A. Distribution of the test statistic

In the sequential test, we first compute the test statistic from a mini-batch of size m. If a decision cannot be made with this statistic, we keep increasing the mini-batch size by m datapoints until we reach a decision. This procedure is guaranteed to terminate as explained in Section 4.

The parameter ϵ controls the probability of making an error in a single test and not the complete sequential test. As the statistics across multiple tests are correlated with each other, we should first obtain the joint distribution of these statistics in order to estimate the error of the complete sequential test. Let $\bar{l}_j$ and $s_{l,j}$ be the sample mean and standard deviation respectively, computed using the first j mini-batches. Notice that when the size of a mini-batch is large enough, e.g. $n > 100$, the central limit theorem applies, and also $s_{l,j}$ is an accurate estimate of the population standard deviation. Additionally, since the degrees of freedom is high, the t-statistic in Eqn. 5 reduces to a z-statistic. Therefore, it is reasonable to make the following assumptions:

Assumption 1. *The joint distribution of the sequence* $(\bar{l}_1, \bar{l}_2, \dots)$ *follows a multivariate normal distribution.*

Assumption 2. $s_l = \sigma_l$*, where* $\sigma_l = \mathrm{std}(\{l_i\})$

Fig. 7 shows that when $\mu = \mu_0$ the empirical marginal distribution of t_j (or z_j) is well fitted by both a standard student-t and a standard normal distribution.

Under these assumptions, we state and prove the following proposition about the joint distribution of the z-statistic $\mathbf{z} = (z_1, z_2, \dots)$, where $z_j \overset{\text{def}}{=} (\bar{l}_j - \mu_0)/\sigma_l \approx t_j$, from different tests.

Proposition 2. *Given Assumption 1 and 2, the sequence* $\mathbf{z}$ *follows a* Gaussian random walk process*:*

$$P(z_j|z_1, \dots, z_{j-1}) = \mathcal{N}(m_j(z_{j-1}), \sigma_{z,j}^2) \tag{9}$$

where

$$\begin{aligned} m_j(z_{j-1}) = {} & \mu_{\text{std}} \frac{\pi_j - \pi_{j-1}}{1 - \pi_{j-1}} \frac{1}{\sqrt{\pi_j(1-\pi_j)}} \\ & + z_{j-1} \sqrt{\frac{\pi_{j-1}}{\pi_j} \frac{1-\pi_j}{1-\pi_{j-1}}} \end{aligned} \tag{10}$$

$$\sigma_{z,j}^2 = \frac{\pi_j - \pi_{j-1}}{\pi_j(1-\pi_{j-1})} \tag{11}$$

with $\mu_{\text{std}} = \frac{(\mu-\mu_0)\sqrt{N-1}}{\sigma_l}$ *being the standardized mean, and* $\pi_j = jm/N$ *the proportion of data in the first* j *mini-batches.*

Proof of Proposition 2. Denote by x_j the average of m l's in the j-th mini-batch. Taking into account the fact that the l's are drawn without replacement, we can compute the mean and covariance of the x_j's as:

$$\mathbb{E}[x_j] = \mu \tag{12}$$

$$\mathrm{Cov}(x_i, x_j) = \begin{cases} \dfrac{\sigma_l^2}{m}\left(1 - \dfrac{m-1}{N-1}\right) & , i = j \\ -\dfrac{\sigma_l^2}{N-1} & , i \neq j \end{cases} \tag{13}$$

It is trivial to derive the expression for the mean. For the covariance, we first derive the covariance matrix of single data points as

$$\begin{aligned} \mathrm{Cov}(l_k, l_{k'}) &= \mathbb{E}_{k,k'}[l_k l_{k'}] - \mathbb{E}_k[l_k]\mathbb{E}_{k'}[l_{k'}] \\ \text{if } k = k' & \\ &= \overline{l_k^2} - \mu^2 \overset{\text{def}}{=} \sigma_l^2 \\ \text{if } k \neq k' & \\ &= \mathbb{E}_{k \neq k'}[l_k l_{k'}] - \mu^2 \\ &= \frac{1}{N(N-1)}\left(\sum_{k,k'} l_k l'_k - \sum_k l_k^2\right) - \mu^2 \\ &= \frac{N}{N-1}\mu^2 - \frac{\overline{l_k^2}}{N-1} - \mu^2 \\ &= -\frac{\sigma_l^2}{N-1} \end{aligned} \tag{14}$$

Now, as x_j can be written as a linear combination of the elements in j-th mini-batch as $x_j = \frac{1}{m}\mathbf{1}^T\boldsymbol{l}_j$, the expression for covariance in Eqn. 13 follows immediately from:

$$\mathrm{Cov}(x_i, x_j) = \mathbb{E}[x_i x_j] - \mathbb{E}[x_i]\mathbb{E}[x_j] = \frac{1}{m^2}\mathbf{1}^T \mathrm{Cov}(\boldsymbol{l}_i \boldsymbol{l}_j^T)\mathbf{1} \tag{15}$$

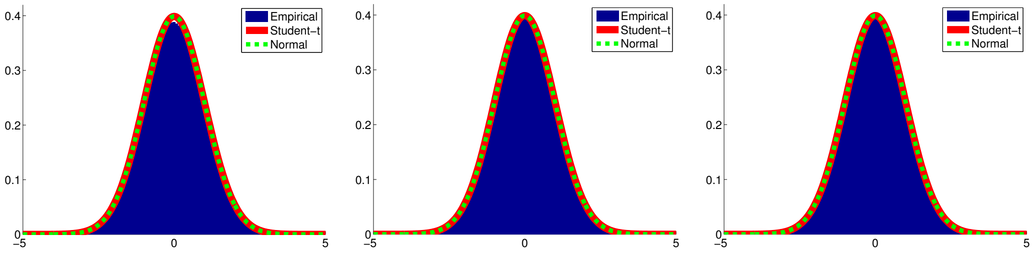
According to Assumption 1, the joint distribution of z_j's is Gaussian because z_j is a linear combination of $\bar{l}_j$'s. It is however easier to derive the mean and covariance matrix of z_j's by considering the vector $\mathbf{z}$ as a linear function of $\mathbf{x}$: $\mathbf{z} = Q(\mathbf{x} - \mu_0\mathbf{1})$ with

$$Q = \left|\begin{array}{cccc} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_j \end{array}\right| \left|\begin{array}{cccc} 1 & & & \\ 1 & 1 & & \\ \vdots & \vdots & \ddots & \\ 1 & 1 & \dots & 1 \end{array}\right| \tag{16}$$

where

$$d_j = \frac{\sqrt{N-1}}{j\sigma_x\sqrt{\frac{N-jm}{jm}}} \tag{17}$$

The mean and covariance can be computed as $\mathbb{E}[\mathbf{z}] = Q\mathbf{1}(\mu - \mu_0)$ and $\mathrm{Cov}(\mathbf{z}) = Q\mathrm{Cov}(\mathbf{x})Q^T$ and the conditional distribution $P(z_j|z_1, \dots, z_{j-1})$ follows straightforwardly. We conclude the proof by plugging the definition of μ_{std} and π_j into the distribution. □



(a) n=500 (b) n=5000 (c) n=10000

Figure 7. Empirical distribution (blue bars) of the t-statistic under resampling n datapoints without replacement from a dataset composed of digits 7 and 9 from the MNIST dataset (total $N = 12214$ points, mean of l's is removed). Also shown are a standard normal (green dashed) and a student-t distribution with $n - 1$ degrees of freedom (red solid).

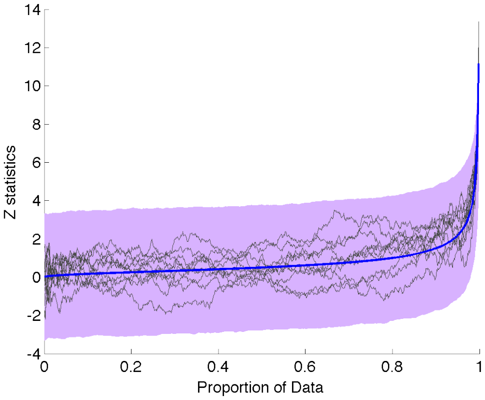


Figure 8. An example of the random walk followed by $\mathbf{z}$ with $\mu_{\text{std}} > 0$.

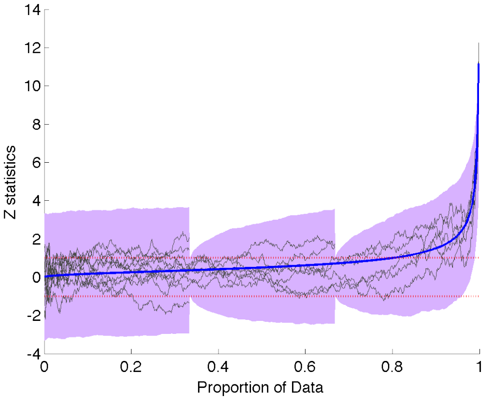


Figure 9. Sequential test with 3 mini-batches. Red dashed line is the bound G.

Fig. 8 shows the mean and 95% confidence interval of the random walk as a function of π with a few realizations of the z sequence. Notice that as the proportion of observed data π_j approaches 1, the mean of z_j approaches infinity with a constant variance of 1. This is consistent with the fact that when we observe all the data, we will always make a correct decision.

It is also worth noting that given the standardized mean μ_{std} and π_j, the process is independent of the actual size of a mini-batch m, population size N, or the variance of l's σ_l^2. Thus, Eqns. 10 and 11 apply even if we use a different size for each mini-batch. This formulation allows us to study general properties of the sequential test, independent of any particular dataset.

Applying the individual tests $\delta \gtrless \epsilon \Leftrightarrow |z_j| \gtrless \Phi(1-\epsilon) \stackrel{\text{def}}{=} G$ at the j-th mini-batch corresponds to thresholding the absolute value of z_j at π_j with a bound G as shown in Fig. 9. Instead of m and ϵ, we will use $\pi_1 = m/N$ and G as the parameters of the sequential test in the supplementary. The probability of incorrectly deciding $\mu < \mu_0$ when $\mu \geq \mu_0$ over the whole sequential test is computed as:

$$\mathcal{E}(\mu_{\text{std}}, \pi_1, G) = \sum_{j=1}^{J} P(z_j < -G, |z_i| \leq G, \forall i < j) \tag{18}$$

where $J = \lceil 1/\pi_1 \rceil$ is the maximum number of tests. Similarly the probability of incorrectly deciding $\mu \geq \mu_0$ when $\mu < \mu_0$ can be computed similarly by replacing $z_j < -G$ with $z_j > G$ in Eqn. 18. We can also compute the expected proportion of data that will be used in the sequential test as:

$$\begin{aligned}\bar{\pi}(\mu_{\text{std}}, \pi_1, G) &= \mathbb{E}_{\mathbf{z}}[\pi_{j'}] \\ &= \sum_{j=1}^{J} \pi_j P(|z_j| > G, |z_i| \leq G, \forall i < j)\end{aligned} \tag{19}$$

where j' denotes the time when the sequential test terminates. Eqn. 18 and 19 can be efficiently approximated together using a dynamic programming algorithm by discretizing the value of z_j between $[-G, G]$. The time complexity of this algorithm is $\mathcal{O}(L^2 J)$ where L is the number of discretized values.

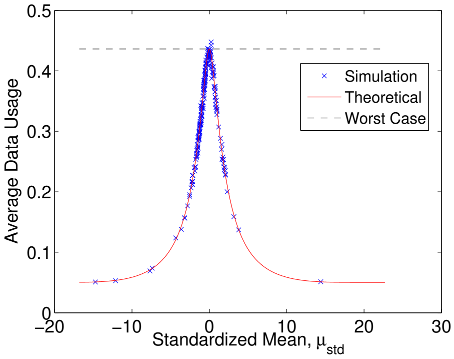


Figure 10. Average data usage $\bar{\pi}$ estimated using simulation (blue cross) and dynamic programming (red line). The worst case scenario with $\mu_{\text{std}} = 0$ is also shown (black dashed line).

The error and data usage as functions of μ_{std} are maximum in the worst case scenario when $\mu_{\text{std}} \to 0 \Leftrightarrow \mu \to \mu_0$. In this case we have:

$$\mathcal{E}(0, \pi_1, G) = \lim_{\mu_{\text{std}} \to 0} \mathcal{E}(\mu_{\text{std}}, \pi_1, G) = (1 - P(j' = J))/2$$
$$\overset{\text{def}}{=} \mathcal{E}_{\text{worst}}(\pi_1, G) \tag{20}$$

Figs. 1 and 10 show respectively that the theoretical value of the error ($\mathcal{E}$) and the average data usage ($\bar{\pi}$) estimated using our dynamic programming algorithm match the simulated values. Also, note that both error and data usage drop off very fast as μ moves away from μ_0.

B. Error in One Metropolis-Hastings Step

In the approximate Metropolis-Hasting test, one first draws a uniform random variable u, and then conducts the sequential test. As μ_{std} is a function of u (and μ, σ_l, both of which depend on θ and θ'), $\mathcal{E}$ measures the probability that one will make a wrong decision conditioned on u. One might expect that the average error in the accept/reject step of M-H using sequential test is the expected value of $\mathcal{E}$ w.r.t. to the distribution of u. But in fact, we can usually achieve a significantly smaller error than a typical value of $\mathcal{E}$. This is because with a varying u, there is some probability that $\mu > \mu_0(u)$ and also some probability that $\mu < \mu_0(u)$. Part of the error one will make given a fixed u can be canceled when we marginalize out the distribution of u. Following the definition of $\mu_0(u)$ for M-H in Eqn. 2, we can compute the actual error in the acceptance probability as:

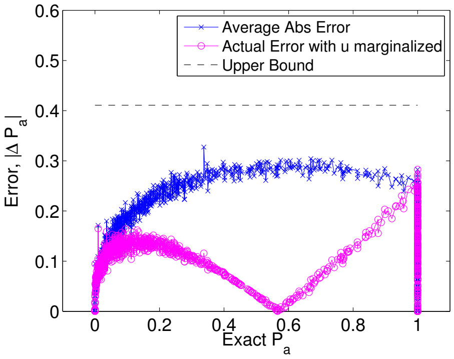


Figure 11. Error Δ in the acceptance probability (magenta circle) vs. exact acceptance probability P_a. Blue crosses are the expected value of $|\mathcal{E}|$ w.r.t. the distribution of u. Black dashed line shows the upper bound.

$$\Delta(\mu(\theta, \theta'), \sigma_l(\theta, \theta'), \pi_1, G) = P_{a,\epsilon} - P_a$$
$$= \int_0^1 P_\epsilon(\mu > \mu_0(u)) \mathrm{d}u - \int_0^{P_a} \mathrm{d}u$$
$$= \int_{P_a}^1 P_\epsilon(\mu > \mu_0(u)) \mathrm{d}u - \int_0^{P_a} (1 - P_\epsilon(\mu > \mu_0(u))) \mathrm{d}u$$
$$= \int_{P_a}^1 \mathcal{E}(\mu - \mu_0(u)) \mathrm{d}u - \int_0^{P_a} \mathcal{E}(\mu - \mu_0(u)) \mathrm{d}u \tag{21}$$

Therefore, it is often observed in experiments (see Fig. 11 for example) that when $P_a \approx 0.5$, a typical value of $\mu_{\text{std}}(u)$ is close to 0, and the average value of the absolute error $|\mathcal{E}|$ can be large. But due to the cancellation of errors, the actual acceptance probability $P_{a,\epsilon}$ can approximate P_a very well. Fig. 12 shows the approximate P_a in one step of M-H. This result also suggests that making use of some (approximate) knowledge about μ and σ_l will help us obtain a much better estimate of the error than the worst case analysis in Eqn. 20.

C. Proof of Theorem 1

C.1. Upper Bound Based on One Step Error

We first prove a lemma that will be used for the proof of Theorem 1.

Lemma 3. *Given two transition kernels, $\mathcal{T}_0$ and $\mathcal{T}_\epsilon$, with respective stationary distributions, $\mathcal{S}_0$ and $\mathcal{S}_\epsilon$, if $\mathcal{T}_0$ satisfies the following contraction condition with a constant $\eta \in [0, 1)$ for all probability distributions P:*

$$d_v(P\mathcal{T}_0, \mathcal{S}_0) \le \eta d_v(P, \mathcal{S}_0) \tag{22}$$

and the one step error between $\mathcal{T}_0$ and $\mathcal{T}_\epsilon$ is upper bounded uniformly with a constant $\Delta > 0$ as:

$$d_v(P\mathcal{T}_0, P\mathcal{T}_\epsilon) \le \Delta, \forall P \tag{23}$$

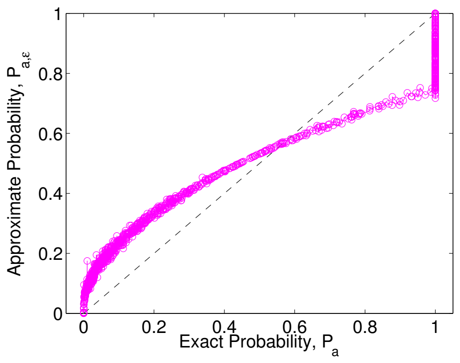


Figure 12. Approximate acceptance probability vs. true acceptance probability.

then the distance between $\mathcal{S}_0$ and $\mathcal{S}_\epsilon$ is bounded as:

$$d_v(\mathcal{S}_0, \mathcal{S}_\epsilon) \le \frac{\Delta}{1-\eta} \tag{24}$$

Proof. Consider a Markov chain with transition kernel $\mathcal{T}_\epsilon$ initialized from an arbitrary distribution P. Denote the distribution after t steps by $P^{(t)} \overset{\text{def}}{=} P\mathcal{T}_\epsilon^t$. At every time step, $t \ge 0$, we apply the transition kernel $\mathcal{T}_\epsilon$ on $P^{(t)}$. According to the one step error bound in Eqn. 23, the distance between $P^{(t+1)}$ and the distribution obtained by applying $\mathcal{T}_0$ to $P^{(t)}$ is upper bounded as:

$$d_v(P^{(t+1)}, P^{(t)}\mathcal{T}_0) = d_v(P^{(t)}\mathcal{T}_\epsilon, P^{(t)}\mathcal{T}_0) \le \Delta \tag{25}$$

Following the contraction condition of $\mathcal{T}_0$ in Eqn. 22, the distance of $P^{(t)}\mathcal{T}_0$ from its stationary distribution $\mathcal{S}_0$ is less than $P^{(t)}$ as

$$d_v(P^{(t)}\mathcal{T}_0, \mathcal{S}_0) \le \eta d_v(P^{(t)}, \mathcal{S}_0) \tag{26}$$

Now let us use the triangle inequality to combine Eqn. 25 and 26 to obtain an upper bounded for the distance between $P^{(t+1)}$ and $\mathcal{S}_0$:

$$\begin{aligned} d_v(P^{(t+1)}, \mathcal{S}_0) &\le d_v(P^{(t+1)}, P^{(t)}\mathcal{T}_0) + d_v(P^{(t)}\mathcal{T}_0, \mathcal{S}_0) \\ &\le \Delta + \eta d_v(P^{(t)}, \mathcal{S}_0) \end{aligned} \tag{27}$$

Let $r < 1 - \eta$ be any positive constant and consider the ball $\mathcal{B}(\mathcal{S}_0, \frac{\Delta}{r}) \overset{\text{def}}{=} \{P : d_v(P, \mathcal{S}_0) < \frac{\Delta}{r}\}$. When $P^{(t)}$ is outside the ball, we have $\Delta \le r d_v(P^{(t)}, S)$. Plugging this into Eqn. 27, we can obtain a contraction condition for $P^{(t)}$ towards $\mathcal{S}_0$:

$$d_v(P^{(t+1)}, \mathcal{S}_0) \le (r + \eta) d_v(P^{(t)}, \mathcal{S}_0) \tag{28}$$

So if the initial distribution P is outside the ball, the Markov chain will move monotonically into the ball within a finite number of steps. Let us denote the first time it enters the ball as t_r. If the initial distribution is already inside the ball, we simply let $t_r = 0$. We then show by induction that $P^{(t)}$ will stay inside the ball for all $t \ge t_r$.

1. At $t = t_r$, $P^{(t)} \in \mathcal{B}(\mathcal{S}_0, \frac{\Delta}{r})$ holds by the definition of t_r.
2. Assume $P^{(t)} \in \mathcal{B}(\mathcal{S}_0, \frac{\Delta}{r})$ for some $t \ge t_r$. Then, following Eqn. 27, we have

$$\begin{aligned} d_v(P^{(t+1)}, \mathcal{S}_0) \le \Delta + \eta \frac{\Delta}{r} = \frac{r+\eta}{r}\Delta < \frac{\Delta}{r} \\ \implies P^{(t+1)} \in \mathcal{B}(\mathcal{S}_0, \frac{\Delta}{r}) \end{aligned} \tag{29}$$

Therefore, $P^{(t)} \in \mathcal{B}(\mathcal{S}_0, \frac{\Delta}{r})$ holds for all $t \ge t_r$. Since $P^{(t)}$ converges to S_ϵ, it follows that:

$$d_v(\mathcal{S}_\epsilon, \mathcal{S}_0) < \frac{\Delta}{r}, \forall r < 1 - \eta \tag{30}$$

Taking the limit $r \to 1 - \eta$, we prove the lemma:

$$d_v(\mathcal{S}_\epsilon, \mathcal{S}) \le \frac{\Delta}{1-\eta} \tag{31}$$

□

C.2. Proof of Theorem 1

We first derive an upper bound for the one step error of the approximate Metropolis-Hastings algorithm, and then use Lemma 3 to prove Theorem 1. The transition kernel of the exact Metropolis-Hastings algorithm can be written as

$$\mathcal{T}_0(\theta, \theta') = P_a(\theta, \theta') q(\theta'|\theta) + (1 - P_a(\theta, \theta'))\delta_D(\theta' - \theta) \tag{32}$$

where δ_D is the Dirac delta function. For the approximate algorithm proposed in this paper, we use an approximate MH test with acceptance probability $\tilde{P}_{a,\epsilon}(\theta, \theta')$ where the error, $\Delta P_a \overset{\text{def}}{=} P_{a,\epsilon} - P_a$, is upper bounded as $|\Delta P_a| \le \Delta_{\text{max}}$. Now let us look at the distance between the distributions generated by one step of the exact kernel $\mathcal{T}_0$ and the approximate kernel $\mathcal{T}_\epsilon$. For any P,

$$\begin{aligned} &\int_{\theta'} d\Omega(\theta') |(P\mathcal{T}_\epsilon)(\theta') - (P\mathcal{T}_0)(\theta')| \\ &= \int_{\theta'} d\Omega(\theta') \left| \int_\theta dP(\theta) \Delta P_a(\theta, \theta') \left(q(\theta'|\theta) - \delta_D(\theta' - \theta)\right) \right| \\ &\le \Delta_{\text{max}} \int_{\theta'} d\Omega(\theta') \left| \int_\theta dP(\theta) (q(\theta'|\theta) + \delta_D(\theta' - \theta)) \right| \\ &= \Delta_{\text{max}} \int_{\theta'} d\Omega(\theta') \left(g_Q(\theta') + g_P(\theta')\right) = 2\Delta_{\text{max}} \end{aligned} \tag{33}$$

where $g_Q(\theta') \overset{\text{def}}{=} \int_\theta dP(\theta) q(\theta'|\theta)$ is the density that would be obtained by applying one step of Metropolis-Hastings without rejection. So we get an upper bound for the total variation distance as

$$d_v(P\mathcal{T}_\epsilon, P\mathcal{T}_0) = \frac{1}{2}\int_{\theta'} d\Omega(\theta')|P\mathcal{T}_\epsilon - P\mathcal{T}_0| \leq \Delta_{\text{max}} \quad (34)$$

Apply Lemma 3 with $\Delta = \Delta_{\text{max}}$ and we prove Theorem 1.

D. Optimal Sequential Test Design

It is possible to design optimal tests that minimize the amount of data used while keeping the error below a given tolerance. Ideally, we want to do this based on a tolerance on the error in the stationary distribution $\mathcal{S}_\epsilon$. Unfortunately, this error depends on the contraction parameter, η, of the exact transition kernel, which is difficult to compute. A more practical choice is a bound Δ_{max} on the error in the acceptance probability, since the error in $\mathcal{S}_\epsilon$ increases linearly with Δ_{max}.

Given Δ_{max}, we want to minimize the average data usage $\bar{\pi}$ over the parameters ϵ (or G) and/or m (or π_1) of the sequential test. Unfortunately, the error is a function of μ and σ_l which depend on θ and θ', and we cannot afford to change the test design at every iteration.

One solution is to base the design on the upper bound of the worst case error in Eqn. 20 which does not rely on μ_{std}. But we have shown in Section B that this is a rather loose bound and will lead to a very conservative design that wastes the power of the sequential test. Therefore, we instead propose to design the test by bounding the expectation of the error w.r.t. the distribution $P(\mu, \sigma_l)$. This leads to the following optimization problem:

$$\min_{\pi_1, G} \mathbb{E}_{\mu,\sigma_l}\mathbb{E}_u \bar{\pi}(\mu, \sigma_l, \mu_0(u), \pi_1, G)$$
$$\text{s.t. } \mathbb{E}_{\mu,\sigma_l}|\Delta(\mu, \sigma_l, \pi_1, G)| \leq \Delta_{\text{max}} \quad (35)$$

The expectation w.r.t. u can be computed accurately using one dimensional quadrature. For the expectation w.r.t. μ and σ_l, we collect a set of parameter samples (θ, θ') during burn-in, compute the corresponding μ and σ_l for each sample, and use them to empirically estimate the expectation. We can also consider collecting samples periodically and adapting the sequential design over time. Once we obtain a set of samples $\{(\mu, \sigma_l)\}$, the optimization is carried out using grid search.

We have been using a constant bound G across all the individual tests. This is known as the Pocock design (Pocock, 1977). A more flexible sequential design can be obtained by allowing G to change as a function of π. (Wang & Tsiatis, 1987) proposed a bound sequence $G_j = G_0\pi_j^{0.5-\alpha}$ where $\alpha \in [0.5, 1]$ is a free parameter. When $\alpha = 0$, it reduces to the Pocock design, and when $\alpha = 1$, it reduces to O'Brien-Fleming design (O'Brien & Fleming, 1979). We can adopt this more general form in our optimization problem straightforwardly, and the grid search will now be conducted over three parameters, π_1, G_0, and α.

E. Reversible Jump MCMC

We give a more detailed description of the different transition moves used in experiment 6.3. The update move is the usual MCMC move which involves changing the parameter vector β without changing the model γ. Specifically, we randomly pick an active component $j : \gamma_j = 1$ and set $\beta_j = \beta_j + \eta$ where $\eta \sim \mathcal{N}(0, \sigma_{update})$. The birth move involves (for $k < D$) randomly picking an inactive component $j : \gamma_j = 0$ and setting $\gamma_j = 1$. We also propose a new value for $\beta_j \sim \mathcal{N}(0, \sigma_{birth})$. The birth move is paired with a corresponding death move (for $k > 1$) which involves randomly picking an active component $j : \gamma_j = 1$ and setting $\gamma_j = 0$. The corresponding β_j is discarded. The probabilities of picking these moves $p(\gamma \to \gamma')$ is the same as in (Chen et al., 2011). The value of μ_0 used in the MH test for different moves is given below.
1. Update move:

$$\mu_0 = \frac{1}{N}\log\left[u\frac{\|\beta\|_1^{-k}}{\|\beta'\|_1^{-k}}\right] \quad (36)$$

2. Birth move:

$$\mu_0 = \frac{1}{N}\log\left[u\frac{\|\beta\|_1^{-k}p(\gamma \to \gamma')\mathcal{N}(\beta_j|0, \sigma_{birth})(D-k)}{\|\beta'\|_1^{-(k+1)}p(\gamma' \to \gamma)\lambda k}\right] \quad (37)$$

2. Death move:

$$\mu_0 = \frac{1}{N}\times$$
$$\log\left[u\frac{\|\beta\|_1^{-k}p(\gamma \to \gamma')}{\|\beta'\|_1^{-(k-1)}p(\gamma' \to \gamma)}\frac{\lambda(k-1)}{\mathcal{N}(\beta_j|0, \sigma_{birth})(D-k+1)}\right] \quad (38)$$

We used $\sigma_{update} = 0.01$ and $\sigma_{birth} = 0.1$ in this experiment. As mentioned in the main text, both the exact reversible jump algorithm and our approximate version suffer from local minima. But, when initialized with the same values, we obtain similar results with both algorithms. For example, we plot the marginal posterior probability of including a feature in the model, i.e. $p(\gamma_j = 1|X_N, y_N, \lambda)$ in figure 13.

F. Application to Gibbs Sampling

The same sequential testing method can be applied to the Gibbs sampling algorithm for discrete models. We study a model with binary variables in this paper while the extension to multi-valued variables is also possible. Consider

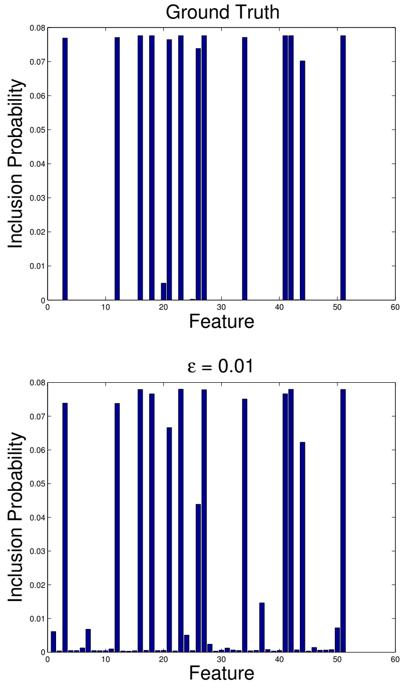


Figure 13. Marginal probability of features to be included in the model

running a Gibbs sampler on a probability distribution over D binary variables $P(X_1, \ldots, X_D)$. At every iteration, it updates one variable X_i using the following procedure:

1. Compute the conditional probability:
$$P(X_i = 1|x_{-i}) = \frac{P(X_i = 1, x_{-i})}{P(X_i = 1, x_{-i}) + P(X_i = 0, x_{-i})} \tag{39}$$
where x_{-i} denotes the value of all variables other than the i^{th} one.

2. Draw $u \sim \text{Uniform}[0, 1]$. If $u < P(X_i = 1|x_{-i})$ set $X_i = 1$, otherwise set $X_i = 0$.

The condition in step 2 is equivalent to checking:

$$\frac{\log u}{\log(1-u)} < \frac{\log P(X_i = 1, x_{-i})}{\log P(X_i = 0, x_{-i})} \tag{40}$$

When the joint distribution is expensive to compute but can be represented as a product over multiple terms, $P(X) = \prod_{n=1}^{N} f_n(X)$, we can apply our sequential test to speed up the Gibbs sampling algorithm. In this case the variable μ_0 and μ is given by

$$\mu_0 = \frac{1}{N} \frac{\log u}{\log(1-u)} \tag{41}$$

$$\mu = \frac{1}{N} \sum_{n=1}^{N} \log \frac{f_n(X_i = 1, x_{-i})}{f_n(X_i = 0, x_{-i})} \tag{42}$$

Similar to the Metropolis-Hastings algorithm, given an upper bound in the error of the approximate conditional probability

$$\Delta_{\text{max}} = \max_{i, x_{-i}} |P(X_i \text{ is assigned } 1|x_{-i}) - P(X_i = 1|x_{-i})|$$

we can prove the following theorem:

Theorem 4. *For a Gibbs sampler with a Dobrushin coefficient $\eta \in [0, 1)$ (Brémaud, 1999, §7.6.2), the distance between the stationary distribution and that of the approximate Gibbs sampler S_ϵ is upper bounded by*

$$d_v(S_0, S_\epsilon) \le \frac{\Delta_{max}}{1-\eta}$$

Proof. The proof is similar to that of Theorem 1. We first obtain an upper bound for the one step error and then plug it into Lemma 3.

The exact transition kernel of the Gibbs sampler for variable X_i can be represented by a matrix $\mathcal{T}_{0,i}$ of size $2^D \times 2^D$:

$$\mathcal{T}_{0,i}(x, y) = \begin{cases} 0 & \text{if } x_{-i} \neq y_{-i} \\ P(Y_i = y_i | y_{-i}) & \text{otherwise} \end{cases} \tag{43}$$

where $1 \le i \le N, x, y \in \{0, 1\}^D$. The approximate transition kernel $\mathcal{T}_{\epsilon,i}$ can be represented similarly as

$$\mathcal{T}_{\epsilon,i}(x, y) = \begin{cases} 0 & \text{if } x_{-i} \neq y_{-i} \\ P_\epsilon(Y_i = y_i | y_{-i}) & \text{otherwise} \end{cases} \tag{44}$$

where P_ϵ is the approximate conditional distribution. Define the approximation error $\Delta \mathcal{T}_i(x, y) \overset{\text{def}}{=} \mathcal{T}_{\epsilon,i}(x, y) - \mathcal{T}_{0,i}(x, y)$. We know that $\Delta \mathcal{T}_i(x, y) = 0$ if $y_{-i} \neq x_{-i}$ and it is upper bounded by Δ_{max} from the premise of Theorem 4.

Notice that the total variation distance reduces to a half of the L_1 distance for discrete distributions. For any distribution P, the one step error is bounded as

$$\begin{aligned} d_v(P\mathcal{T}_{\epsilon,i}, P\mathcal{T}_{0,i}) &= \frac{1}{2} \|P\mathcal{T}_{\epsilon,i} - P\mathcal{T}_{0,i}\|_1 \\ &= \frac{1}{2} \sum_y \left| \sum_x P(x) \Delta \mathcal{T}(x, y) \right| \\ &= \frac{1}{2} \sum_y \left| \sum_{x_i \in \{0,1\}} P(x_i, y_{-i}) \Delta P(x_i | y_{-i}) \right| \\ &\le \frac{1}{2} \Delta_{\text{max}} \sum_y |P(Y_{-i} = y_{-i})| \\ &= \Delta_{\text{max}} \end{aligned} \tag{45}$$

For a Gibbs sampling algorithm, we have the contraction condition (Brémaud, 1999, §7.6.2):

$$d_v(P\mathcal{T}, S) \le \eta d_v(P, S) \tag{46}$$

Plug $\Delta = \Delta_{\max}$ and η into Lemma 3 and we obtain the conclusion. □

F.1. Experiments on Markov Random Fields

We illustrate the performance of our approximate Gibbs sampling algorithm on a synthetic Markov Random Field. The model under consideration has $D = 100$ binary variables and they are densely connected by potential functions of three variables $\psi_{i,j,k}(X_i, X_j, X_k), \forall i \neq j \neq k$. There are $D(D-1)(D-2)/6$ potential functions in total (we assume potential functions with permuted indices in the argument are the same potential function), and every function has $2^3 = 8$ values. The entries in the potential function tables are drawn randomly from a log-normal distribution, $\log \psi_{i,j,k}(X_i, X_j, X_k) \sim \mathcal{N}(0, 0.02)$. To draw a Gibbs sample for one variable X_i we have to compute $(D-1)(D-2)/2 = 4851$ pairs of potential functions as

$$\frac{P(X_i = 1|x_{-i})}{P(X_i = 0|x_{-i})} = \frac{\prod_{i \neq j \neq k} \psi_{i,j,k}(X_i = 1, x_j, x_k)}{\prod_{i \neq j \neq k} \psi_{i,j,k}(X_i = 0, x_j, x_k)} \quad (47)$$

The approximate methods use a mini-batches of 500 pairs of potential functions at a time. We compare the exact Gibbs sampling algorithm with approximate versions with $\epsilon \in \{0.01, 0.05, 0.1, 0.15, 0.2, 0.25\}$.

To measure the performance in approximating $P(X)$ with samples x_t, the ideal metric would be a distance between the empirical joint distribution and P. Since it is impossible to store all the 2^{100} probabilities, we instead repeatedly draw $M = 1600$ subsets of 5 variables, $\{s_m\}_{m=1}^M, s_m \subset \{1, \ldots, D\}, |s_m| = 5$, and compute the average L_1 distance of the joint distribution on these subsets between the empirical distribution and P:

$$\text{Error} = \frac{1}{M} \sum_{s_m} \|\hat{P}(X_{s_m}) - P(X_{s_m})\|_1 \quad (48)$$

The true P is estimated by running exact Gibbs chains for a long time. We show the empirical conditional probability obtained by our approximate algorithms (percentage of X_i being assigned 1) for different ϵ in Fig. 14. It tends to underestimate large probabilities and overestimate on the other end. When $\epsilon = 0.01$, the observed maximum error is within 0.01.

Fig. 15 shows the error for different ϵ as a function of the running time. For small ϵ, we use fewer mini-batches per iteration and thus generate more samples in the same amount of time than the exact Gibbs sampler. So the error decays faster in the beginning. As more samples are collected the variance is reduced. We see that these plots converge towards their bias floor while the exact Gibbs sampler outperforms all the approximate methods at around 1000 seconds.

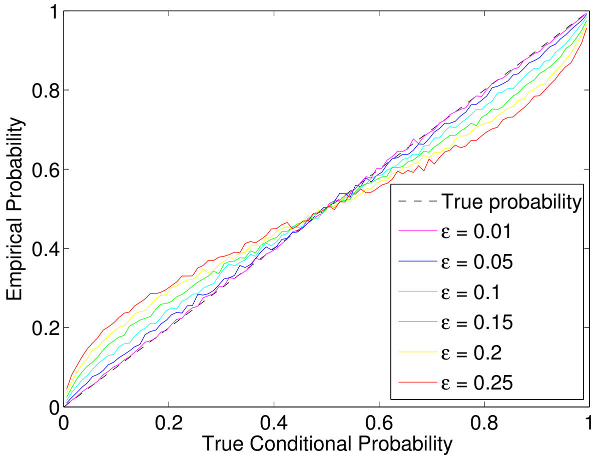


Figure 14. Empirical conditional probability vs exact conditional probability for different values of ϵ. The dotted black line shows the result for exact Gibbs sampling.

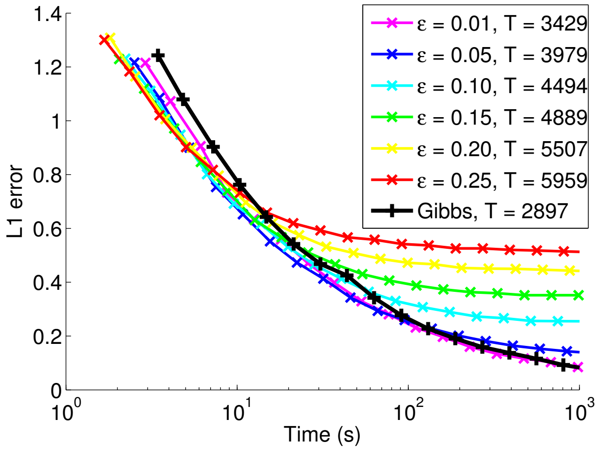


Figure 15. Average L_1 error in the joint distribution over cliques of 5 variables vs running time for different values of ϵ. The black line shows the error of Gibbs sampler with an exact acceptance probability. T in the legend indicates the number of samples obtained after 1000 seconds.