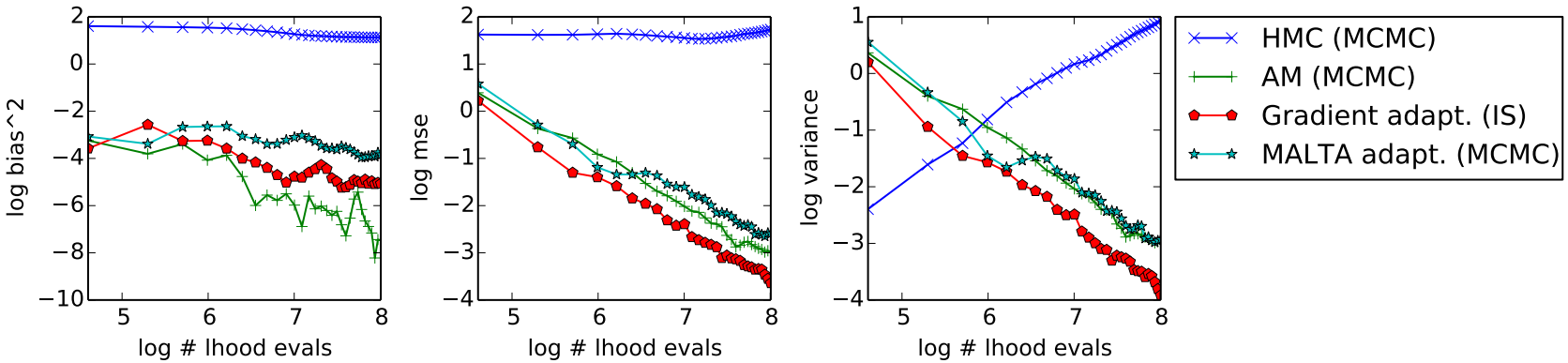
Evaluation framework

- three challenging synthetic distributions and one Bayesian posterior used as targets π
- $H_{id} = \int x \mathrm{d}\pi(x)$ known in closed form for synthetic targets
- start simulation at H_{id} (in order to not have to deal with burn-in for MCMC)
- run 20 Monte Carlo simulations for 3000 samples
- measure MSE, squared bias and variance across 20 runs

Gaussian Grid Target

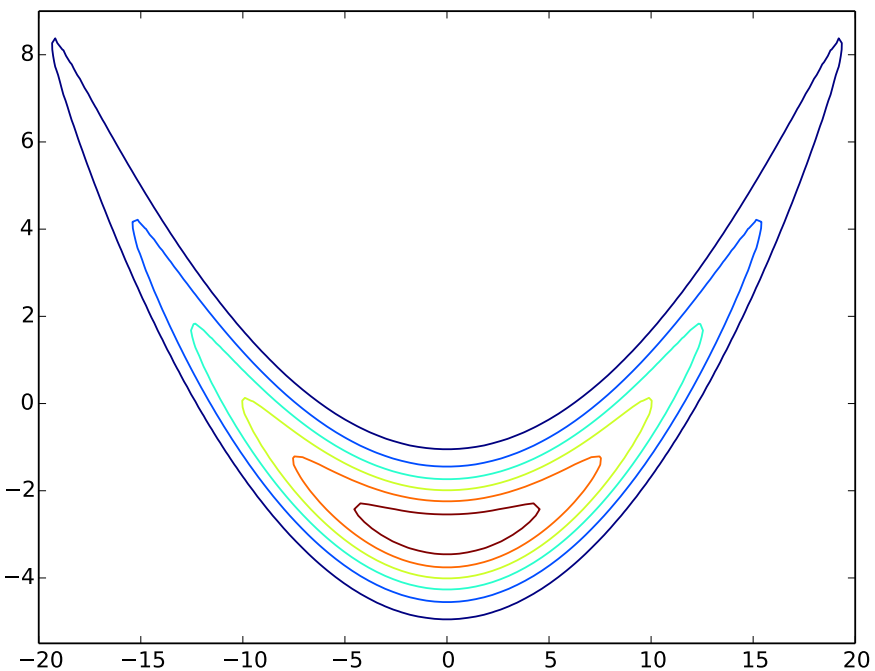


- Mixture of 25 2D Gaussians
- equidistant means and varying mixture weights

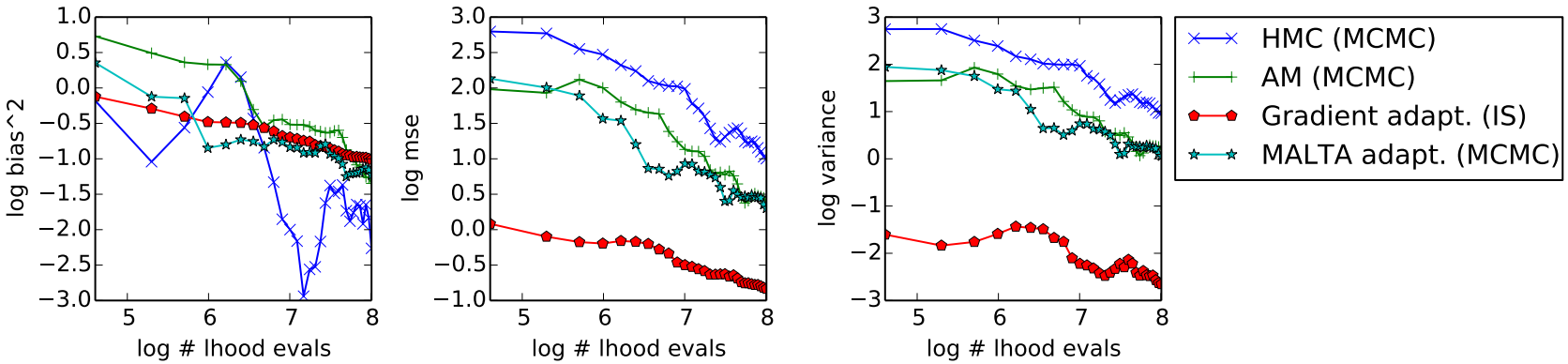


- performance is very close for adaptive algorithms
- adapt. Gradient IS exhibits smallest MSE and variance
- HMC presumably gets stuck in different modes for different runs, explaining high variance
- covariance adaptivity seems to ameliorate possible problems when using gradient information in sampling multimodal targets

Banana Target



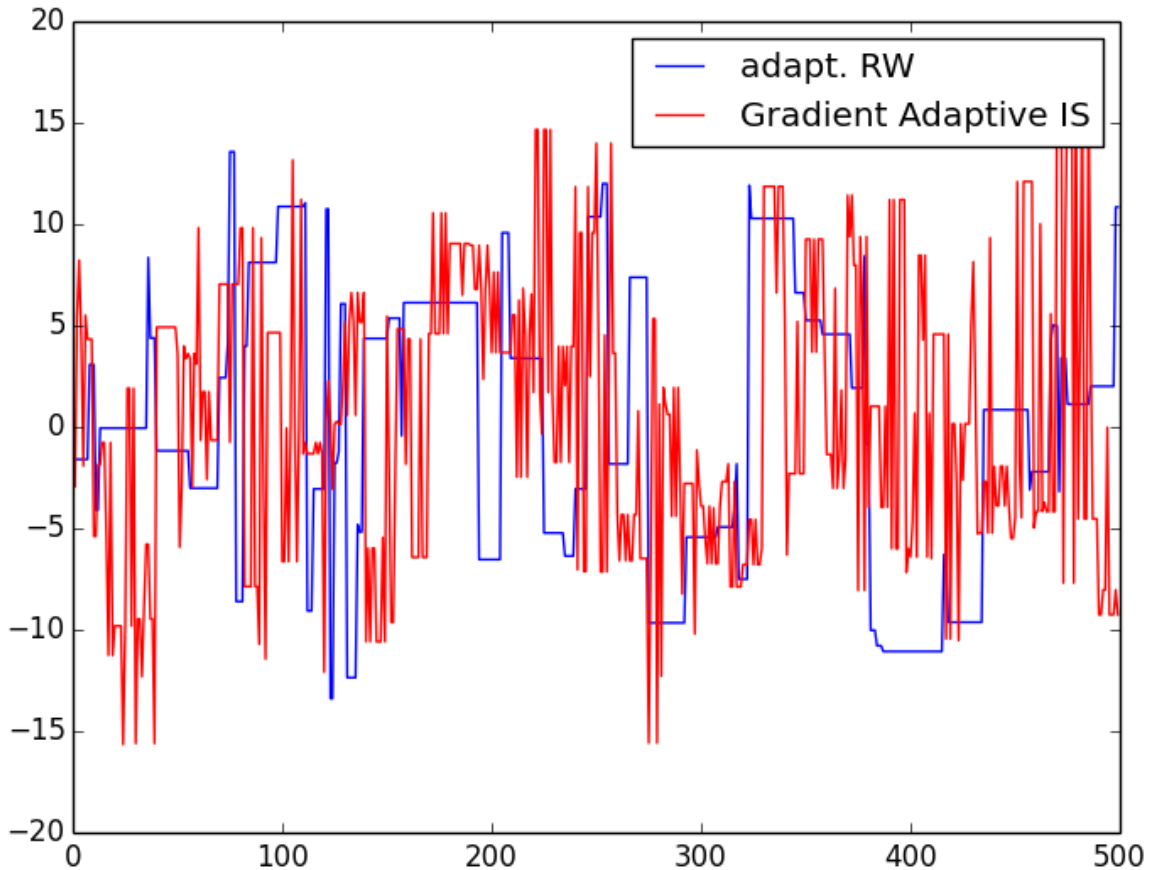
- MVN with transformed second dimension $y_2 = x_2 + b(x_1^2 - v)$
- Target from Haario et al. (2001)

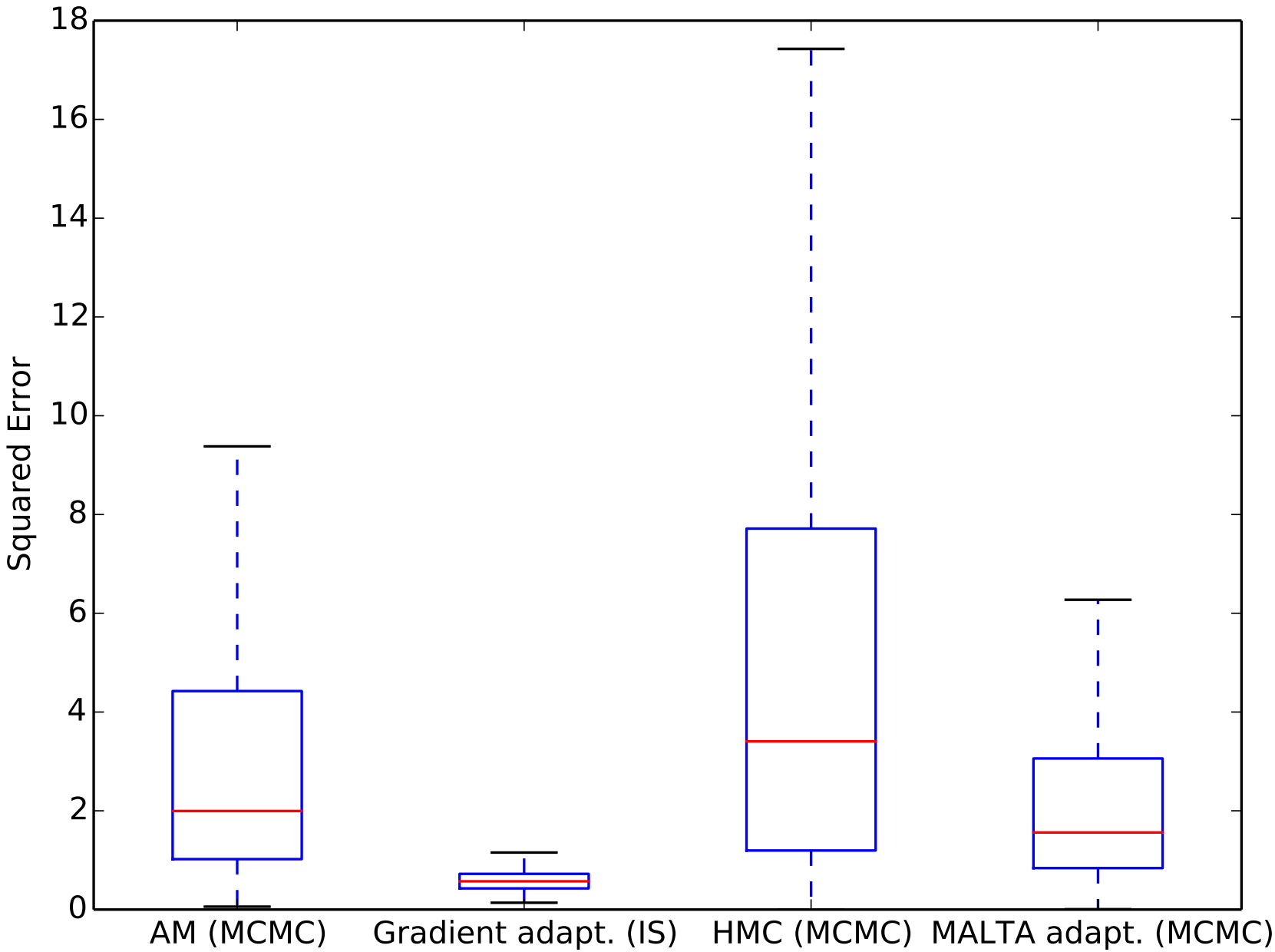


- GRIS shows clearest gains here
- trace plot of first coordinate (next slide) suggests that Adaptive Metropolis steps are not accepted very often
- Especially performance of adaptive MALTA is surprising, given that it adapts almost every parameter

Visualizing the first coordinate

Location of first coordinate for last 500 posterior samples




Squared Error
0
2
4
6
8
10
12
14
16
18
AM (MCMC)
Gradient adapt. (IS)
HMC (MCMC)
MALTA adapt. (MCMC)

Trimodal T Mixture

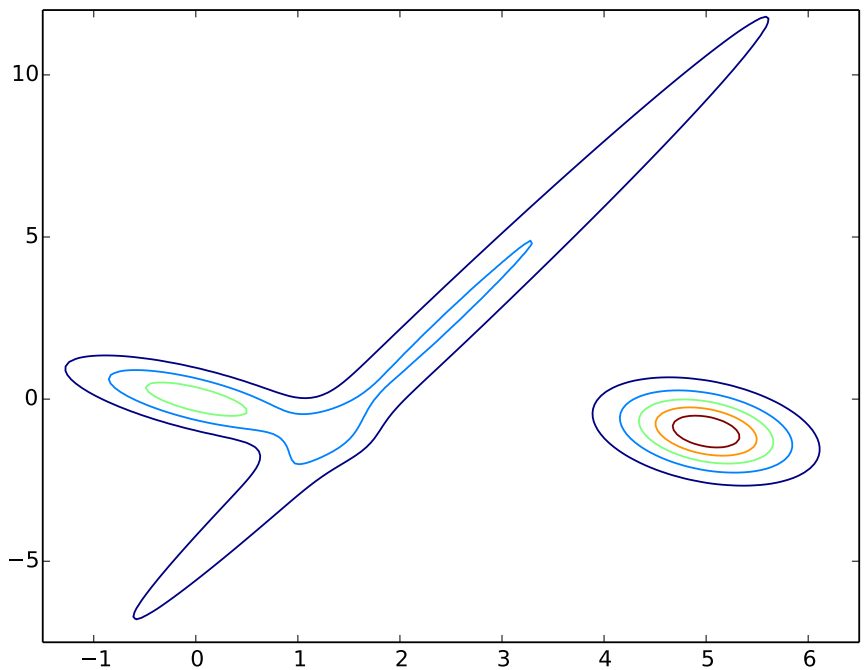
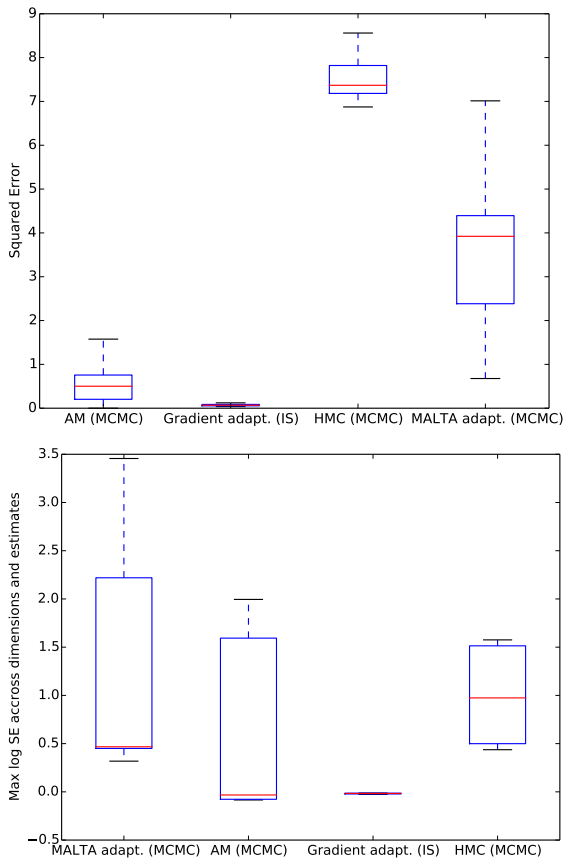


Figure: Last two dimensions

- Mixture of three 10D Student t distributions with
  - different means
  - 10 degrees of freedom (i.e. heavy tails)
  - scale matrices drawn from Inverse Wishart with scale I and 10 degrees of freedom (strong correlations between dimensions)

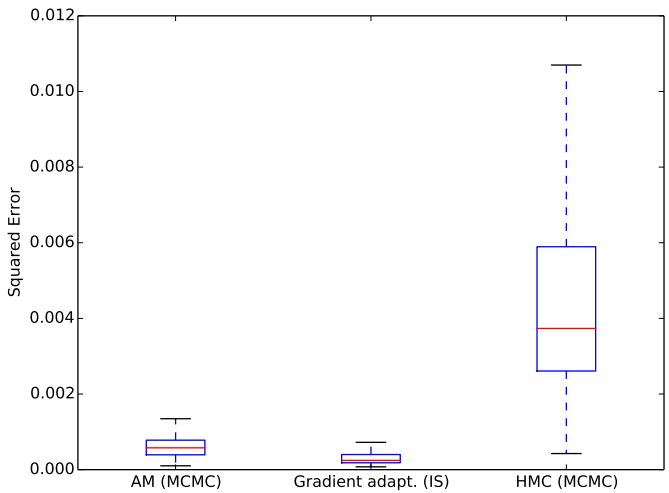
Trimodal T Mixture

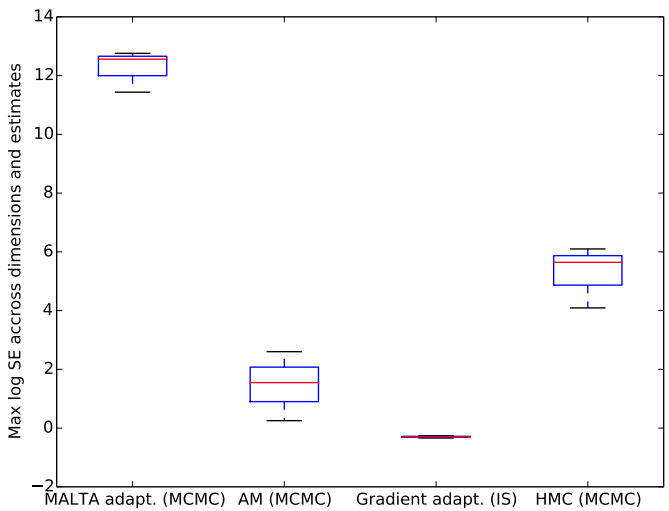


- Again, AM very good algorithm with few tuning parameters
- as Effective Sample Size (ironically) hard to measure for GRIS, instead
  - Measure SE of estimate for mean and variance of every dimension, take maximum
  - take maximum across dimensions

  resulting in a worst-case measure

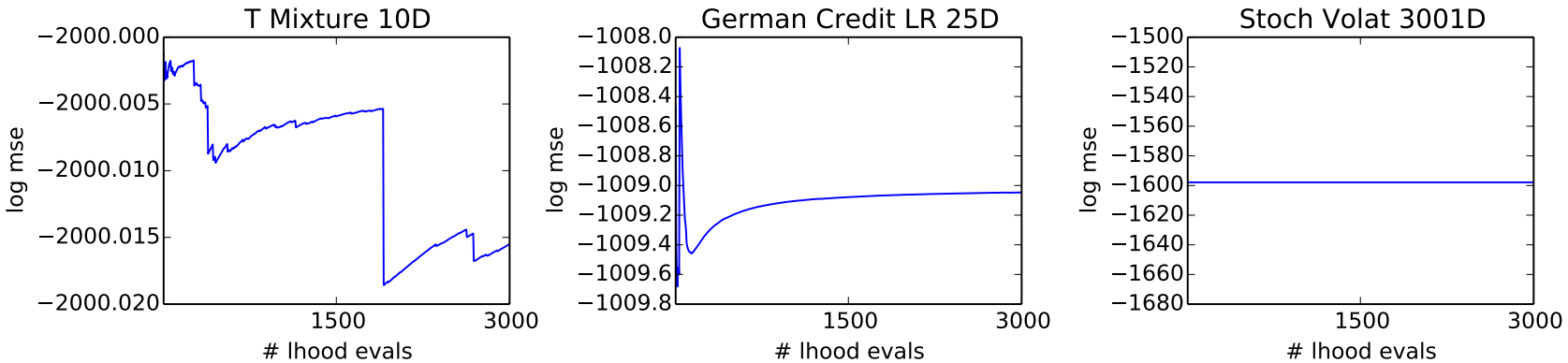
Logistic Regression





- Logistic Regression Problem with German Credit dataset (Hoffman and Gelman, 2014)
  - 24 predictor variables, 1 classification outcome (creditworthy or not)
  - 1000 data points
- Especially MALTA performance is surprising, given it adapts almost every parameter

Evidence estimation



- evidence estimates are stable, especially if target is peaky (i.e. many data points)
- log ground truth
  - -1000 for t Mixture
  - -504 for Logistic Regression
  - -799 for Stochastic Volatility

Section 6

Discussion

Contributions

- First gradient informed sampling algorithm based on an iterated importance sampling scheme
- Extremely simple to implement (as compared to adaptive MALTA and HMC)
- Few and very easily tuned parameters
- automatic adaptation using Bayesian Optimization straight forward
- ability to represent modes using population
- stable evidence estimates

Outlook

- Getting rid of resampling might be beneficial in high dimensions
- sample from Unadjusted Langevin Algorithm (ULA)

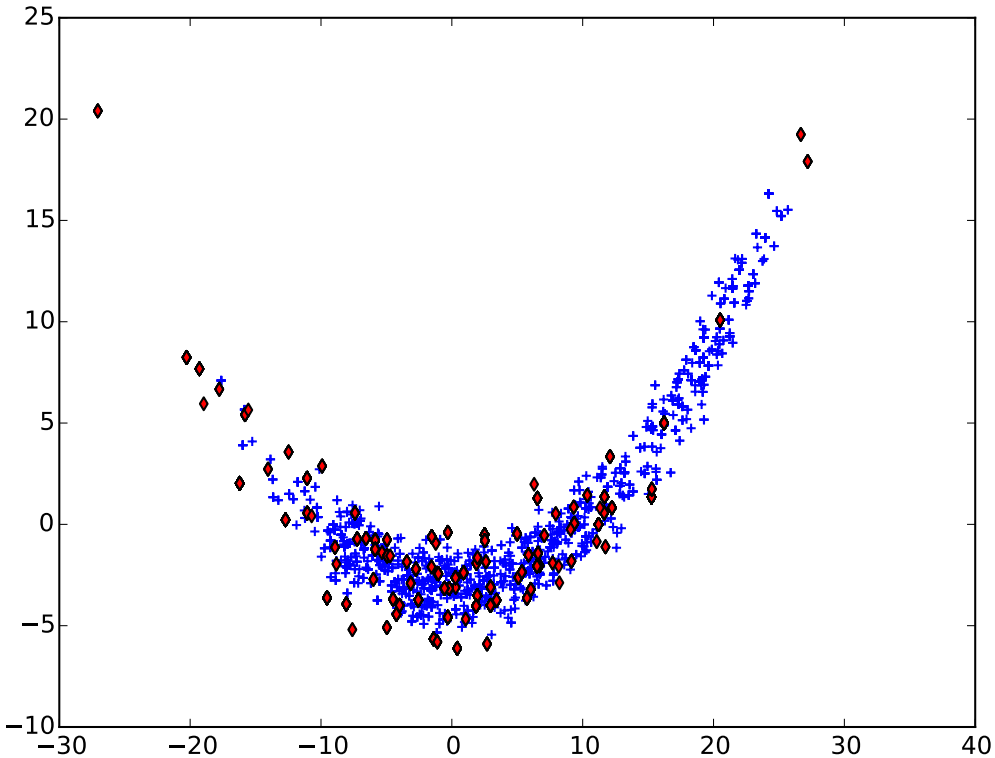
$$X_{t+1} \sim \mathcal{N}\left(\cdot | X_t + \delta \ \nabla \log \ \pi(X_t), \sigma^2 I\right)$$

and importance weight with Rao-Blackwellized proposal

$$w(X) = \frac{\pi(X)}{\sum_t \mathcal{N}\left(X | X_t + \delta \ \nabla \log \ \pi(X_t), \sigma^2 I\right)}$$

- experimentally strong gains over MALA
- not sure how to prove consistency
- two parallel ULAs might help

Unadjusted Langevin with RB Importance Weighting



- Unadjusted Langevin with RB Importance Weighting (red diamonds) vs. MALA (blue crosses)
- improve RMSE of estimate of first four moments on average

Thanks!

Literature I

Andrieu, C. and Thoms, J. (2008). A tutorial on adaptive MCMC. *Statistics and Computing*, 18(November):343–373.

Atchadé, Y. F. (2006). An adaptive version for the metropolis adjusted Langevin algorithm with a truncated drift. *Methodology and Computing in Applied Probability*, 8(March):235–254.

Cappé, O., Guillin, a., Marin, J. M., and Robert, C. P. (2004). Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929.

Haario, H., Saksman, E., and Tamminen, J. (2001). An Adaptive Metropolis Algorithm. *Bernoulli*, 7(2):223–242.

Hoffman, M. and Gelman, A. (2014). The No-U-Turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:1351–1381.

Literature II

Robert, C. P. and Casella, G. (2004). *Monte Carlo statistical methods*. Springer, 2nd edition.

Roberts, G. O. and Rosenthal, J. S. (2007). Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of Applied Probability*, 44(2):458–475.