Monitoring the Convergence of MCMC Chains

STAT 548 Qualifying Paper Report

Justin J. Zhang

February 22, 2026

1 Introduction

Using *Markov Chain Monte Carlo* (MCMC) sampling algorithms, statisticians and researchers are able to estimate parameters of interest in complex problems ranging from planet detection to disease modelling. However, one prevalent concern that has not been completely solved is accurately diagnosing whether MCMC chains have converged (unless otherwise stated, this implies convergence to a stationary distribution). A popular statistic to do so is the scale reduction factor $\hat{R}$ (Gelman and Rubin, 1992), which measures how well multiple chains initialized at different points can recover the same estimate. In essence, we want to see how well the chains "forget" their respective starting points, and converge to stationarity (note that this may not be the target distribution). In most cases, $\hat{R}$ does an excellent job of monitoring convergence, but there are a number of significant underlying issues that can arise:

1. What threshold implies a "good" $\hat{R}$ value is somewhat arbitrary and problem dependent.

2. $\hat{R}$ may confuse non-convergence (high value) with a short sampling phase, i.e. we simply need to sample longer chains.

3. Since long sampling phases are needed for $\hat{R}$, the necessary computational power needed for MCMC is high.

Margossian et al. (2025), henceforth denoted MEA, create a generalization of $\hat{R}$ called *nested* $\hat{R}$, denoted $\hat{R}_\nu$, that leverages the *many short-chains regime* to mitigate the 2nd and 3rd issues. Here, we are sampling many more MCMC chains, but for a far lesser number of sampling iterations. Aiding in the effectiveness and practicality of $\hat{R}_\nu$ is the recent rise in Graphics Processing Unit (GPU) programming, which allows us to get a far greater number of samples for marginally more computational cost by running significantly more chains.

In this paper, we will first provide relevant background on $\hat{R}$, GPU programming and its corresponding GPU-compatible MCMC algorithms. Then we will formally introduce $\hat{R}_\nu$, and prove some convergence properties. Lastly, we will discuss strengths and limitations of this new method, along with some further research avenues.

2 Relevant Background Work

We start by introducing core ideas that motivate $\hat{R}_\nu$, namely the standard $\hat{R}$ diagnostic, rise of GPU programming, and associated algorithms.

2.1 Scale Reduction Factor

Iterative finite-length simulation (i.e. running multiple chains) was introduced in order to reduce the bias induced by the starting point in MCMC samples (Gelman and Rubin, 1992). Multiple chains allow us to measure the variability of our MCMC estimator, which in turn helps track convergence. Intuitively, we hope to see that the variance between chains is dominated by the variance within chains, as this means our chains are "forgetting" its starting point and producing estimators in close agreement. Formally, for M MCMC chains of length N, let $f(x_n^{(m)})$ be the estimator of the nth point from the mth chain, $\hat{f}_N^{(m)}$ be the estimator of the mth chain and $\bar{f}_N$ be the estimator across all chains. We define the *scale reduction factor* $\hat{R}$ as

$$\widehat{B} = \frac{1}{M-1}\sum_{m=1}^{M}(\hat{f}_N^{(m)} - \bar{f}_N)^2 \qquad \widehat{W} = \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N-1}\sum_{n=1}^{N}(f(x_n^{(m)}) - \hat{f}_N^{(m)})^2$$

$$\widehat{R} = \sqrt{\frac{N-1}{N} + \frac{\widehat{B}}{\widehat{W}}}\,.$$

Generally, $\hat{R} < 1.01$ signals convergence (Vehtari et al., 2021), but this is a debated topic and there is no concrete threshold. Probabilistic software like stan (Stan Development Team, 2025) often use 4 independently initialized chains to estimate $\hat{R}$, but this may require software to run very long chains to provide accurate estimates, and so MEA introduce $\hat{R}_\nu$ as an alternative.

2.2 GPU Programming

In order to run many chains without exhausting compute time, programs must make use of *parallel accelerators* like GPU's, that can run multiple chains *at the same time.* As an example, MEA shows that GPU's allow us to run 512 chains in $\sim 20\%$ more compute time than 4 chains when sampling from the Rosenbrock distribution, giving us 128 times more samples for a modest increase in time. However, we cannot simply apply our favorite MCMC method on GPU's to get good results, as many algorithms are not GPU-compatible. GPU-compatibility requires algorithms to use *Single-Instruction Multiple-Data* (SIMD) instructions, meaning they run the exact same code at the same time (Sountsov et al., 2024). To illustrate this idea, consider *Hamiltonian Monte Carlo* (HMC), a broad class of MCMC algorithms that use a momentum variable to drive our chains quickly towards areas of high probability in a series of subsamples at each iteration, called leapfrog steps (Neal, 2011). Methods that run different number of leapfrog steps for each chain will waste significant computation as it "slows down" to match the maximum number of leapfrog steps. As an example, the *No U-Turn Sampler* (NUTS), a popular HMC variant, runs a dynamic number of steps for each chain until it would 'U-turn' (which is the orbit length), and thus is not SIMD-compatible (Hoffman and Gelman, 2014). Parallelization can be done through libraries like JAX for Python, with BlackJAX for algorithmic support (Sountsov et al., 2024), which we will demonstrate in the project section. Though GPU programming is far faster than CPU programming, a trade off is that its memory capacity is significantly lower, which may affect its ability to handle extremely high-dimensional data and parameters.

2.3 Parallel MCMC Methods

In order to leverage parallelization to increase computational efficiency we must use algorithms that follow SIMD. One such option is *Change in Estimator of Expected Square* Hamiltonian Monte Carlo (ChEES HMC) (Hoffman et al., 2021). This algorithm tunes each chain *synchronously*, ensuring the same number of leapfrog steps are taken for SIMD-compatibility, though different number of leapfrog steps may be taken between iterations. For each HMC iteration, we tune each chain by maximizing the weighted gradients of autocorrelation of centered second moments

$$\nabla\text{ChEES} = \nabla\frac{1}{4}\mathbb{E}[(||\theta' - \mathbb{E}(\theta)||^2 - ||\theta - \mathbb{E}(\theta)||^2)^2].$$

The step size is tuned across chain to achieve a specific harmonic-mean acceptance probability. ChEES HMC outperforms NUTS when run on GPU's with respect to ESS per gradient calculation, a measure of computational efficiency (Hoffman et al., 2021). Moreover, it performs substantially more (between double and 10-fold) gradient calculations per second, and so has an even greater advantage looking at ESS per second. Now, since we are waiting for the “slowest” chain to catch up by tuning shorter trajectory lengths, we will see higher autocorrelation and hence a lower ESS. This implies that in lower-dimensional problems where computation time is less of an issue, non-GPU compatible methods like NUTS would work better. There are also concerns with complex, multimodal distributions where the modes have differing variances, as step size and trajectory may be trapped at small values by chains initialized in “narrow” modes. This is to say, parallelization success is algorithmic dependent and not always the solution.

3 Nested $\hat{R}$

We will now introduce the superchain regime, which leverages GPU’s to run many chains in parallel, and produces a diagnostic, $\hat{R}_\nu$, that accurately reflects convergence. In contrast to $\hat{R}$, we initialize groups of M chains from the same point, and call these clusters *constrained superchains* (there are also *naive superchains* initialized at different points, but we assume superchains to be constrained in this paper). This construction allows us to remove the uncertainty in our MCMC estimate from a single initialization in our between-chain variance, because M independent chains from an identical starting point will have variance a factor of M less than a single chain. Thus, the between-chain variance becomes a proxy for convergence to a stationary distribution. We formalize this idea throughout this section.

3.1 Definitions

Introducing superchains adds another layer to our notation. For the remainder of this paper, consider the superchain regime introduced by MEA where we have K superchains initialized independently of a distribution $x_0^{(k)} \sim p_0$. We wish p_0 to be overdispersed to obtain theoretical guarantees of convergence (see project section for details). For each superchain, we run M independent subchains starting at $x_0^{(k)}$ for N_W warmup phases and N sampling phases. Let $f^{(nmk)}$ be the nth (sampling phase) sample from the mth subchain of kth superchain. We can define the respective subchain, superchain, and overall means for $m = 1, \dots, M$ and $k = 1, \dots, K$.

$$\bar{f}^{(.mk)} = \frac{1}{N}\sum_{n=1}^{N} f^{(nmk)} \qquad \bar{f}^{(..k)} = \frac{1}{M}\sum_{m=1}^{M} \bar{f}^{(.mk)} \qquad \bar{f}^{(...)} = \frac{1}{M}\sum_{k=1}^{K} \bar{f}^{(..k)}.$$

As before, we define between-superchain and within-superchain variance, however this time the latter consists of both between-subchain variance and within-subchain variance.

$$\widehat{B}_\nu = \frac{1}{K-1}\sum_{k=1}^{K}(\bar{f}^{(..k)} - \bar{f}^{(...)})^2 \qquad \widehat{W}_\nu = \frac{1}{K}\sum_{k=1}^{K}(\widetilde{B}_k + \widetilde{W}_k)$$

$$\widetilde{B}_k = \begin{cases} \frac{1}{M-1}\sum_{m=1}^{M}(\bar{f}^{(.mk)} - \bar{f}^{(..k)})^2 & M > 1 \\ 0 & M = 1 \end{cases}$$

$$\widetilde{W}_k = \begin{cases} \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N-1}\sum_{n=1}^{N}(\bar{f}^{(nmk)} - \bar{f}^{(.mk)})^2 & N > 1 \\ 0 & N = 1 \end{cases}.$$

We formally define $\hat{R}_\nu$ as the ratio of the standard deviations of all superchains to the average within-superchain standard deviation.

$$\widehat{R}_v = \sqrt{\frac{\widehat{W}_\nu + \widehat{B}_\nu}{\widehat{W}_\nu}} = \sqrt{1 + \frac{\widehat{B}_\nu}{\widehat{W}_\nu}}\,. \tag{1}$$

3.2 Decomposing $\hat{R}_\nu$

We now wish to understand how $\hat{R}_\nu$ behaves asymptotically. By the (strong) law of large numbers and law of total variance.

$$\begin{aligned}\widehat{B}_\nu \underset{K\to\infty}{\xrightarrow{a.s.}} B_v = \operatorname{Var}(\bar{f}^{(..k)}) &= \operatorname{Var}[\mathbb{E}(\bar{f}^{(..k)} \mid x_0^{(k)})] + \mathbb{E}[\operatorname{Var}(\bar{f}^{(..k)} \mid x_0^{(k)})] \\ &= \operatorname{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{(k)})] + \frac{1}{M}\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})].\end{aligned}$$

The first term, $\operatorname{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{(k)})]$, we call *non-stationary variance*, which quantifies how well each chain "forgets" its initial value. This goes to 0 when the estimators of each chain are in close agreement, and hence variance between chains decays. In this case we say the chain has converged to a stationary distribution, which makes non-stationary variance the quantity we wish to monitor. The second term, $\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})]$, we call *persistent variance*, which quantifies the variance of the subchains themselves. In the standard $\hat{R}$ setting, $M = 1$, and so persistent variance is a substantial component of $\hat{B}$, meaning $\hat{R}$ does not explicitly model non-stationary variance. To "kill" persistent variance for a single chain, we would have to run very long chains so that the ESS is substantial and $\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})] \approx \frac{1}{ESS}\mathbb{E}[\operatorname{Var}(\bar{f}^{(nmk)} \mid x_0^{(k)})]$. This is a major weakness of $\hat{R}$ we discussed in Section 1 that is mitigated by the superchain regime.

Now $\hat{R}_\nu$ does not exactly monitor non-stationary variance. From Eq. (6) we see it is scaled by within-superchain variance $\widetilde{W}_v$. The asymptotic behaviour of $\widetilde{W}_v$ is difficult to measure due to within-subchain variance (full details in Section 4.1) but in the special case of $N = 1$, that goes away. In fact, MEA shows that by (strong) law of large numbers, for $N = 1, M > 1$

$$\widehat{W}_\nu \underset{K\to\infty}{\xrightarrow{a.s.}} W_v = \mathbb{E}(\widetilde{B}_k) = \mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^k)]. \tag{2}$$

Now we can rewrite Eq. (6) asymptotically (in K) by Continuous Mapping Theorem

$$\begin{aligned}\hat{R}_v \overset{K\to\infty}{\to} \sqrt{1 + \frac{B_v}{W_v}} &= \sqrt{1 + \frac{\operatorname{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^k)] + \frac{1}{M}\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})]}{\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})]}} \\ &= \sqrt{1 + \frac{1}{M} + \frac{\operatorname{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{(k)})]}{\mathbb{E}[\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})]}}\,.\end{aligned} \tag{3}$$

Persistent variance should converge (in N_W) to $\operatorname{Var} f$, as we will discuss in the project section. Given this, we have the relationship $\hat{R}_v < 1 + \epsilon_1 \iff \operatorname{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{(k)})] < \epsilon_2$ for some small tolerances ϵ_1, ϵ_2.

3.3 Further Considerations

To properly monitor superchain convergence to the target distribution, we must also consider sample bias in addition to variance. In fact, we can break down the MCMC error into squared bias, non-stationary variance (monitored by $\hat{R}_\nu$), and persistent variance (negligible with superchains)

$$\mathbb{E}((\bar{f}^{(..k)} - Ef)^2) = (\mathbb{E}\bar{f}^{(..k)} - \mathbb{E}f)^2 + \operatorname{Var}(\mathbb{E}(\bar{f}^{(..k)} \mid x_0^{(k)})) + \mathbb{E}(\operatorname{Var}(\bar{f}^{(..k)} \mid x_0^{(k)}))$$

Generally, a substantial warmup phase will eliminate the bias, but it is not immediately clear how long that should be. A more pressing problem is that bias may not converge to 0 if our chains are not finding the target distribution. A clear example of this is a multimodal distribution, say a mixture of Gaussians, with underdispersed initial distribution centered around a single mode. In this case we may have small non-stationary variance as the chains do converge to a stationary distribution (single mode), but clearly that is not the target. In the Project section, we will explore conditions where non-stationary variance actually bounds squared bias, and so a small $\hat{R}_\nu$ directly implies that the MCMC error has decayed.

Another issue we have that carries over from Section 1 is what threshold on $\hat{R}_\nu$ actually implies convergence. As we have seen in Eq. (3), the magnitude of persistent variance (and indirectly the target variance) will impact how well $\hat{R}_\nu$ monitors non-stationary variance. Specifically, if we set a threshold on $\hat{R}_\nu$, then our non-stationary variance will be bounded

$$\hat{R}_v \leq \delta \iff \frac{\mathrm{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{(k)})]}{\mathbb{E}[\mathrm{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)})]} \leq \delta^2 - 1 - \frac{1}{M} = \epsilon.$$

Of course for this bound to even be feasible, we need $\delta^2 - 1 > \frac{1}{M}$, which for $\delta = 1.01$, requires $M \geq 50$. Moreover, when persistent variance is large, this bound can still be substantial, in which case non-stationary variance may not have converged. MEA proposes to instead set a tolerance for scaled non-stationary variance

$$\frac{\mathrm{Var}[\mathbb{E}(\bar{f}^{(.mk)} \mid x_0^{k})]}{\mathbb{E}[\mathrm{Var}(\bar{f}^{(.mk)} \mid x_0^{k})]} \leq \tau \implies \hat{R}_v \leq \sqrt{1 + \frac{1}{M} + \tau}$$

This bound is clearly context specific but should be small next to tolerable square error. In the project section, we will discuss how tight these bounds are.

4 Analysis and Discussion

MEA gives an excellent breakdown of how $\hat{R}_\nu$ improves upon the problems with $\hat{R}$ mentioned in Section 1. However, this is not to say that it is inherently more useful in all situations. In one sense, efficiently running superchains hinges on algorithms that leverage parallel computation, and we have already touched on problems that can be faced there. In this section, we will analyze 3 conditions where our proposed $\hat{R}_\nu$ computation falls short.

4.1 Algorithmic Dependence

Nested $\hat{R}_\nu$ relies on SIMD-compatible algorithms to run many chains in parallel. We provide ChEES HMC as one algorithm, but it may not be an efficient option for certain target distributions, and in some cases will return extremely biased estimators. Consider a suitably distanced multimodal distribution, for which ChEES HMC will trap individual chains in the respective modes, hence not converging in general. A far better algorithm would allow for mode jumping, for example parallel tempering or annealing (Neal, 2001). These algorithms have their own drawbacks of longer chains, and need to set good temperature schedules between chains. What is so nice with ChEES HMC is that everything is automated and does not have to be manually tuned, whereas an algorithm like parallel tempering require that extra effort, though recent advances have helped address this (Syed et al., 2021). Nowadays, there is seemingly a MCMC algorithm for every use case, but generalizing them to be SIMD-compatible can be a chore, and users must be able to recognize this when deciding to use the many-chains regime. When the resulting parallel sampling algorithms deliver biased estimators, it would still be better to sample few chains for many samples with a CPU-compatible method.

4.2 False Convergence

There are specific use cases where $\hat{R}_\nu$ gives false signals for convergence in both directions. We have previously seen in Section 3.3 that a multimodal Gaussian with initial distribution in a single mode will have $\hat{R}_\nu$ signalling convergence, though it does not. Consider also a specific case where the target distribution is $0.5N(n,1) + 0.5N(-n,1)$ for some $n > 0$ with initial distribution normally distributed about 0. Because of the symmetry here, we expect for large K that about half of the chains will initialize and converge within each respective mode, and produce an estimator around 0, which is the true mean. However, $\hat{R}_\nu$ will be large because the between-superchain variance is substantial here. This highlights the difference between the MCMC estimator converging (in probability) and the chains themselves actually mixing well (we should note here that when the weights are not equal, $\hat{R}_\nu$ does correctly diagnose non-convergence). When K is small here, we can drastically underestimate between-superchain variance if most initializations occur in the same mode. Though these examples are quite contrived, it does merit consideration in real applications when we think the target distribution may be multimodal.

4.3 Alternative Use Cases

A class of target distributions that have issues with standard $\hat{R}$ are heavy tailed distributions. If the target has infinite variance, then within-subchain variance will dominate between-superchain variance, and $\hat{R}_\nu$ will show convergence as well regardless if the chains have found the same estimate. A proposed fix in the standard regime is to use rank-normalized or folded $\hat{R}$, which use ranks and deviation from median respectively as opposed to raw estimates (Vehtari et al., 2021). This can also be applied in our many-chains regime, producing equivalent modifications for $\hat{R}_\nu$. Moreover, there are other variations including split-$\hat{R}$ (Gelman et al., 2013), local-$\hat{R}$ (Moins et al., 2022) that can be used with the nested scheme, depending on use case.

Abstract

With the use of parallel accelerators like GPU's we can now run many Markov Chain Monte Carlo (MCMC) chains in marginally more time than a single chain. To analyze convergence of these short superchains, we use $\hat{R}_\nu$, a generalization of the scale reduction factor $\hat{R}$. This measures how well our chains forget its starting point and converge to a stationary distribution by mitigating the influence of variation within each chain. However, it does not directly measure if sample bias is also converging to 0, and so we cannot guarantee that we have found the correct stationary distribution. We propose conditions that guarantee sample bias is bounded by $\hat{R}_\nu$, and hence it can directly diagnose convergence. In this paper, we will propose and derive theoretical bounds for sample bias and the components of $\hat{R}_\nu$, as well as provide numerical experiments that verify our claims.

1 Introduction

An overarching problem in Markov Chain Monte Carlo (MCMC) sampling is how to correctly monitor convergence to a stationary distribution. One diagnostic that was recently developed is *nested* $\hat{R}$, denoted $\hat{R}_\nu$, which monitors the variance of MCMC chains (Margossian et al., 2025). It measures how well chains forget their starting point and produce estimators in agreement. However, we must ensure bias also decays, which means we need our chains to converge to the correct stationary distribution, something not directly monitored by $\hat{R}_\nu$. In general, we expect bias to decay with a sufficiently long warmup phase, but optimal length is difficult to determine, and varies between distributions. A more pressing problem is that bias may not converge to 0 if our chains are not finding the target distribution. A clear example of this is a multimodal distribution with underdispersed initial distribution. In this case we may have small variance between chains, and so they seem to converge to a stationary distribution, but are in fact trapped within a single mode. A diagnostic that also monitors bias is ideal, but that is quite difficult in practice when we do not reliably know the true mean of our target. Instead, we will show that with an overdispersed initial distribution, squared bias is bounded by *non-stationary variance* (to be defined), which is directly monitored by $\hat{R}_\nu$, and hence we can use it to diagnose convergence. In this paper, we will introduce the fundamentals of MCMC convergence and the $\hat{R}_\nu$ diagnostic, propose and prove bounds for squared bias and non-stationary variance, and verify our claims with numerical experiments.

1.1 Nested $\hat{R}$

We start by introducing the convergence diagnostic $\hat{R}_\nu$. To compute this, we run K *constrained superchains*, which are groups of M independent MCMC subchains initialized at the same point $x_0^{(k)} \sim p_0$. To run many chains efficiently, we use SIMD-compatible parallel MCMC methods on GPU's like ChEES HMC, which in favorable conditions can run hundreds more chains than CPU methods in marginally more time (Sountsov et al., 2024). We define $\hat{R}_\nu$ to measure the ratio of variance between superchains and variance within superchains. A formal derivation is in Section 4.2.

1.2 Convergence of MCMC Chains

To analyze error of MCMC superchains, we can break it down into bias and variance components. Let $f^{(nmk)}$ be the nth sample from mth subchain in kth superchain, $\bar{f}^{(\cdot mk)}$ be the mean of mth subchain in kth superchain, and $\bar{f}^{(\cdot\cdot k)}$ be the mean of kth superchain. Then,

$$\mathbb{E}((\bar{f}^{(\cdot\cdot k)} - Ef)^2) = (\mathbb{E}\bar{f}^{(\cdot\cdot k)} - \mathbb{E}f)^2 + \text{Var}(\mathbb{E}(\bar{f}^{(\cdot\cdot k)} \mid x_0^{(k)})) + \mathbb{E}(\text{Var}(\bar{f}^{(\cdot\cdot k)} \mid x_0^{(k)})).$$

The last term, *persistent variance*, is negligible in our superchain regime, as $\mathbb{E}(\mathrm{Var}(\bar{f}^{(..k)} \mid x_0^{(k)})) = \frac{1}{M}\mathbb{E}(\mathrm{Var}(\bar{f}^{(.mk)} \mid x_0^{(k)}))$, which goes to 0 as M increases. The second term, *non-stationary variance*, will disappear as chains converge to stationarity, and is estimated by the sample variance of the superchain Monte Carlo estimators. Together, $\hat{R}_\nu$ directly monitors variance of our MCMC chains. Estimating the first term, squared bias, requires us to have a good estimate of the sample mean, which we cannot assume to be true. Thus, we require a proxy measure of bias, which we will introduce in the next section.

1.3 Related Works

A detailed analysis of $\hat{R}_\nu$ properties and performance is found in Margossian et al. (2025). The question of what threshold on $\hat{R}_\nu$ (equivalently standard $\hat{R}$) actually implies convergence to a stationary distribution is prevalent, with no definitive answers (Vehtari et al., 2021). Margossian et al. (2025) do argue that it suffices to have a tolerable error on non-stationary variance, though that is problem-dependent as well. There are unbiased MCMC algorithms that remove squared bias through coupling, and thus allow $\hat{R}_\nu$ to monitor squared error (Jacon et al., 2020). However, the natural trade off is the relative inefficiency (i.e. higher variance) and far longer compute time (in fact it is not SIMD compatible and so the benefits of parallelization are not readily applicable). Additionally, there are methods like annealed importance sampling (Neal, 2001) and sequential Monte Carlo (Del Moral et al., 2006) that control bias without running long warmups, but they exhibit increased variance and are difficult to tune.

2 Analysis of Convergence

We now show that with an overdispersed distribution, the squared bias is bounded by the non-stationary variance. The problem of how to choose a distribution that is both overdispersed and similar to our target distribution (in the sense that it does not initialize in areas with negligible density) is important but will not be touched on. In this section we will theoretically derive bounds for non-stationary variance and bias, and show that they decay at the same rate. We base our work on ideas from Margossian.

2.1 Bounding Non-stationary Variance

To simplify our discussion, we will consider a single chain within our many short chains regime. We denote x_0 as the initial point and $\hat{f}_N$ as the estimator after N sampling iterations (including warmup). We write the conditional bias

$$|\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f| = b(x_0)h(x_0, N). \tag{4}$$

where $b(x_0) = |f(x_0) - \mathbb{E}f|$ is the initial bias and h is a decay function with $h(x_0, 0) = 1$. Here, N is the total number of sampling steps including warmup. For a chain that converges to stationarity, we expect $h(x_0, N) \to 0$ in our sampling phase (but may not hold otherwise). We first state a bound for non-stationary variance in terms of these quantities.

Theorem 2.1. *Given Eq. (4), we can derive the following bound:*

$$\mathrm{Var}(b(x_0)h(x_0, N)) \leq \mathrm{Var}\,\mathbb{E}(\hat{f}_N \mid x_0) \leq \mathbb{E}(b^2(x_0)h^2(x_0, N)). \tag{5}$$

Proof is in Section 4.3. We originally assumed that h depends on both initialization and sampling length, but we can relax this by assuming our decay $h(N)$ is solely dependent on number of iterations.

It has been shown that this is possible under uniform boundedness of the target to proposal ratio using the Metropolis Hastings algorithm (Wang, 2022). Alternatively, it would be worthwhile to investigate upper and lower bounds of the decay function depending on initialization. Under this relaxation we get the following corollary.

Corollary 2.2. *Suppose a decay function $h(N)$ independent of the initialization. Then we can rewrite Eq. (5) as*

$$\operatorname{Var}(b(x_0))h^2(N) \leq \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0) \leq \mathbb{E}(b^2(x_0))h^2(N).$$

Under these assumptions, we see that the non-stationary variance will decay at approximately the rate $h(N)$ as $N \to \infty$. How tight these bounds are depends on the difference $\mathbb{E}(b^2(x_0)) - \operatorname{Var}(b(x_0)) = (\mathbb{E}b^2(x_0))^2$. When the expected initial bias is very small, this gives an exact equation for the decay of non-stationary variance.

2.2 Bounding Squared Bias

To derive bounds on the marginal bias, we write the initial variance as

$$\operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0) = \operatorname{Var} f(x_0)g(N)$$

where $\operatorname{Var} f(x_0)$ is the initial variance and $g(N)$ is some decay function. We are assuming that the decay of non-stationary variance is independent of initialization but can relax that and use decay $g(x_0, N)$ as with our breakdown of conditional bias.

Proposition 2.3. *Suppose the conditional bias and non-stationary variances have decay rates $h(N), g(N)$ that are independent of initial point. Furthermore, assume that $g(N) = h^2(N)$. Then,*

$$\operatorname{Var} f(x_0) \geq (\mathbb{E}b(x_0))^2 \implies \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0) \geq (\mathbb{E}\hat{f}_N - \mathbb{E}f)^2$$

Proof is in Section 4.4. Notice that our condition here is exactly that of overdispersion, requiring the variance of our initial distribution to be greater than the squared initial bias. Note however that the converse is not true in general unless the conditional and marginal biases are equal. To conclude this section, we will state another condition for which squared bias will be bounded by non-stationary variance, which is does not require identical decay rates by using Corollary 2.2 (proof in Section 4.5).

Proposition 2.4. *Suppose the conditional bias and non-stationary variances have decay rates $h(N), g(N)$ that are independent of initial point. Then,*

$$\operatorname{Var} f(x_0) + C \geq (\mathbb{E}b(x_0))^2 \implies \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0) \geq (\mathbb{E}\hat{f}_N - \mathbb{E}f)^2$$

where $C = [\mathbb{E}(f(x_0) - \mathbb{E}f)]^2 - [\mathbb{E}|f(x_0) - \mathbb{E}f|]^2$.

2.3 Bounding persistent Variance

For $\hat{R}_\nu$ to monitor non-stationary variance, we require persistent variance to be constant with respect to N and x_0, otherwise the $\hat{R}_\nu$ estimate will be unduly influenced by the sampling step. Moreover, it is of interest how it behaves relative to the target variance, $\operatorname{Var} f$, which it should converge to. The following proposition provides bounds for persistent variance.

Proposition 2.5. *Suppose as before that the conditional bias has decay rate independent of initial point. Additionally, we assume that $\mathbb{E}|f| < \infty$. Given Eq. (4), we can bound the persistent variance around the target variance*

$$C - 2\mathbb{E}|f|\mathbb{E}(b(x_0))h(N) + \operatorname{Var} f \leq \mathbb{E}\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0) \leq C - \mathbb{E}|f|\mathbb{E}(b(x_0))h(N) + \operatorname{Var} f$$

where $C = \mathbb{E}((\bar{f}^{(.mk)})^2) - \mathbb{E}(f^2)$ is the difference in squared estimators.

Proof is in Section 4.6. Under the assumption $\mathbb{E}|f| < \infty$, the middle terms will be dominated by the conditional bias, which will go to 0 if the MCMC chains converge to stationarity. Hence, persistent variance being stable as N increases boils down to whether the chain variance is similar to the target variance. If this is true, we can be confident that the persistent variance will not overly influence non-stationary variance, and in nice cases can adjust explicitly obtain non-stationary variance from $\hat{R}_\nu$, using Eq. (3).

3 Numerical Experiments

We run a series of experiments to demonstrate the bounds we derived in the previous section. We will first consider a simple Gaussian example to illustrate proof of concept. Then we will consider the high-dimensional *Item Response Theory* (IRT) model, to analyze performance in a more setting. Finally, we look at a bimodal distribution, to understand how our quantities work under non-convergence. A full description of the models will be in the Section 4.7. In each example, we will consider the first moment. We will run 32 superchains of 64 subchains each, using ChEES HMC on a T4 GPU to tune these chains in parallel. Results are based on iteratively sampling $N = 1$ iterations, with each iteration being considered part of the "warmup" for subsequent iterations.

Fig. 1 compares non-stationary variance to squared bias on the log scale. We see that they clearly decrease together at the same rate until they reach stationarity, which supports our assumption in Proposition 2.3 that the decay rates are equal. Along with the fact that initial variance is greater than initial squared bias in our simulated examples, satisfying assumptions (shown in Section 4.8), these numerical experiments support our conclusions in Proposition 2.3. We should note that once the chains are approximately stationary, there are numerical issues (i.e. significant oscillation, and even negative variances), which may be induced by HMC properties (there is room for further investigation). Even for the bimodal Gaussian case, non-stationary variance dominates initial bias even though the chains do not converge to stationarity.

Fig. 2 compares the persistent variance over successive sampling iterations, to determine whether they converge to the target variance, and if the bounds in Eq. (7) hold. In the Gaussian and IRT examples, it converges to $\operatorname{Var} f$ and oscillates around it (note that IRT is sufficiently warmed up at start of sampling phase while Gaussian is not). Though we cannot explicitly compute the bounds from Proposition 2.5, it likely holds because we know from Fig. 1 that decay rate $h(N)$ goes to 0 quickly, and the additional term C is small because the bias is negligible. In our bimodal Gaussian example, persistent variance is not anywhere near the target variance, because each chain converges inside a small mode. This does not immediately contradict our bound from Proposition 2.5, as we expect both high bias for f^2 and a non-decaying $h(N)$ function.

4 Appendix

4.1 Convergence of Within-Subchain Variance

In this appendix section, we wish to formally state the within-subchain variance for general N, which was stated for $N = 1$ in Eq. (2). The general limit is

$$\widehat{W}_\nu \xrightarrow[K\to\infty]{a.s.} W_v = \mathbb{E}(\widetilde{B}_k) = \mathbb{E}[\mathrm{Var}(\bar{f}^{(.mk)} \mid x_0^k)] + W'$$

$$W' = \begin{cases} \frac{1}{N-1}\sum_{n=1}^{N}[\mathrm{Var}\, f^{(nmk)} - \mathrm{Var}\, \bar{f}^{(.mk)} + (\mathbb{E}f^{(nmk)})^2 - (\mathbb{E}\bar{f}^{(.mk)})^2] & N > 1 \\ 0 & N = 1 \end{cases}$$

The proof will be omitted.

4.2 Formal Definition of $\hat{R}_\nu$

In this appendix section, we wish to formally define $\hat{R}_\nu$ for completeness. Consider the superchain regime introduced in Margossian et al. (2025) where we have K superchains initialized independently of a distribution $x_0^{(k)} \sim p_0$. For theoretical guarantees to hold, we wish p_0 to be overdispersed. For each superchain, we run M independent subchains starting at $x_0^{(k)}$ for N_W warmup phases and N sampling phases. Let $f^{(nmk)}$ be the nth (sampling phase) sample from the mth subchain of kth superchain. We can define the respective subchain, superchain, and overall means for $m = 1, \ldots, M$ and $k = 1, \ldots, K$.

$$\bar{f}^{(.mk)} = \frac{1}{N}\sum_{n=1}^{N} f^{(nmk)} \qquad \bar{f}^{(..k)} = \frac{1}{M}\sum_{m=1}^{M} \bar{f}^{(.mk)} \qquad \bar{f}^{(...)} = \frac{1}{M}\sum_{k=1}^{K} \bar{f}^{(..k)}.$$

As before, we define between-superchain and within-superchain variance, however this time the latter consists of both between-subchain variance and within-subchain variance.

$$\widehat{B}_\nu = \frac{1}{K-1}\sum_{k=1}^{K}(\bar{f}^{(..k)} - \bar{f}^{(...)})^2 \qquad \widehat{W}_\nu = \frac{1}{K}\sum_{k=1}^{K}(\widetilde{B}_k + \widetilde{W}_k)$$

$$\widetilde{B}_k = \begin{cases} \frac{1}{M-1}\sum_{m=1}^{M}(\bar{f}^{(.mk)} - \bar{f}^{(..k)})^2 & M > 1 \\ 0 & M = 1 \end{cases}$$

$$\widetilde{W}_k = \begin{cases} \frac{1}{M}\sum_{m=1}^{M} \frac{1}{N-1}\sum_{n=1}^{N}(\bar{f}^{(nmk)} - \bar{f}^{(.mk)})^2 & N > 1 \\ 0 & N = 1 \end{cases}.$$

We formally define $\hat{R}_\nu$ as the ratio of the standard deviations of all superchains to the average within-superchain standard deviation.

$$\widehat{R}_v = \sqrt{\frac{\widehat{W}_\nu + \widehat{B}_\nu}{\widehat{W}_\nu}} = \sqrt{1 + \frac{\widehat{B}_\nu}{\widehat{W}_\nu}}. \tag{6}$$

4.3 Bounding Non-Stationary Variance

We wish to prove Theorem 2.1.

Proof. To obtain these bounds, we apply variance expansions and Eq. (4)

$$\begin{aligned}\operatorname{Var}(b(x_0)h(x_0,N)) = \operatorname{Var}|\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f| &\leq \operatorname{Var}(\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f)\\ &= \mathbb{E}[(\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f)^2]\\ &= \mathbb{E}[b^2(x_0)h^2(x_0,N)]\end{aligned}$$

The first inequality holds by the monotonicity of expectation (writing out variance in expectation form). The second equality holds by variance expansion. Since $\mathbb{E}f$ is constant, we can substitute $\operatorname{Var}(\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f) = \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0)$ in our above equations to get the desired bounds. □

4.4 Bounding Squared Bias 1

We wish to prove Proposition 2.3.

Proof. We first bound the marginal squared bias by the conditional squared bias with Jensen's inequality using the fact $\mathbb{E}\mathbb{E}f = \mathbb{E}f$ since it is constant,

$$|\mathbb{E}\hat{f}_N - \mathbb{E}f| = |\mathbb{E}(\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f)| \leq \mathbb{E}|\mathbb{E}(\hat{f}_N \mid x_0) - \mathbb{E}f| = h(N)\mathbb{E}|b(x_0)|$$

It follows directly by assumption that

$$(\mathbb{E}\hat{f}_N - \mathbb{E}f)^2 \leq h^2(N)(\mathbb{E}b(x_0))^2 \leq h^2(N)\operatorname{Var}f(x_0) = \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0)$$

as desired. □

4.5 Bounding Squared Bias 2

We wish to prove Proposition 2.4.

Proof. To do this, consider the lower bound for Corollary 2.2

$$\operatorname{Var}(b(x_0))h^2(N) \leq \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0) = \operatorname{Var}f(x_0)g(N)$$

We can rewrite the variance of the conditional bias

$$\begin{aligned}\operatorname{Var}(b(x_0)) &= \operatorname{Var}|f(x_0) - \mathbb{E}f|\\ &= \mathbb{E}[|f(x_0) - \mathbb{E}f|^2] - [\mathbb{E}|f(x_0) - \mathbb{E}f|]^2\\ &= \mathbb{E}[(f(x_0) - \mathbb{E}f)^2] - [\mathbb{E}(f(x_0) - \mathbb{E}f)]^2 + [\mathbb{E}(f(x_0) - \mathbb{E}f)]^2 - [\mathbb{E}|f(x_0) - \mathbb{E}f|]^2\\ &= \operatorname{Var}(f(x_0) - \mathbb{E}f) + [\mathbb{E}(f(x_0) - \mathbb{E}f)]^2 - [\mathbb{E}|f(x_0) - \mathbb{E}f|]^2\\ &= \operatorname{Var}f(x_0) + C\end{aligned}$$

where $C = [\mathbb{E}(f(x_0) - \mathbb{E}f)]^2 - [\mathbb{E}|f(x_0) - \mathbb{E}f|]^2$. Recall, by assumption that $\operatorname{Var}f(x_0) + C \geq (\mathbb{E}b(x_0))^2$. We showed in the proof for Proposition 2.4 that we can bound marginal squared bias by conditional squared bias, and so using Eq. (4)

$$\begin{aligned}(\mathbb{E}\hat{f}_N - \mathbb{E}f)^2 &\leq h^2(N)(\mathbb{E}b(x_0))^2\\ &\leq h^2(N)(\operatorname{Var}f(x_0) + C)\\ &= h^2(N)\operatorname{Var}(b(x_0))\\ &\leq \operatorname{Var}\mathbb{E}(\hat{f}_N \mid x_0)\end{aligned}$$

as desired. □

Verifying this condition requires us to compute C, but that is immediate if we can compute $\mathbb{E}(b(x_0))$. Thus, this bound is more easily verifiable than Proposition 2.3, which requires us to verify that decay rates are identical.

4.6 Bounding Persistent Variance

We wish to verify our proof in Proposition 2.5.

Proof. To start, we decompose the persistent variance in order to induce the target variance.

$$\begin{aligned}\mathbb{E}\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0) &= \mathbb{E}\mathbb{E}[(\bar{f}^{(.mk)} - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))^2 \mid x_0] \qquad (7)\\ &= \mathbb{E}[(\bar{f}^{(.mk)} - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))^2]\\ &= \mathbb{E}[((\bar{f}^{(.mk)} - f) + (f - \mathbb{E}f) + (\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0)))^2]\\ &= \mathbb{E}[(\bar{f}^{(.mk)} - f)^2] + \mathbb{E}[(f - \mathbb{E}f)^2] + \mathbb{E}[(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))^2]\\ &\quad + 2\mathbb{E}[(\bar{f}^{(.mk)} - f)(f - \mathbb{E}f)] + 2\mathbb{E}[(\bar{f}^{(.mk)} - f)(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))]\\ &\quad + 2\mathbb{E}[(f - \mathbb{E}f)(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))]\end{aligned}$$

Notice that the second and third terms are exactly the target variance and conditional bias squared (from Eq. (4)).

$$\begin{aligned}\mathbb{E}[(f - \mathbb{E}f)^2] &= \operatorname{Var} f\\ \mathbb{E}[(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))^2] &= \mathbb{E}(b^2(x_0))h^2(N)\end{aligned}$$

We individually break down the cross terms. We will use the fact that x_0, f are both independent with respect to x_0.

$$\begin{aligned}\mathbb{E}[(\bar{f}^{(.mk)} - f)(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))] &= \mathbb{E}\mathbb{E}[(\bar{f}^{(.mk)} - f)(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0)) \mid x_0]\\ &= \mathbb{E}[(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))\mathbb{E}(\bar{f}^{(.mk)} - f \mid x_0)]\\ &= -\mathbb{E}[(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))^2]\\ &= -\mathbb{E}(b^2(x_0))h^2(N)\\ \mathbb{E}[(f - \mathbb{E}f)(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))] &= \mathbb{E}(f - \mathbb{E}f)\mathbb{E}(\mathbb{E}f - \mathbb{E}(\bar{f}^{(.mk)} \mid x_0))\\ &= 0\end{aligned}$$

Lastly, we combine the first squared term and the first cross term to get a crude bound

$$\begin{aligned}\mathbb{E}[(\bar{f}^{(.mk)} - f)^2] + 2\mathbb{E}[(\bar{f}^{(.mk)} - f)(f - \mathbb{E}f)] &= \mathbb{E}[(\bar{f}^{(.mk)} - f)(\bar{f}^{(.mk)} + f - 2\mathbb{E}f)]\\ &= [\mathbb{E}((\bar{f}^{(.mk)})^2) - \mathbb{E}(f^2)] - 2\mathbb{E}f(\mathbb{E}(\bar{f}^{(.mk)} - f))\end{aligned}$$

We are left with the bias of the squared estimator $C = \mathbb{E}((\bar{f}^{(.mk)})^2) - \mathbb{E}(f^2)$, which will be small when the superchain variance is close to the target variance. Plugging our terms into Eq. (7) gives

$$\begin{aligned}\mathbb{E}\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0) &= C - 2\mathbb{E}f(\mathbb{E}(\bar{f}^{(.mk)} - f)) + \operatorname{Var} f + \mathbb{E}(b^2(x_0))h^2(N) - 2\mathbb{E}(b^2(x_0))h^2(N) \qquad (8)\\ &= C - 2\mathbb{E}f(\mathbb{E}(\bar{f}^{(.mk)} - f)) + \operatorname{Var} f\end{aligned}$$

Now we can bound the middle term here

$$| - \mathbb{E}f(\mathbb{E}(\bar{f}^{(.mk)} - f))| \leq \mathbb{E}|f|\mathbb{E}|\bar{f}^{(.mk)} - f| \leq \mathbb{E}|f|\mathbb{E}(b(x_0))h(N)$$

Putting this back into Eq. (8) yields the result

$$C - 2\mathbb{E}|f|\mathbb{E}(b(x_0))h(N) + \operatorname{Var} f \leq \mathbb{E}\operatorname{Var}(\bar{f}^{(.mk)} \mid x_0) \leq C - \mathbb{E}|f|\mathbb{E}(b(x_0))h(N) + \operatorname{Var} f$$

□

Under the assumption $\mathbb{E}|f| < \infty$, we can scale f and the middle terms will be dominated by the conditional bias, which if we converge to stationarity will go to 0 (should note that if we do not converge to stationarity, there may be variation in persistent variance, but $\hat{R}_\nu$ should still convey non-convergence). Hence, persistent variance being stable as N increases boils down to whether the chain variance is similar to the target variance.

4.7 Short Explanation of Experiments

In our numerical experiments we run 3 models:

1. Unimodal Gaussian: $N(5, 4)$, with initial distribution $N(5, 100)$. We only use 1 warmup iterations to establish a meaningful decay rate, because this target will converge very fast.

2. Item Response Theory (used in Margossian et al. (2025)): We have a hierarchical model to model students exam taking abilities. This is a 501-dimensional model with $J = 400$ students and $L = 100$ questions. The parameters are mean student ability $\delta \in \mathbb{R}$, individual abilities $\boldsymbol{\alpha} \in \mathbb{R}^J$ and question difficulty $\boldsymbol{\beta} \in \mathbb{R}^L$. Each observation y_{jl} is student j response to question l. Our model is

$$\begin{aligned} \delta &\sim N(0.75, 1) \\ \boldsymbol{\alpha} &\sim MVN(0, I) \\ \boldsymbol{\beta} &\sim MVN(0, I) \\ y_{jl} &\sim \text{Bernoulli}(\text{logit}^{-1}(\alpha_j - \beta_l + \delta)) \end{aligned}$$

   We use 25 warmup iterations and consider the first marginal only. IRT (and its associated functions $\mathbb{E}f, \operatorname{Var} f$) are available in the *inference gym* package.

3. Bimodal Gaussian: $0.7N(10, 1) + 0.3N(-10, 1)$, with initial distribution $N(0, 400)$. We use 10 warmup iterations.

For these examples, we will run 32 superchains of 64 subchains each, and use a sampling phase of $N = 1$. We use ChEES HMC on a T4 GPU to tune these chains in parallel.

4.8 Non-stationary Variance and Bias Comparison

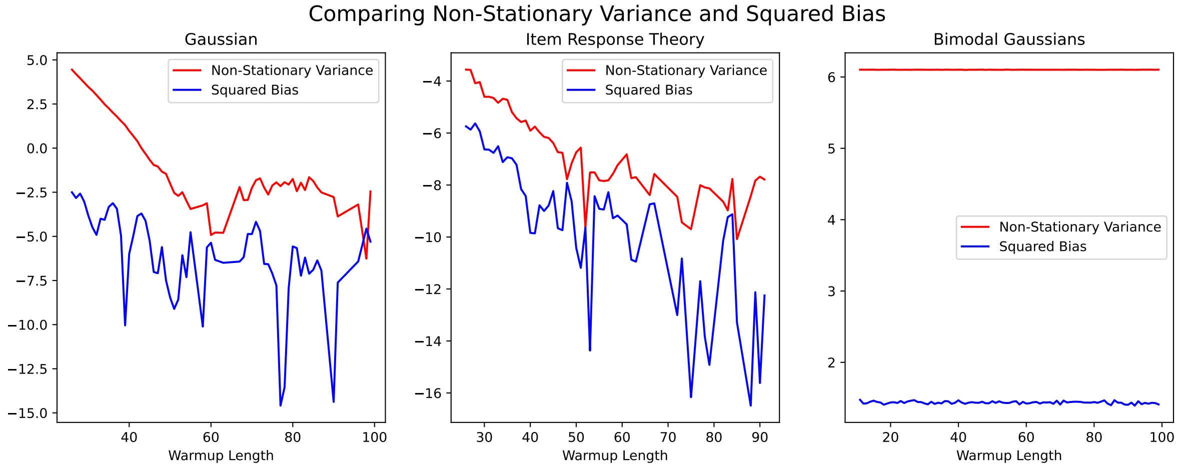

Figure 1: We compare the non-stationary bias and squared bias to see if it satisfies the bounds in Proposition 2.3. In each plot, the red line represents non-stationary variance and blue line represents squared bias. The y-axis is on the log scale. For each experiment, we used 32 superchains of 64 subchains, and sampled $N = 1$ points. We ran 1, 25, and 10 warmup phases respectively for unimodal Gaussian, IRT, and bimodal Gaussian. We then sampled 100 successive "sampling phase points". Note that when chains hit stationarity, i.e. non-stationary variance stops decreasing, there are computational issues with our estimates $(\widehat{B}_v, \widehat{W}_v)$ because our algorithm is moving around pseudo-randomly.

In order for Proposition 2.3 to be applied, we wish to check our assumption holds, that is Var $f(x_0) \geq (\mathbb{E}(b^2(x_0)))^2$. This is trivially true in our Gaussian case, because the initial bias is already 0. For IRT, we assumed a $N(0, 100)$ initial distribution, which has greater variance than the initial squared bias of around 2.25. Note that the IRT function in inference gym provides the ground truth and variance values. For bimodal Gaussians, we assumed an $N(0, 400)$ initial distribution, which has greater variance than the initial squared bias of 16.

4.9 Comparing Persistent Variance over Time

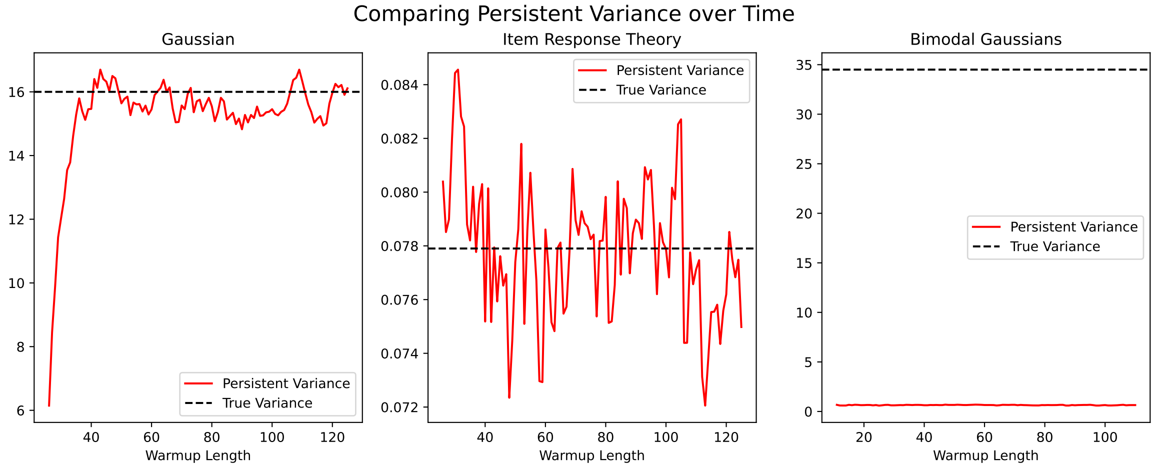

Figure 2: We compare the persistent variance over sampling iterations to see whether they follow the results in Proposition 2.5. In each plot, the red line represents persistent variance and dashed black line represents the true target variance, which we can calculate in our simulated examples. For each experiment, we used 32 superchains of 64 subchains, and sampled $N = 1$ points repeatedly. We ran 1, 25, and 10 warmup phases respectively for unimodal Gaussian, IRT, and bimodal Gaussian. We then sampled 100 successive "sampling phase points".

References

Del Moral, P., Doucet, A. and Jasra, A. (2006) Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B*, **68**, 411–536.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A. and Rubin, D. B. (2013) Basics of markov chain simulation. In *Bayesian Data Analysis*, chap. 11, 275–292. Chapman and Hall, CRC Press, 3 edn.

Gelman, A. and Rubin, D. B. (1992) Inference from iterative simulation using multiple sequences. *Statistical Sciences*, **7**, 457–472.

Hoffman, M. D. and Gelman, A. (2014) The no-u-turn sampler: Adaptively setting path length in hamiltonian monte carlo. *Journal of Machine Learning Research*, **15**, 1593–1623.

Hoffman, M. D., Radul, A. and Sountsov, P. (2021) An adaptive mcmc scheme for setting trajectory lengths in hamiltonian monte carlo. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics* (eds. A. Banerjee and K. Fukumizu), vol. 130 of *Proceedings of Machine Learning Research*.

Jacon, P. E., O'Leary, J. and Atchade, Y. F. (2020) Unbiased markov chain monte carlo methods with couplings. *Journal of the Royal Statistical Society: Series B*, **82**, 543–600.

Margossian, C. C., Hoffman, M. D., Sountsov, P., Riou-Durand, L., Vehtari, A. and Gelman, A. (2025) Nested $\hat{R}$: Assessing the convergence of markov chain monte carlo when running manny short chains. *Bayesian Analysis*, **20**, 1587–1614.

Moins, T., Arbel, J., Dutfoy, A. and Girard, S. (2022) A local version of r-hat for mcmc convergence diagnostic. *Journees de Statistique de la Societe Francaise de Statistique*, **53**, 1–6.

Neal, R. M. (2001) Annealed importance sampling. *Statistics and Computing*, **11**, 125–139.

— (2011) Mcmc using hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo* (eds. S. Brooks, A. Gelman, G. L. Jones and X.-L. Meng), chap. 5, 113–162. Chapman and Hall, CRC Press.

Sountsov, P., Carroll, C. and Hoffma, M. D. (2024) Running markov chain monte carlo on modern hardwareand software. URL: arXiv:2411.04260.

Stan Development Team (2025) *Stan Reference Manual*. URL: https://mc-stan.org.

Syed, S., Romaniello, V., Campbell, T. and Bouchard-Cote, A. (2021) Parallel tempering on optimized paths. In *Proceedings of the 38th International Conference on Artificial Intelligence and Statistics*, vol. 139 of *Proceedings of Machine Learning Research*.

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B. and Burkner, P.-C. (2021) Rank-normalization, folding, and localization: An improved $\hat{R}$ for assessing convergence of mcmc (with discussion). *Bayesian Analysis*, **16**, 667–718.

Wang, G. (2022) Exact convergence analysis of the independent metropolis-hastings algorithms. *Bernoulli*, **28**, 2012–2033.