
An Efficient Minibatch Acceptance Test for Metropolis-Hastings

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Markov chain Monte Carlo (MCMC) methods have many applications in machine
2 learning. We are particularly interested in their application to modeling very
3 large datasets, where it is impractical to perform Metropolis-Hastings tests on the
4 full data. Previous work on reducing the cost of Metropolis-Hastings tests yield
5 variable data consumed per sample, with only constant factor reductions vs. using
6 the full dataset for each sample. Here we present a method that can be tuned
7 to provide arbitrarily small batch sizes, by adjusting either proposal step size or
8 temperature. Our approach uses the natural noise present in minibatch likelihood
9 estimates to furnish the randomness in a Metropolis-Hastings test. Our test uses the
10 noise-tolerant Barker acceptance test with a novel additive correction variable. The
11 resulting test can be combined with minibatch proposals to yield updates with the
12 same complexity as a simple SGD update. In this paper we derive the test, analyze
13 its performance, discuss its implementation, and present several experiments.

14

1 Introduction

15 Markov chain Monte Carlo (MCMC) sampling is a powerful method for computation on intractable
16 distributions. We are interested primarily in large dataset applications, where the goal is to sample
17 a posterior distribution $p(\theta | x_1, \dots, x_N)$ of parameter θ , and where the number of data instances
18 N is large. The Metroplis-Hastings method (M-H) generates sample candidates from a proposal
19 distribution q which is in general different from the target distribution p , and decides whether to accept
20 or reject them based on an acceptance test. The acceptance test is usually a Metropolis test [1, 2].
21 Conventional Metropolis-Hastings requires all N data instances to generate one posterior sample.

22 Many state-of-the-art machine learning methods, and deep learning in particular, are based on
23 minibatch updates (such as SGD) to a model. Minibatch updates produce many improvements to
24 the model for each pass over the dataset, and have high sample efficiency. They also map very well
25 onto hardware such as GPUs. In contrast, M-H requires calculations over the full dataset to produce a
26 new sample. Recent results from [3, 4] perform approximate (bounded error) acceptance tests using
27 subsets (minibatches) of the full dataset. The tests depend on minibatch statistics, and on the value
28 of an additional random variable u . The amount of data consumed for each test varies significantly
29 from one minibatch to the next, and depends on the current sample, the proposed sample, and on the
30 random variable u . By contrast, [5] performs exact tests but requires a lower bound on parameter
31 distribution across its domain. The amount of data reduction depends on the accuracy of this bound,
32 and such bounds are only available for relatively simple distributions.

33 Here we derive a new test which incorporates the variability in minibatch statistics as *a natural part*
34 *of the test*. Because of this, the amount of data required for each test is fixed with high probability
35 and in expected value. We use a Barker test function [6] rather than a Metropolis test, which makes
36 the test naturally error tolerant. The idea of using a noise-tolerant test using Barker's test function

37 were suggested but not explored empirically in [7] section 6.3. But the asymptotic test statistic CDF
38 and the Barker function are different, which leads to fixed errors for the approach in [7]. Here we
39 show that the difference between the distributions can be corrected with an additive random variable.
40 This leads to a test which is fast, and whose error can be made arbitrarily small.

41 Our test is applicable when the variance (over data samples) of the log acceptance probability is small
42 enough (less than 1). Its not clear at first why this quantity should be so bounded. But we will see that
43 it is “natural” for well-specified models running Metropolis-Hastings sampling with optimal proposals
44 [8] on a full dataset. When we reduce the amount of data for the test, the variance goes up, and we
45 have to reduce it in one of several ways. Either:

- 46 • Increase the temperature of the target distribution. Log likelihoods scale as $1/T$, and so the
47 variance of the likelihood ratio will vary as $1/T^2$. Our model is no longer well-specified (we
48 are doing inference at a temperature different from that assumed during data generation),
49 but higher temperature can be advantageous for parameter exploration.
- 50 • For continuous probability distributions, reduce the proposal step size and variance (for
51 stochastic proposals) compared to an optimal proposal. The variance of the log acceptance
52 probability scales as the square of proposal step size.
- 53 • Increase the minibatch size. log acceptance variance scales as $1/k$ vs the minibatch size k .
54 Increased minibatch size also reduces the error rate for the test.

55 Its worth discussing at this point what are the goals (typically) of M-H sampling on very large
56 datasets. By the Bernstein-von Mises theorem, the posterior distribution of the parameter θ for a
57 Bayesian inference task is asymptotically normal, and has variance that scales inversely with the
58 number of data samples N . This mode is extremely sharp for large datasets, which may contain
59 millions or billions of samples. Simply sampling from this distribution is one application, but an
60 efficient proposal distribution [8] has similar variance to the target distribution and will diffuse to it
61 extremely slowly from an initialization value which is (likely to be) many standard deviations away.
62 If there are any other strong modes, it is very likely for the sampler to find one of them and become
63 trapped in it when run at the normal distribution temperature ($T=1$). A common solution is to anneal
64 the sampler, running first at high temperature (scaling log likelihoods by $1/T$) which flattens the
65 likelihood landscape. This in turn reduces the variance of the log acceptance probability and allows
66 our acceptance test to be applied.

67 A second question concerns step size. Once we have fixed temperature, our variance constraint
68 implies that we have to trade-off proposal step size s and batch size b ($b \propto p^2$). i.e. we can make
69 many small steps, or one large step, with a given batch of data. One of the primary drivers of this work
70 is our belief in the value of small steps. For applications to neural networks or other models where the
71 posterior is multimodal, posterior inference is arguably a search process. Covering the search space
72 densely with small steps is much more valuable than few sparse steps toward the nearest optimum. In
73 this mode, Metropolis-Hastings can be used in similar fashion to Stochastic Gradient Descent. The
74 goal in SGD is to make gradual progress to a posterior mode with each step, taking small steps so that
75 the cumulative displacement has progressively lower variance. A substantial part of the computational
76 work of MCMC on massive datasets will be similarly in reaching a stationary distribution, which
77 really means finding a deep posterior mode. Taking noisy small steps will nevertheless make steady
78 progress to a posterior mode since their bias is in that direction. We demonstrate this behavior with
79 our experiments.

80 The contributions of this paper are as follows:

- 81 • We develop a new, more efficient (in samples per test) minibatch acceptance test with
82 quantifiable error bounds. The test uses a novel additive correction variable to implement a
83 Barker test based on minibatch mean and variance.
- 84 • We analyze the test for accuracy and speed.
- 85 • We compare performance of our new test and prior approaches on several datasets. We
86 demonstrate the test is several orders of magnitude more efficient than prior work measured
87 as data consumed per test, and that it does not suffer from long-tailed batch sizes (up to the
88 dataset size).

89 **2 Preliminaries and Related Work**

90 In the Metropolis-Hastings method [9, 10], a difficult-to-compute probability distribution $p(\theta)$ is
 91 sampled using a Markov chain $\theta_1, \dots, \theta_n$. The sample θ_{t+1} at time $t + 1$ is generated using a
 92 candidate θ' from a (simpler) proposal distribution $q(\theta' | \theta_t)$, filtered by an acceptance test. The
 93 acceptance test is usually a Metropolis test. The Metropolis test has acceptance probability:

$$\alpha(\theta_t, \theta') = \frac{p(\theta')q(\theta_t | \theta')}{p(\theta_t)q(\theta' | \theta_t)} \wedge 1 \quad (1)$$

94 where $a \wedge b$ denotes $\min(a, b)$. With probability $\alpha(\theta_t, \theta')$, we accept θ' and set $\theta_{t+1} = \theta'$, otherwise
 95 set $\theta_{t+1} = \theta_t$. The test is often implemented with an auxiliary random variable $u \sim \mathcal{U}(0, 1)$ with
 96 a comparison $u < \alpha(\theta_t, \theta')$. $\mathcal{U}(a, b)$ denotes the uniform distribution on the interval $[a, b]$. For
 97 notational simplicity, from now on we will drop the subscript t for the current sample θ_t and denote
 98 it simply as θ .

99 The acceptance test guarantees detailed balance, which means

$$p(\theta)p(\theta' | \theta) = p(\theta')p(\theta | \theta') \quad (2)$$

100 where $p(\theta' | \theta)$ is the probability of a transition from state θ to θ' . Here $p(\theta' | \theta) = q(\theta' | \theta)\alpha(\theta, \theta')$
 101 so the detailed balance equation becomes

$$p(\theta)q(\theta' | \theta)\alpha(\theta, \theta') = p(\theta')q(\theta | \theta')\alpha(\theta', \theta) \quad (3)$$

102 The detailed balance condition, together with ergodicity, guarantee that the Markov chain has a
 103 unique stationary distribution $\pi(\theta) = p(\theta)$.

104 For Bayesian inference, we would like to sample from a parameter distribution for θ based on
 105 some observed data x_1, \dots, x_N . i.e. we want to sample from $p(\theta | x_1, \dots, x_N)$. The acceptance
 106 probability is now:

$$\alpha(\theta, \theta') = \frac{p_0(\theta') \prod_{i=1}^N p(x_i | \theta') q(\theta | \theta')}{p_0(\theta) \prod_{i=1}^N p(x_i | \theta) q(\theta' | \theta)} \wedge 1 \quad (4)$$

107 where $p_0(\theta)$ is a prior, and $p(x_i | \theta)$ are the probabilities of the observations. Computing samples
 108 this way requires the use of all N training data points, but this is very expensive for large datasets.
 109 To address this challenge, [3, 4] perform approximate Metropolis-Hastings tests using sequential
 110 hypothesis testing. During each iteration, they start with a small minibatch of data and test the
 111 hypothesis that the sample θ' based on an approximate version of the test $u < \alpha(\theta, \theta')$. If the
 112 approximate test cannot make a decision with sufficient confidence, then the minibatch size is
 113 increased and the test repeats. This process continues until a decision. The bounds depend on either a
 114 asymptotic central limit theorem [3] or a concentration bound [4], the latter requiring direct bounds on
 115 the log likelihood ratio. The problem with both methods is the expense of resolving small differences.
 116 In the worst case, all N data points may be needed, and as we show next this worst case can occur
 117 about $\Omega(N)$ times during the performance of N tests.

118 Following [4], we write the test $u < \alpha(\theta, \theta')$ in the equivalent form $\Lambda(\theta, \theta') > \psi(u, \theta, \theta')$. Our
 119 definitions differ from those in [4] by a factor of N to simplify our analysis later on.

$$\Lambda(\theta, \theta') = \sum_{i=1}^N \log \left(\frac{p(x_i | \theta')}{p(x_i | \theta)} \right) \quad \text{and} \quad \psi(u, \theta, \theta') = \log \left(u \frac{q(\theta' | \theta)p_0(\theta)}{q(\theta | \theta')p_0(\theta')} \right) \quad (5)$$

120 To reduce computational effort, an unbiased estimate of $\Lambda(\theta, \theta')$ based on a minibatch can be used:

$$\Lambda^*(\theta, \theta') = \frac{N}{b} \sum_{i=1}^b \log \left(\frac{p(x_i | \theta')}{p(x_i | \theta)} \right) \quad (6)$$

121 Finally, it will be convenient in what follows to define the individual components that contribute to
 122 the sums above:

$$\Lambda_i(\theta, \theta') = N \log \left(\frac{p(x_i | \theta')}{p(x_i | \theta)} \right) \quad (7)$$

123 So $\Lambda(\theta, \theta')$ is the mean of $\Lambda_i(\theta, \theta')$ over the entire dataset, and $\Lambda^*(\theta, \theta')$ is the mean over the
 124 minibatch of $\Lambda_i(\theta, \theta')$.

125 Since the minibatch contains randomly selected samples x_i , the values Λ_i are independent, IID
 126 random variables. By the central limit theorem, we expect $\Lambda^*(\theta, \theta')$ to be approximately Normal.
 127 The acceptance test then becomes a statistical test of the hypothesis that $\Lambda(\theta, \theta') > \psi(u, \theta, \theta')$ by
 128 establishing that $\Lambda^*(\theta, \theta')$ is substantially larger than $\psi(u, \theta, \theta')$. In [3] an asymptotic central limit
 129 argument was used to derive this gap, while in [4] a concentration bound was used. In both cases, the
 130 resulting tests were shown to give useful reductions in number of samples required over using the full
 131 dataset. But no worst-case bounds were given on average number of samples needed.

132 We next show that for some simple distributions, the lower bound of average number of data instances
 133 consumed for one iteration of [3] and [4] is $\Omega(N)$, where N is the number of data points.

134 2.1 An Example

135 Consider a simple model where samples x_i for $i = 1, \dots, N$ are drawn from a Normal distribution
 136 $\mathcal{N}(\theta, 1)$ where $\theta = 0.5$, and with a uniform prior on θ . The log likelihood ratio is

$$\Lambda^*(\theta, \theta') = \frac{N}{b} \sum_{i=1}^b \log \frac{p(x_i | \theta')}{p(x_i | \theta)} = N(\theta' - \theta) \left(\frac{1}{b} \sum_{i=1}^b x_i - \theta - \frac{\theta' - \theta}{2} \right) \quad (8)$$

137 which is normally distributed over selection of the Normal samples x_i . Since the x_i have unit variance,
 138 their mean has variance $1/b$, and the variance $\sigma^2(\Lambda^*)$ of $\Lambda^*(\theta, \theta')$ is $(\theta' - \theta)^2 N/b$. In order to pass
 139 a hypothesis test that $\Lambda > \psi$, we will need to establish a large enough gap (several $\sigma(\Lambda^*)$) between
 140 $\Lambda^*(\theta', \theta)$ and $\psi(u, \theta', \theta)$.

141 We would like to sample from the posterior distribution of θ which is a normal distribution centered
 142 on the sample mean, and with variance $1/N$. The target distribution is therefore $\mathcal{N}(\mu, 1/N)$ where
 143 μ is the sample mean, and $\mathcal{N}(\mu, \sigma^2)$ is a normal distribution with mean μ and variance σ^2 . An
 144 efficient proposal distribution has the same variance as the target distribution in one dimension [8],
 145 so take the proposal $\mathcal{N}(\theta' - \theta, 1/N)$, which is implemented as $q(\theta' | \theta) = \phi((\theta' - \theta)/N)$, where
 146 $\phi(x)$ is the Normal density function with zero mean and unit variance. This proposal is symmetric
 147 $q(\theta' | \theta) = q(\theta | \theta')$, and we assumed uniform prior for θ , so $\psi(u, \theta', \theta)$ reduces to $\log u$.

148 **Lemma 1.** *For the model above, there exists a fixed (independent of N) constant c such that with
 149 probability $\geq c$ over the joint distribution of (θ, θ', u) , the tests from [4] and [3] consume all N
 150 samples.*

151 Proof is given in the appendix.

152 Similar results can be proved for other distributions and proposals, by identifying regions in product
 153 space $(\theta, \theta' - \theta, u)$ such that the hypothesis test needs to separate nearly-equal values.

154 It follows that the accelerated M-H tests in [4] and [3] require at least a constant fraction $\geq c$ in the
 155 amount of data consumed per test, compared to full-dataset tests. i.e. their speed-up is at most $1/c$.

156 2.2 MCMC Posterior Inference

157 There is a separate line of MCMC work drawing principles from statistical physics. By viewing
 158 random variables as particles in a system, one can apply Hamiltonian Monte Carlo (HMC) [11]
 159 methods which generate high acceptance *and* distant proposals when run on full batches of data.
 160 Recently Langevin Dynamics [12, 13] has been applied to Bayesian estimation on minibatches of data.
 161 This simplified dynamics uses local proposals and avoids MH tests by using small proposal steps
 162 whose acceptance approaches 1 in the limit. However, the constraint on proposal step size is severe,
 163 and the state space exploration reduces to a random walk. Full minibatch HMC for minibatches
 164 was recently described in [14] which allows momentum-augmented proposals with larger step sizes.
 165 However, step sizes are still limited by the need to run accurately without MH tests. Our work opens
 166 the door to applying those methods with much more aggressive step sizes without loss of accuracy.
 167 We demonstrate this in Section ??.

168 **3 A New Metropolis-Hastings Acceptance Test**

169 For our new M-H test, we denote the exact and approximate log likelihood ratios as Δ and Δ^* . First,
 170 Δ is defined as:

$$\Delta(\theta, \theta') = \log \frac{p(\theta')q(\theta | \theta')}{p(\theta)q(\theta' | \theta)} \quad (9)$$

171 and in the Bayesian case

$$\Delta(\theta, \theta') = \log \frac{p_0(\theta') \prod_{i=1}^N p(x_i | \theta') q(\theta | \theta')}{p_0(\theta) \prod_{i=1}^N p(x_i | \theta) q(\theta' | \theta)} \quad (10)$$

172 and we can separate out terms dependent and independent of the data x as:

$$\Delta(\theta, \theta') = \sum_{i=1}^N \log \frac{p(x_i | \theta')}{p(x_i | \theta)} - \psi(1, \theta, \theta') = \Lambda(\theta, \theta') - \psi(1, \theta, \theta') \quad (11)$$

173 A minibatch estimator of Δ can be defined as:

$$\Delta^*(\theta, \theta') = \frac{N}{b} \sum_{i=1}^b \log \frac{p(x_i | \theta')}{p(x_i | \theta)} - \psi(1, \theta, \theta') = \Lambda^*(\theta, \theta') - \psi(1, \theta, \theta') \quad (12)$$

174 Note that Δ and Δ^* are evaluated on the full dataset and a minibatch of size b respectively. The
 175 scaling term $\frac{N}{b}$ ensures that $\Delta^*(\theta, \theta')$ is an unbiased estimator of $\Delta(\theta, \theta')$.

176 **3.1 A Full Dataset Acceptance Test**

177 The key to our test is a smooth acceptance function. We consider tests other than the Metropolis
 178 test that satisfy the conditions needed for accurate posterior estimation. This condition is the
 179 detailed balance condition defined earlier. Expressed in terms of Δ , the classical Metropolis test is
 180 $f(\Delta) = \min(\exp(\Delta), 1)$. More generally we have:

181 **Lemma 2.** *If $g(s)$ is any function such that $g(s) = \exp(s)g(-s)$, then the acceptance function
 182 $\alpha(\theta, \theta') \triangleq g(\Delta(\theta, \theta'))$ satisfies detailed balance. That is,*

$$p(\theta)q(\theta' | \theta)\alpha(\theta, \theta') = p(\theta')q(\theta | \theta')\alpha(\theta', \theta) \quad (13)$$

183 This result is used in [6] to define the Barker acceptance test. As a sanity check, choosing $g(s) =$
 184 $\exp(s) \wedge 1$ produces the standard Metropolis acceptance test $\alpha(\theta, \theta') = g(\Delta(\theta, \theta')) = \frac{p(\theta')q(\theta|\theta')}{p(\theta)q(\theta'|\theta)} \wedge 1$.
 185 $g(s)$ satisfies the condition $g(s) = \exp(s)g(-s)$, and therefore detailed balance. In fact it is the
 186 optimal acceptance function in terms of acceptance rate, since it accepts with probability 1 for $\Delta > 0$.

187 **3.2 Barker (Logistic) Acceptance Function**

188 For our new MH test we use the Barker logistic function: $g(s) = (1 + \exp(-s))^{-1}$ [6]. Straightfor-
 189 ward arithmetic shows that it satisfies the condition in Lemma 2. While it is slightly less efficient
 190 than the Metropolis test when used on the full dataset, we will see soon that its smoothness allows it
 191 to naturally tolerate substantial variance in its input argument. This in turn will lead to a much more
 192 efficient test on subsets of data.

193 Assume we begin with the current sample θ and a candidate sample θ' . Let V be a uniform random
 194 variable $V \sim \mathcal{U}(0, 1)$. We accept θ' if $g(\Delta(\theta, \theta')) > V$, and reject otherwise. Since $g(s)$ is
 195 monotonically increasing, its inverse $g^{-1}(s)$ is well-defined and unique. So an equivalent test is to
 196 accept θ' iff

$$\Delta(\theta, \theta') > X = g^{-1}(V) \quad (14)$$

197 where X is a random variable with the logistic distribution (its CDF is the logistic function). To see
 198 this notice that $\frac{dV}{dX} = g'$, that g' is the density corresponding to a logistic CDF, and finally that $\frac{dV}{dX}$ is
 199 the density of X . The density of X is symmetric, so we can equivalently test whether

$$\Delta(\theta, \theta') + X > 0 \quad (15)$$

200 for a logistic random X .

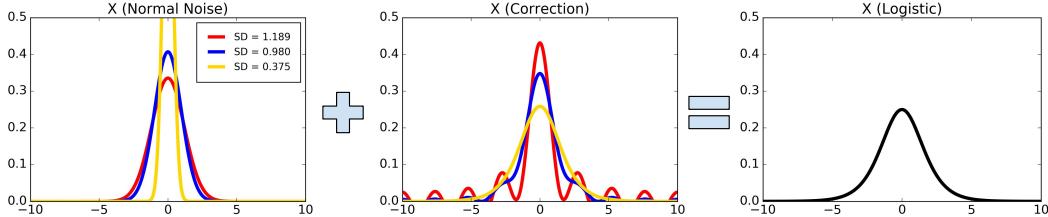


Figure 1: Three examples of X_{norm} and X_{corr} distributions that convolve to form the standard logistic distribution. We use three standard deviation values of X_{norm} . The two red curves convolve to form the logistic, etc. The y -axis is capped at 0.5 for readability. This figure must be viewed in color.

201 3.3 A Minibatch Acceptance Test

202 We now move to acceptance testing using the minibatch estimator $\Delta^*(\theta, \theta')$. From equation (12),
 203 $\Delta^*(\theta, \theta')$ can be represented as a constant term plus the mean of b IID terms $\Lambda_i(\theta, \theta')$ of the form
 204 $N \log \frac{p(x_i|\theta')}{p(x_i|\theta)}$. As b increases, $\Delta^*(\theta, \theta')$ therefore has a distribution which approaches a normal
 205 distribution by the central limit theorem. Later we will provide specific bounds between the CDF of
 206 $\Delta^*(\theta, \theta')$ and a Normal CDF, but for now we use an asymptotic argument.

207 In the limit, since Δ^* is normally distributed about its mean Δ , we can write

$$\Delta^* = \Delta + X_{\text{norm}}, \quad X_{\text{norm}} \sim \bar{\mathcal{N}}(0, \sigma^2(\Delta^*)), \quad (16)$$

208 where $\bar{\mathcal{N}}(0, \sigma^2(\Delta^*))$ denotes a distribution which is approximately normal with variance $\sigma^2(\Delta^*)$.
 209 But to perform the test in equation (15) we want $\Delta + X$ for a logistic random X (call it X_{log} from
 210 now on). In [7] it was proposed to use Δ^* in a Barker test anyway and tolerate the fixed error caused
 211 by this approximation.

212 Our approach is to instead decompose X_{log} as

$$X_{\text{log}} = X_{\text{norm}} + X_{\text{corr}}, \quad (17)$$

213 where we assume X_{norm} has zero-mean, Normal distribution with variance σ^2 , and X_{corr} is a zero-
 214 mean “correction” variable with density $C_\sigma(X)$. The two variables are added (i.e., their distributions
 215 convolve) to form X_{log} . It will be convenient to express this relationship in terms of the CDFs
 216 $N_\sigma(X_{\text{norm}}) = \Phi(X_{\text{norm}}/\sigma)$ and $S(X_{\text{log}})$, where $\Phi(\cdot)$ is the Normal CDF with zero mean and unit
 217 variance and $S(\cdot)$ is the standard logistic function.

218 Equation (17) implies that $S = N_\sigma * C_\sigma$ where $*$ denotes convolution. We will later derive a
 219 numerical representation of C by deconvolution, and using this we can generate samples of X_{corr} .

220 The acceptance test is now:

$$\Delta + X_{\text{log}} = (\Delta + X_{\text{norm}}) + X_{\text{corr}} = \Delta^* + X_{\text{corr}} > 0. \quad (18)$$

221 i.e. given an estimate Δ^* from the current data minibatch (assuming its variance is small enough) we
 222 test acceptance by adding a random variable X_{corr} and then accept θ' if the result is positive.

223 If $\bar{\mathcal{N}}(0, \sigma^2(\Delta^*))$ is exactly $\mathcal{N}(0, \sigma^2(\Delta^*))$, the above test is exact as well. And as we will see later, if
 224 there is a maximum error ϵ between the CDF of $\bar{\mathcal{N}}(0, \sigma^2(\Delta^*))$ and the CDF of $\mathcal{N}(0, \sigma^2(\Delta^*))$, then
 225 the acceptance test has an error of at most ϵ .

226 Algorithm 1 describes the M-H iteration using our acceptance test.

227 4 Computing the Correction Distribution

228 Our goal is to compute the distribution of the correction variable X_{corr} such that $X_{\text{norm}} + X_{\text{corr}} =$
 229 X_{log} , where $X_{\text{norm}} \in \mathcal{N}(0, \sigma)$ and X_{log} has a standard logistic CDF $(1 + \exp(-X))^{-1}$. In the next
 230 section we show that the accuracy of the test depends on the absolute error between the CDFs of
 231 $X_{\text{norm}} + X_{\text{corr}}$ and the distribution of X_{log} . So we minimize this in our construction.

232 Our goal then is to minimize

$$|\Phi_\sigma * C_\sigma - S| \quad (19)$$

Input : Number of samples T , minibatch size m , error bound δ , pre-computed correction $C_1(X)$ distribution, initial sample θ_1 .
Output : A chain of T samples $\{\theta_1, \dots, \theta_T\}$ from $p(\theta)$;
for $t = \{1, \dots, T\}$ **do**
 | Propose a candidate θ' from proposal distribution $q(\theta' | \theta_t)$;
 | Draw a minibatch of m points x_i , compute $\Delta^*(\theta_t, \theta')$ and sample variance $\sigma^2(\Delta^*)$;
 | Estimate moments $E|\Lambda_i - \Lambda|$ and $E|\Lambda_i - \Lambda|^3$ from the sample, and error ϵ from equation (32);
 | **while** $\sigma^2(\Delta^*) \geq 1$ **or** $\epsilon > \delta$ **do**
 | | Draw m more samples to augment the minibatch, update Δ^* , $\sigma^2(\Delta^*)$ and ϵ estimates;
 | **end**
 | Draw $X_{\text{norm}} \in \mathcal{N}(0, 1 - \sigma^2(\Delta^*))$ and X_{corr} from the correction distribution $C_1(X)$;
 | **if** $\Delta^* + X_{\text{norm}} + X_{\text{corr}} > 0$ **then**
 | | Accept the candidate, $\theta_{t+1} = \theta'$;
 | **else**
 | | Reject and re-use the old sample, $\theta_{t+1} = \theta_t$;
 | **end**
end

Algorithm 1: A description of M-H sampling with our acceptance test.

233 where $\Phi_\sigma(X) = \Phi(X/\sigma)$ and Φ is the standard normal CDF, $S(X)$ is the logistic function, $C_\sigma(X)$
234 is the density of the correction distribution, and $*$ denotes convolution. In continuous form, this value
235 is

$$\sup_X \left| \int_{-V}^V \Phi_\sigma(X - Y) C_\sigma(Y) dY - S(X) \right| \quad (20)$$

236 Assuming X_{corr} lies in the interval $[-V, V]$. For computation of $C_\sigma(Y)$, we discretize X and Y
237 into $4N + 1$ and $2N + 1$ values respectively, with $X \in [-2V, 2V]$ and $Y \in [-V, V]$. Writing
238 $X_i = i * V/N$ and $Y_j = j * V/N$, the condition in equation (20) can be written:

$$\max_{i \in \{-2N, \dots, 2N\}} \left| \sum_{j=-N}^N \Phi_\sigma(X_i - Y_j) C_\sigma(Y_j) - S(X_i) \right| \quad (21)$$

239 and defining a matrix $M_{ij} = \Phi_\sigma(X_i - Y_j)$, vectors $U_j = C_\sigma(Y_j)$ and $V_i = S(X_i)$, the above is
240 equivalent to minimizing the 1-norm:

$$\|MU - V\|_1 \quad (22)$$

241 wrt U . We have an additional constraint that $U_j > 0$ for all j , since U represents a density. We first
242 explored optimizing this system with linear programming to find the U such that:

$$\begin{aligned} \min(\epsilon) \quad & \text{where} \\ -\epsilon \leq MU - V \leq \epsilon \\ U \geq 0 \end{aligned} \quad (23)$$

243 which was tractable up to a few hundred dimensions. However, the discretization error is bounded by
244 the curvature of the underlying functions which are here slightly less than one. i.e. the errors are of
245 the order of $(V/N)^2$. Here we chose $V = 20$ to provide adequate containment of the distributions
246 (the CDFs are extremely close to either zero or one outside this range). So with 200 points, we have a
247 discretization error of approximately 0.01. This is quite high. To yield higher resolution and lower
248 error, we switched to a least squares solution. Thus we seek to find U that minimizes

$$\|MU - V\|_2 \quad (24)$$

249 Its not clear that this will yield a good 1-norm for equation (22), but in practice it does. Because
250 the solution of this system is not very stable, especially in high dimensions, we actually minimize a
251 regularized version

$$\|MU - V\|_2 + \lambda \|U\|_2 \quad (25)$$

252 the solution is

$$U = (M^T M + \lambda I)^{-1} V \quad (26)$$

253 with this approach, there is no guarantee that $U \geq 0$. However, we have some flexibility in the choice
 254 of σ in equation (20). As we decrease the variance of X_{norm} , the variance of X_{corr} grows by the
 255 same amount and is in fact the result of convolution with a gaussian whose variance is the difference.
 256 Thus as σ decreases, $C(X)$ grows and approaches the derivative of a logistic function at $\sigma = 0$. It
 257 retains some very weak negative values for $\sigma > 0$ but removal of those values leads to very small
 error.

N	σ	λ	L1 error
4000	0.9	1	1.0e-4
4000	0.8	0.03	5.0e-6

258

259 5 Analysis

260 In this section we derive error bounds for our M-H test, and for the approximate target distribution
 261 that it generates. We discuss generation of the correction variable in the next section. To cut to the
 262 chase, it is possible to generate samples X_{corr} with a CDF error of at most single-precision floating
 263 point error, and possibly better. We therefore treat the X_{corr} variable as a sample from the exact
 264 correction distribution and we will not analyze its errors.

265 In the most similar prior works, [3] uses asymptotic arguments based on the central limit theorem
 266 to argue that its approximate acceptance test error tends to zero as batch size increases. But no
 267 quantitative bounds are given. In [4], explicit bounds are given, but they depend on a bound on the
 268 quantity:

$$\max_{1 \leq i \leq N} |\log p(x_i|\theta') - \log p(x_i|\theta)| \quad (27)$$

269 while such bounds can be derived easily for simple models, its unclear how to derive them for a
 270 complex model such as a neural network.

271 In this paper, we rely on quantitave forms of the central limit theorem which rely on measurable
 272 statistics from the minibatch $\{x_i\}$. Thus a sampler using our approach does not need to “see” data
 273 beyond the current minibatch. This supports use of the sampler on very large datasets, and in online
 274 mode where the dataset is presented as a stream.

275 The outline of this section is as follows: we first present bounds on the absolute and relative error (in
 276 CDF) of the distribution of Δ^* vs a Normal distribution. We then show that these errors bounds are
 277 preserved after the addition of other random variables, in particular X_{norm} and X_{corr} . From this it
 278 follows that the acceptance test has the same error bound.

279 The quantitative central-limit result below comes from [15]:

280 **Lemma 3.** *Let X_1, \dots, X_n be a set of zero-mean, independent, identically-distributed random
 281 variables with sample mean \hat{X} and variance $\sigma^2(\hat{X})$ where:*

$$\hat{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad \sigma(\hat{X}) = \frac{1}{n} \left(\sum_{i=1}^n (X_i - \hat{X})^2 \right)^{\frac{1}{2}} \quad (28)$$

282 then $t = \hat{X}/\sigma(\hat{X})$ has an approximate Student’s distribution which approaches a normal distribution
 283 in the limit. Then from [15]:

$$\sup_x |\Pr(t < x) - \Phi(x)| \leq \frac{6.4E|X|^3 + 2E|X|}{\sqrt{n}} \quad (29)$$

284 So as long as we know the first and third absolute moments $E|X|$ and $E|X|^3$, we can bound the error
 285 of the normal approximation, which decays as $O(n^{-\frac{1}{2}})$. Making the change of variables $y = x\sigma(\hat{X})$,
 286 equation (42) becomes

$$\sup_y \left| \Pr(\hat{X} < y) - \Phi \left(\frac{y}{\sigma(\hat{X})} \right) \right| \leq \frac{6.4E|X|^3 + 2E|X|}{\sqrt{n}} \quad (30)$$

287 which shows that the distribution of \hat{X} approaches the normal distribution $\mathcal{N}(0, \sigma^2(\hat{X}))$ whose
 288 variance is $\sigma^2(\hat{X})$, measured from the sample.

289 To apply this to our test, let $X_i = \Lambda_i(\theta, \theta') - \Lambda(\theta, \theta')$, then the X_i are zero-mean, IID variables.
 290 Take $n = b$ the minibatch size, and then:

$$\hat{X} = \Delta^*(\theta, \theta') - \Delta(\theta, \theta') \quad (31)$$

291 so that $\sigma(\hat{X}) = \sigma(\Delta^*)$.

292 **Corollary 1.** We can now substitute into equation (33) and displace by the mean, giving:

$$\sup_y \left| \Pr(\Delta^* < y) - \Phi\left(\frac{y - \Delta}{\sigma(\Delta^*)}\right) \right| \leq \frac{6.4E|X|^3 + 2E|X|}{\sqrt{b}} = \epsilon(\theta, \theta', b) \quad (32)$$

293 We have shown that the distribution of Δ^* approximates a Normal distribution with mean Δ and
 294 variance $\sigma^2(\Delta^*)$, and bounded its error with estimable quantities. Both $E|X|$ and $E|X|^3$ can be
 295 estimated as means of $|\Lambda_i - \Lambda|$ and $|\Lambda_i - \Lambda|^3$ respectively on each minibatch. We expect this
 296 will often be accurate enough on minibatches with hundreds or thousands of points, but otherwise
 297 bootstrap CIs can be computed from those sequences. Since the bounds are monotone in $E|X|$ and
 298 $E|X|^3$, using upper bootstrap CI limits will provide high-confidence error bounds.

299 So as long as we know the first and third absolute moments $E|X|$ and $E|X|^3$, we can bound the error
 300 of the normal approximation, which decays as $O(n^{-\frac{1}{2}})$. Making the change of variables $y = x\sigma(\hat{X})$,
 301 equation (42) becomes

$$\sup_y \left| \Pr(\hat{X} < y) - \Phi\left(\frac{y}{\sigma(\hat{X})}\right) \right| \leq \frac{6.4E|X|^3 + 2E|X|}{\sqrt{n}} \quad (33)$$

302 which shows that the distribution of \hat{X} approaches the normal distribution $\mathcal{N}(0, \sigma^2(\hat{X}))$ whose
 303 variance is $\sigma^2(\hat{X})$, measured from the sample.

304 **Lemma 4.** Let $P(x)$ and $Q(x)$ be two cumulative distributions satisfying $\sup_x |P(x) - Q(x)| \leq \epsilon$
 305 with x in some real range. Let $R(y)$ be the density of another random variable y . Let P' be the
 306 convolution $P * R$ and Q' be the convolution $Q * R$. Then $P'(z)$ (resp. $Q'(z)$) is the CDF of sum
 307 $z = x + y$ of independent random variables x with CDF $P(x)$ (resp. $Q(x)$) and y with density $R(y)$.
 308 Then

$$\sup_x |P'(x) - Q'(x)| \leq \epsilon \quad (34)$$

Proof.

$$P'(z) - Q'(z) = \int_{-\infty}^{+\infty} (P(z-x) - Q(z-x))R(x)dx \quad (35)$$

309 and since $\sup_x |P(x) - Q(x)| \leq \epsilon$ it follows that for all z :

$$-\epsilon = \int_{-\infty}^{+\infty} -\epsilon R(x)dx \leq \int_{-\infty}^{+\infty} (P(z-x) - Q(z-x))R(x)dx \leq \int_{-\infty}^{+\infty} \epsilon R(x)dx = \epsilon \quad (36)$$

310 \square

311 **Corollary 2.** If $\sup_y |\Pr(\Delta^* < y) - \Phi(\frac{y - \Delta}{\sigma(\Delta^*)})| \leq \epsilon(\theta, \theta', b)$, then

$$\sup_y |\Pr(\Delta^* + X_{\text{norm}} + X_{\text{corr}} < y) - S(y - \Delta)| \leq \epsilon(\theta, \theta', b) \quad (37)$$

312 where $S(x)$ is the standard logistic function, and X_{norm} and X_{corr} are generated as per Algorithm 1.

313 *Proof.* Apply the previous lemma twice. First take:

$$P(y) = \Pr(\Delta^* < y) \quad \text{and} \quad Q(y) = \Phi\left(\frac{y - \Delta}{\sigma(\Delta^*)}\right) \quad (38)$$

314 and convolve with the distribution of X_n which has density $\phi(X/\sigma_n)$ where $\sigma_n^2 = 1 - \sigma^2(\Delta^*)$. This
 315 yields the next iteration of P and Q

$$P'(y) = \Pr(\Delta^* + X_{\text{norm}} < y) \quad \text{and} \quad Q'(y) = \Phi(y - \Delta) \quad (39)$$

316 Now we convolve with the distribution of X_{corr} which gives:

$$P''(y) = \Pr(\Delta^* + X_{\text{norm}} + X_{\text{corr}} < y) \quad \text{and} \quad Q''(y) = S(y - \Delta) \quad (40)$$

317 Both steps preserve the error bound $\epsilon(\theta, \theta', b)$. Finally $S(y - \Delta)$ is a logistic CDF centered at
 318 Δ , and so $S(y - \Delta) = \Pr(\Delta + X_{\log} < y)$ for a logistic random X_{\log} . We conclude that the
 319 probability of acceptance for the actual test $\Pr(\Delta^* + X_{\text{norm}} + X_{\text{corr}} > 0)$ differs from the exact test
 320 $\Pr(\Delta + X_{\log} > 0)$ by at most ϵ .

321 \square

322 In fact we can do better than this approximation (showing the error decreases as $O(n^{-1})$) by using a
 323 more precise limit distribution under an additional assumption. Let μ_i denote the i^{th} moment, and b_i
 324 denote the i^{th} absolute moment of X . If Cramer's condition holds, i.e. if

$$\lim_{t \rightarrow \infty} \sup |E(\exp(itX))| < 1 \quad (41)$$

325 Bentkus et al.'s work on Edgeworth expansions [16] equation 2.2 gives:

326 **Lemma 5.** *Let X_1, \dots, X_n be a set of zero-mean, independent, identically-distributed random
 327 variables with sample mean \hat{X} and with t defined as in Lemma 3. If X satisfies Cramer's condition,
 328 then*

$$\sup_x \left| \Pr(t < x) - G\left(x, \frac{\mu_3}{b_2^{3/2}}\right) \right| \leq \frac{c(\epsilon, b_2, b_3, b_4, b_{4+\epsilon})}{n} \quad (42)$$

329 where

$$G_n(x, y) = \Phi(x) + \frac{y(2x^2 + 1)}{6\sqrt{n}} \Phi'(x) \quad (43)$$

330 This lemma shows that the average of the X_i has a more precise, skewed CDF limit $G_n(x, y)$ where
 331 the skew term has weight proportional to a certain measure of skew derived from the moments: $\frac{\mu_3}{b_2^{3/2}}$.

332 Note that if the X_i are symmetric, the weight of the correction term is zero, and the CDF of the
 333 average of the X_i converges to $\Phi(x)$ at a rate of $O(n^{-1})$.

334 Here the limit $G_n(x, y)$ is a normal CDF plus a correction term that decays as $n^{1/2}$. Importantly,
 335 since $x^2\phi(x) = \phi''(x) + \phi(x)$ where $\phi(x) = \Phi'(x)$, the correction term can be rewritten giving:

$$G_n(x, y) = \Phi(x) + \frac{y}{6\sqrt{n}} (2\phi''(x) + 3\phi(x)) \quad (44)$$

336 From which we see that $G_n(x, y)$ is a linear combination of $\Phi(x)$, $\phi(x)$ and $\phi''(x)$. This is quite
 337 fortuitous. In Algorithm 1, we correct for the difference in σ between Δ^* and the variance needed by
 338 X_{corr} using X_{norm} . This same method works when we wish to estimate the error in Δ^* vs $G_n(x, y)$.
 339 Since all of the component functions of $G_n(x, y)$ are derivatives of a (unit variance) $\Phi(x)$, adding a
 340 normal variable with variance σ' increases the variance of all three functions to $1 + \sigma'$. Thus we add
 341 X_{norm} as per Algorithm 1 preserving the limit in equation (5).

342 The deconvolution approach can be used to construct a correction variable X_{corr} between $G_n(x, y)$
 343 and $S(x)$ the standard logistic function. An additional complexity is that $G_n(x, y)$ has additional
 344 parameters y and n . Since these act as a single multiplier $\frac{y}{6\sqrt{n}}$ in equation (), its enough to consider
 345 a function $g(x, y')$ parametrized by $y' = \frac{y}{6\sqrt{n}}$. This function can be computed and saved offline.
 346 As we have shown above, errors in the “limit” function propagate directly through as errors in the
 347 acceptance test. To achieve a test error of say $1e - 6$ (close to single floating point precision), we
 348 need a y' spacing of $1e - 6$. It should not be necessary to tabulate values all the way to $y' = 1$, since
 349 y' is scaled inversely by the square root of monibatch size. Assuming a max y' of 0.1 requires us to
 350 tabulate about 100,000. Since our x resolution is 10,000, this leads to a table with about 1 billion
 351 values, which can comfortably be stored in memory. However, if $g(x, y)$ is moderately smooth in y ,
 352 it should be possible to achieve similar accuracy with a much smaller table. We leave further analysis
 353 and experiments with $g(x, y)$ as future work.

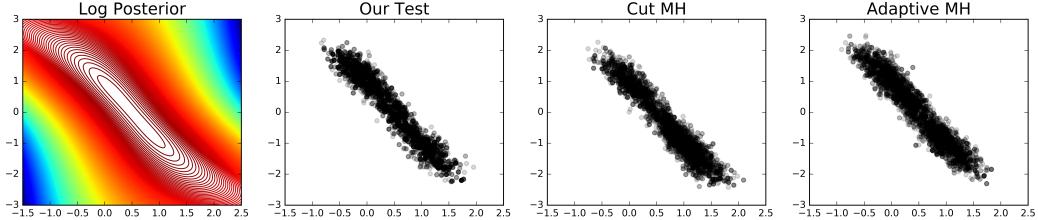


Figure 2: The log posterior contours and scatter plots of sampled θ values using different methods.

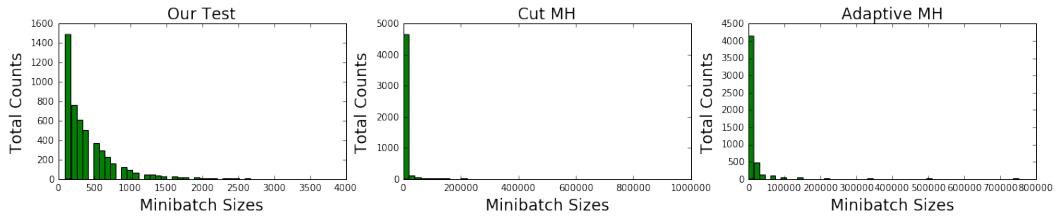


Figure 3: Histograms of minibatch sizes

354 6 Experiments

355 We conduct two sets of experiments to explore the benefits of our minibatch MH test and to benchmark
 356 it with previous work. In Section 6.1, we show that our test enables samples to converge to the
 357 posterior distribution of a heated Gaussian mixture model. In Section 6.2, we analyze its efficiency
 358 on logistic regression. Appendices B, C contain more detailed information on these respective
 359 experiments.

360 6.1 Mixture of Gaussians

361 We start with a simple Gaussian mixture model, borrowing an experiment from [12]. The parameter
 362 is 2-D, $\theta = (\theta_1, \theta_2)$, and the parameter/data generation process is

$$(363) \quad (\theta_1, \theta_2) \sim \mathcal{N}((0, 0), \text{diag}(\sigma_1^2, \sigma_2^2)); \quad x_i \sim 0.5 \cdot \mathcal{N}(\theta_1, \sigma_x^2) + 0.5 \cdot \mathcal{N}(\theta_1 + \theta_2, \sigma_x^2). \quad (45)$$

363 We set $\sigma_1^2 = 10$, $\sigma_2^2 = 1$ and $\sigma_x^2 = 2$. Fixing $\theta = (0, 1)$, we draw 1,000,000 data points so that the
 364 target distribution is $p(\theta) \prod_{i=1}^{1,000,000} p(x_i | \theta)$, with the prior based on the θ generation process in
 365 Equation 45. This results in rather sharp posterior modes and high $\text{std}(\Delta')$ estimates, so we apply a
 366 temperature $T = 10,000$ to reduce $\text{std}(\Delta')$. Taking logs, we get the target as shown in the far left of
 367 Figure 2.

368 We run MCMC with our MH test using minibatch size $m = 100$. We also run this using the method
 369 from [3] (“Cut MH”) and the method from [4]. For the method in [3], we use $m = 100$ and increment
 370 minibatches by that amount within a test if necessary. As for the method in [4], we use $m = 100$ and
 371 increase the minibatch size geometrically with a ratio of $\gamma = 1.5$. The tolerance for making a decision
 372 in [3] is $\epsilon = 0.005$, and the error bound control parameter in [4] is $p = 2, \delta = 0.01$. To make
 373 comparisons easier, all three use the same random walk proposer with covariance $\Sigma = \text{diag}(0.3, 0.3)$.
 374 This is a poor proposer, but it is sufficient for our purposes as the quality of the samples will be
 375 primarily due to the MH test. All methods are run 5000 times to collect 5000 samples.

376 Figure 2 shows scatter plots of the resulting θ samples for the four methods, with darker regions
 377 indicating a greater density of points. The three methods obtain the same rough form of the posterior,
 378 so our MH test can indeed result in the same posterior as the other two methods. Actually, based on a
 379 measure of divergence from the ground-truth distribution, our method is better than the other three
 380 methods in terms of this measurement. The measurement is to calculate the poisson likelihood of the
 381 MCMC-generated parameters over a parameter range, with $\theta_1 \in (-1.5, 2.5)$ and $\theta_2 \in (-2.5, 2.5)$.
 382 The log likelihood value for our method, cut MH method and adaptive MH method are -2144.6,
 383 -4007.4, -2598.4, respectively.

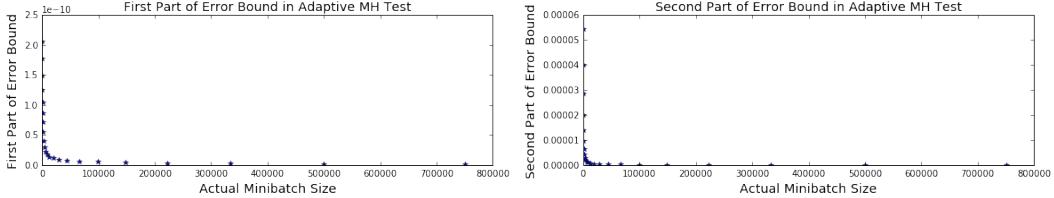


Figure 4: Value of first and second part of equation (9) in [4] with respect to actual minibatch size

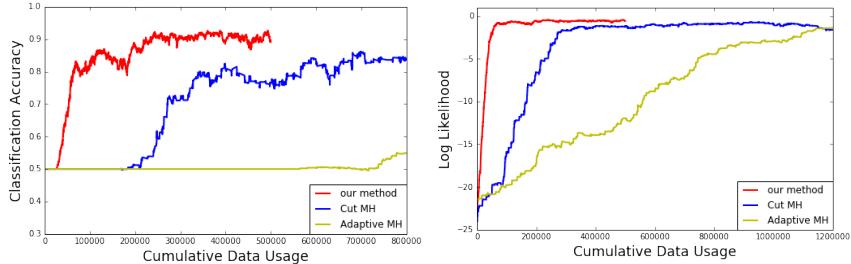


Figure 5: Logistic regression performance (accuracy/log likelihood) and minibatch size analysis

384 Figure 3 shows a histogram of the final minibatch sizes used by the three methods in each iteration. It
 385 is obvious that our methods consumes significantly less data, and most minibatch sizes are smaller
 386 than 1000, with an average minibatch size of 420. The other two methods occasionally consumes a
 387 large proportion of the entire data set, and the average minibatch sizes are 16378, 16755 for cut MH
 388 method, adaptive MH method, respectively. In terms of average minibatch size, our method is nearly
 389 39 times faster the cut MH method, and nearly 40 times faster than the adaptive MH method.

390 It is useful to notice that, in [4], they need to evaluate the likelihood term $C_{\theta, \theta'}$ over the entire the
 391 data set at every iteration in order to calculate the error bound in equation (9) in [4]. In Figure 4, the
 392 value of the first part and second part of equation (9) in [4] is shown with respect to the size of actual
 393 minibatch data used. The second part which contains the $C_{\theta, \theta'}$ term is larger than the first part, which
 394 means that the second part is non-negligible and has to be calculated at every iteration. This is one of
 395 the very time-consuming parts in this method.

396 6.2 Logistic Regression

397 We next use logistic regression for the binary classification of 1s versus 7s in the MNIST dataset [17].
 398 The data has 12007 and 1000 training and testing points, respectively (we used a random subset of
 399 the test data). The proposer is again a random walk with covariance matrix $0.05I$ for the 784×784
 400 identity matrix I . We set the posterior temperature at $T = 1000$. We set the minibatch size $m = 100$
 401 and compare with [3] with tolerance 0.005 and with [4] with error bound control parameter
 402 $p = 2, \delta = 0.01$.

403 Figure 5 shows the prediction accuracy and log likelihood on the test set as a function of the cumulative
 404 training data points processed. Our test increments the cumulative data by a fixed amount per iteration,
 405 but the other two methods may require more data per iteration. We see that our minibatch MH test is
 406 more efficient; it has similar or better performance while consuming fewer data points.

407 The plot of Figure 6 shows the histogram of minibatch sizes. Choosing an initial minibatch size of
 408 100, the cut MH method, adaptive MH method achieves final average minibatch size of 585 and 2731
 409 respectively, while our method achieves that of 100, showing significant better performance than the
 410 benchmark test methods.

411 7 Conclusions

412 In this paper, we have derived a new MH test for minibatch MCMC methods. We demonstrated
 413 how a simple deconvolution process allows us to use a minibatch approximation to the full data
 414 tests. We experimentally show the benefits of our test on Gaussian mixtures and logistic regression.

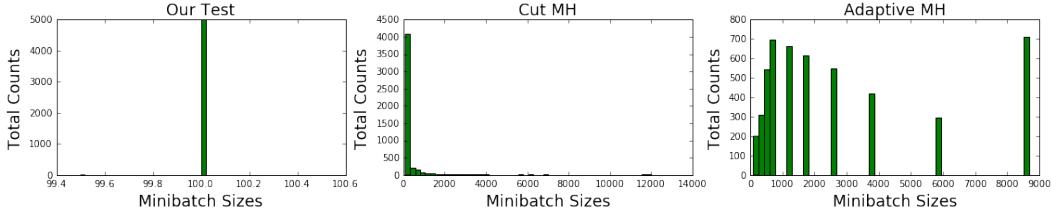


Figure 6: Histograms of minibatch sizes of logistic regression parameter estimation

415 Straightforward directions for future work include running more experiments with a particular focus
 416 on investigation of the variance precondition. More elaborate extensions include combining our
 417 results with Hamiltonian Monte Carlo methods, providing a recipe for how to use our algorithm
 418 (following the framework of [18]), or integrating parallel MCMC [19, 20] concepts.

419 References

- 420 [1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state
 421 calculations by fast computing machines,” *The Journal of Chemical Physics*, vol. 21, 1953.
- 422 [2] W. K. Hastings, “Monte carlo sampling methods using markov chains and their applications,” *Biometrika*,
 423 vol. 57, pp. 97–109, 1970.
- 424 [3] A. Korattikara, Y. Chen, and M. Welling, “Austerity in MCMC land: Cutting the metropolis-hastings
 425 budget,” in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.
- 426 [4] R. Bardenet, A. Doucet, and C. Holmes, “Towards scaling up markov chain monte carlo: an adaptive
 427 subsampling approach,” in *Proceedings of the 31st International Conference on Machine Learning (ICML)*,
 428 2014.
- 429 [5] D. Maclaurin and R. P. Adams, “Firefly monte carlo: Exact MCMC with subsets of data,” in *Proceedings
 430 of the Thirtieth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2014.
- 431 [6] A. A. Barker, “Monte-carlo calculations of the radial distribution functions for a proton-electron plasma,”
 432 *Australian Journal of Physics*, vol. 18, pp. 119–133, 1965.
- 433 [7] R. Bardenet, A. Doucet, and C. Holmes, “On markov chain monte carlo methods for tall data,” *arXiv
 434 preprint arXiv:1505.02827v1*, 2015.
- 435 [8] G. O. Roberts and J. S. Rosenthal, “Optimal scaling for various metropolis–hastings algorithms,” *Statistical
 436 Science*, vol. 16, no. 4, p. 351–367, 2001.
- 437 [9] W. Gilks and D. Spiegelhalter, *Markov chain Monte Carlo in practice*. Chapman & Hall/CRC, 1996.
- 438 [10] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of Markov Chain Monte Carlo*. CRC press,
 439 2011.
- 440 [11] R. M. Neal, “MCMC using Hamiltonian dynamics,” *Handbook of Markov Chain Monte Carlo*, vol. 54,
 441 pp. 113–162, 2010.
- 442 [12] M. Welling and Y. W. Teh, “Bayesian learning via stochastic gradient langevin dynamics,” in *Proceedings
 443 of the 28th International Conference on Machine Learning (ICML)*, 2011.
- 444 [13] S. Ahn, A. K. Balan, and M. Welling, “Bayesian posterior sampling via stochastic gradient fisher scoring.,”
 445 in *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.
- 446 [14] T. Chen, E. Fox, and C. Guestrin, “Stochastic gradient Hamiltonian Monte Carlo,” in *Proceedings of the
 447 31st International Conference on Machine Learning (ICML)*, 2014.
- 448 [15] Y. Novak, “On self-normalized sums and student’s statistic,” *Theory of Probability and its Applications*,
 449 vol. 49, no. 2, pp. 336–344, 2005.
- 450 [16] V. Bentkus, F. Gotze, and W.R.vanZwet, “An edgeworth expansion for symmetric statistics,” *Annals of
 451 Statistics*, vol. 25, no. 2, 1997.
- 452 [17] Y. LeCun and C. Cortes, “MNIST handwritten digit database,”

- 453 [18] Y. Ma, T. Chen, and E. Fox, “A complete recipe for stochastic gradient mcmc,” in *Advances in Neural*
454 *Information Processing Systems 28*, 2015.
- 455 [19] E. Angelino, E. Kohler, A. Waterland, M. Seltzer, and R. P. Adams, “Accelerating MCMC via parallel
456 predictive prefetching,” in *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*,
457 (*UAI*), 2014.
- 458 [20] S. Ahn, B. Shahbaba, and M. Welling, “Distributed stochastic gradient MCMC,” in *Proceedings of the 31st*
459 *International Conference on Machine Learning, (ICML)*, 2014.

460

Outline of Appendix

461 In this appendix, we describe the following topics:

- 462 • Proofs omitted from the main text.
 463 • More information on Section 6.1.
 464 • More information on Section 6.2.
 465 • More information on Section ??.

466 **A Proofs**

467 **lemma 1** *For the model of section 2.1, there exists a fixed (independent of N) constant c such that
 468 with probability $\geq c$ over the joint distribution of (θ, θ', u) , the tests from [4] and [3] consume all N
 469 samples.*

470 *Proof.* Assume $(\theta' - \theta) \in \pm \frac{1}{\sqrt{N}}[0.5, 1]$ and $\theta - 0.5 \in \pm \frac{1}{\sqrt{N}}[0.5, 1]$ with matching sign. These events
 471 occur with probability $p_0 = 2 * (\Phi(1) - \Phi(0.5)) = 0.2997$ and $p_1 = (\Phi(1) - \Phi(0.5)) = 0.14988$
 472 respectively, and guarantee that $\Lambda^*(\theta', \theta)$ is negative, which guarantees that we can find a u so that
 473 $\psi(u, \theta', \theta)$ equals the expected value of $\Lambda^*(\theta', \theta)$. Specifically, choose u_0 to satisfy

$$\log u_0 = N(\Lambda^*(\theta', \theta) - \psi(1, \theta', \theta)) \quad (46)$$

474 which evaluates to

$$\log u_0 = -N(\theta' - \theta) \left(\theta - 0.5 + \frac{\theta' - \theta}{2} \right) \quad (47)$$

475 and rewrite the test now as

$$\frac{1}{b} \sum_{i=1}^b x_i - \left(\theta + \frac{\theta' - \theta}{2} + \frac{\log u_0}{N(\theta' - \theta)} \right) \not\approx 0 \quad (48)$$

476 where $\not\approx$ means “significantly different from” under the distribution over samples of x_i . The above
 477 choice of u_0 ensures that the terms in parenthesis above sum to 0.5. Since the x_i have mean 0.5, the
 478 test will never correctly succeed.

479 Furthermore, if we sample values of u near enough to u_0 , the terms in parenthesis will not be
 480 sufficiently different from 0.5 to allow the test to succeed.

481 The choices above for θ and θ' guarantee that

$$\log u_0 \in -[0.5, 1][0.75, 1.5] = -[0.375, 1.5] \quad (49)$$

482 and consider the range of u values

$$\log u \in \log u_0 + [-0.5, 0.375] \quad (50)$$

483 the size of the range in u is at least $\exp([-2, -1.125]) = [0.13534, 0.32465]$ and occurs with
 484 probability at least $p_2 = 0.18932$. With u in this range, we rewrite the test as:

$$\frac{1}{b} \sum_{i=1}^b x_i - 0.5 \not\approx \frac{\log u / u_0}{N(\theta' - \theta)} \quad (51)$$

485 so that the LHS has expected value zero. Given our choice of intervals for the variables, the RHS is
 486 in the range $1/\sqrt{N}[-1, 0.75]$. The standard deviation of the LHS given the interval constraints is at
 487 least $0.5/\sqrt{b}$. And so the gap between LHS and RHS is at most $2\sqrt{b/N}$ standard deviations.

488 The samples θ , $(\theta' - \theta)$ and u are drawn independently and so the probability of the conjunction of
 489 these events is $c = p_0 p_1 p_2 = 0.0085$. \square

490 **B Gaussian Mixture Experiment Details**

491 In this section, we go over the math details on the Gaussian mixture model problem borrowed
 492 from [12]. Our parameter is a 2-D vector $\theta = (\theta_1, \theta_2)$, where

$$\theta_1 \sim \mathcal{N}(0, \sigma_1^2) \quad \text{and} \quad \theta_2 \sim \mathcal{N}(0, \sigma_2^2) \quad (52)$$

493 where \mathcal{N} indicates the normal distribution (more generally, the multivariate normal). We consider
 494 the above as our prior. Following [12], we set $\sigma_1^2 = 10$ and $\sigma_2^2 = 1$, so the covariance matrix of θ is
 495 $\Sigma = \text{diag}(10, 1)$. Therefore, the log prior probability we endow on θ is

$$\log p(\theta) = \log \left(\frac{1}{2\pi\sqrt{10}} \right) - \frac{1}{2} \theta^T \Sigma^{-1} \theta. \quad (53)$$

496 To generate the data, we use the following Gaussian mixture with tied means:

$$x_i \sim \frac{1}{2} \mathcal{N}(\theta_1, \sigma_x^2) + \frac{1}{2} \mathcal{N}(\theta_1 + \theta_2, \sigma_x^2) \quad (54)$$

497 where, again following [12], we set $\sigma_x^2 = 2$. This means the log likelihood of a single data instance is

$$\log p(x_i | \theta) = \log \left(\frac{1}{4\sqrt{\pi}} \right) + \log \left(\exp \left(-\frac{1}{4}(x_i - \theta_1)^2 \right) + \exp \left(-\frac{1}{4}(x_i - (\theta_1 + \theta_2))^2 \right) \right) \quad (55)$$

498 Here is the problem statement: given some number of conditionally independent data points
 499 x_1, x_2, \dots, x_N generated according to (54), determine the posterior distribution of θ :

$$\log p(\theta | x_1, \dots, x_N) = \log p(\theta) + \sum_{i=1}^N \log p(x_i | \theta). \quad (56)$$

500 Alternatively, if there are too many data points, we may opt to instead pick a point estimate of θ ,
 501 generally the MAP estimate. (If N is extremely large, it will cause the posterior to peak sharply at its
 502 modes, reducing distribution estimates to point estimates.) Note that in many cases, we will need to
 503 take a *minibatch estimate* of (56). In that case, the literature generally uses

$$\log p(\theta | x_1, \dots, x_N) \approx \log p(\theta) + \frac{N}{n} \sum_{i=1}^n \log p(x_i | \theta). \quad (57)$$

504 where we only use $n \ll N$ samples, but we must scale up the likelihood contribution by N/n . If we
 505 didn't add this scaling factor, then the contribution of the likelihood terms would be weaker.

506 One technique we use is adding a *temperature* to our distribution. In general, we will want to add
 507 $T > 1$ so that our posterior is $p(\theta)((\prod_{i=1}^n p(x_i | \theta))^{N/n})^{1/T}$, resulting in the log posterior of

$$\log p(\theta | x_1, \dots, x_N) \approx \log p(\theta) + \frac{1}{T} \frac{N}{n} \sum_{i=1}^n \log p(x_i | \theta). \quad (58)$$

508 which has the extra $1/T$ to decrease the scale factor. Equation (58) is what we use for our experiments,
 509 because warmer distributions help us satisfy our $\text{std}(\Delta') < 1.2$ requirement.

510 To gain some intuition on what the posterior looks like, Figure 7 shows simulated contour plots of
 511 the posterior based on varying numbers of data points N , with the temperature set at $T = 1$. Note
 512 that because we are using all N points here, the scale factor $N/n = 1$. As N increases, the posterior
 513 converges to a multimodal distribution with modes at $(0, 1)$ and $(1, -1)$. Figure 8 is similar, except
 514 this time we fix the number of samples at $N = 10000$, but show how changing the temperature T
 515 affects the distribution. A larger T implies a flatter posterior, one that (weakly) peaks in between the
 516 two true modes.

517 **C Logistic Regression Experiment Details**

518 In this section, we go over some details of our logistic regression experiment. The feature vector for
 519 an image consists of its pixel values, normalized between 0 and 1. For simplicity, we only consider

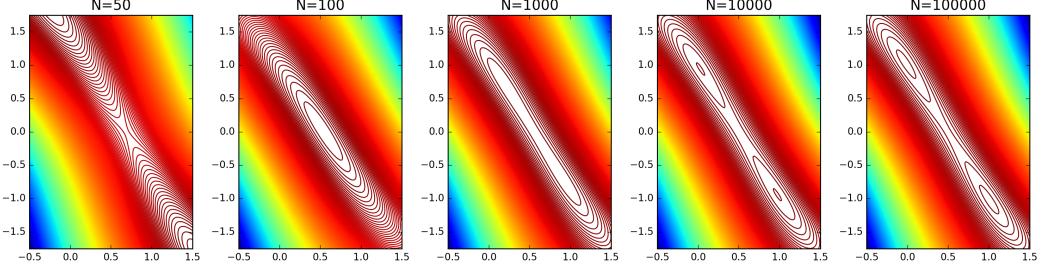


Figure 7: The posterior distribution, from 50 to 100k samples, with temperature set at 1.

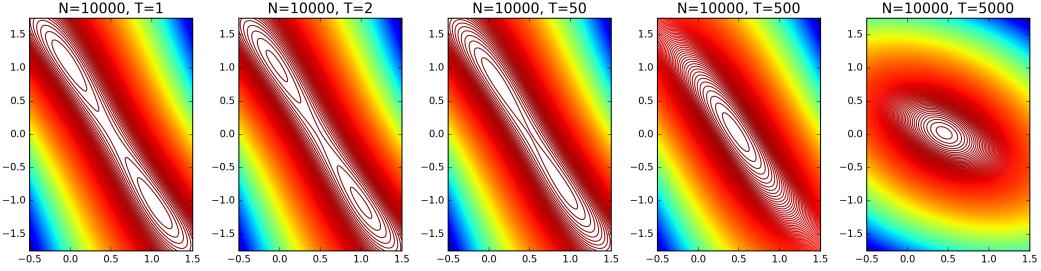


Figure 8: The posterior distribution, with $N = 10000$ but with temperature T varying.

520 the binary classification case, so we only use digits one (denoted as output $y = 1$) and seven (denoted
521 as $y = -1$). The probability of the i^{th} output $y_i \in \{-1, +1\}$ with the input vector x_i is

$$p(y_i | x_i) = \frac{1}{1 + \exp(-y_i \theta^T x_i)}, \quad (59)$$

522 where θ is the 784-length parameter vector.

523 For our experiment, we impose a uniform prior to represent our lack of knowledge about θ . We use
524 a random walk proposer, which can be modeled as $\theta' = \theta_i + \mathcal{N}(0, \sigma^2 I)$, where θ_i is the current
525 sample, θ' is the proposed sample, and we choose the variance to be a constant $\sigma^2 = 0.05$ for all
526 components. We initialize θ_0 to be a vector of all ones, and set our minibatch size as $m = 100$.

527 For our minibatch MH test, in order to enforce the $\text{std}(\Delta') < 1.2$ condition, we use a constant
528 temperature $T = 1000$. If our estimated $\text{std}(\Delta') \geq 1.2$, we ignore the current iteration.