# Post-Processing of MCMC

Leah South

Queensland University of Technology


16th July, 2021

# Setting

We can't simulate directly from $\pi$ so instead we've used another approach to get samples $\{\theta^{(i)}\}_{i=1}^N$, such as

- Standard MCMC
- Biased MCMC
- Any other mechanism for coming up with samples (even if the samples are correlated or weighted[1])

Our probability mass function for these samples[2] is denoted $Q$.

[1] Typically a simple extension but not covered here

[2] $Q = \frac{1}{N}\sum_{i=1}^N \delta_{\theta^{(i)}}$ where $\delta_{\theta^{(i)}}$ puts all probability mass on $\theta^{(i)}$

# Outline

We will cover

- **Sample quality:**
  - Measuring sample quality with the kernel Stein discrepancy
  - Testing if $\{\theta^{(i)}\}_{i=1}^{N} \sim \pi$ with kernel Stein goodness of fit tests
- **Choosing samples:**
  - Selecting a subset of samples with Stein thinning
- **Improving estimates:**
  - Improving upon $\widehat{\mathbb{E}_\pi[f(\theta)]} = \frac{1}{N}\sum_{i=1}^{N} f(\theta^{(i)})$ with Stein-based control variates.

# Measuring Sample Quality
## with the kernel Stein discrepancy (KSD)

# Setting

Our goal is usually to estimate posterior expectations, $\mathbb{E}_\pi[f(\theta)]$.

If we could calculate it, we would like to use

$$d_f(Q, \pi) = \mathbb{E}_Q[f(\theta)] - \mathbb{E}_\pi[f(\theta)]$$

as a measure of performance.

This may be fine if there is only interest in a single $f$ but it doesn't track non-convergence because $d_f(Q, \pi) \to 0$ doesn't imply $Q \to \pi$. Let's discuss why.

# Setting

To assess general sample quality, we would like to calculate

$$d_{\mathcal{F}}(Q, \pi) = \sup_{f \in \mathcal{F}} \left[ \mathbb{E}_Q[f(\theta)] - \mathbb{E}_\pi[f(\theta)] \right].$$

Note that sup is the supremum or the "least upper bound".

To be convergence-determining, we need $\mathcal{F}$ to be sufficiently large.

# Setting

This,

$$d_{\mathcal{F}}(Q, \pi) = \sup_{f \in \mathcal{F}} \left[ \mathbb{E}_Q[f(\theta)] - \mathbb{E}_\pi[f(\theta)] \right],$$

is the definition of an *integral probability metric (IPM)*. Some well-known IPM's include maximum mean discrepancy (MMD), Wasserstein distance and Kolmogorov distance.

This raises three follow-up questions

1. How can we circumvent $\mathbb{E}_\pi[f(\theta)]$?
2. How should we select our class of test functions?
3. Is our final discrepancy convergence-determining?

# Circumventing $\mathbb{E}_\pi[f(\theta)]$

There is a construction that gives us $\mathbb{E}_\pi[f(\theta)] = 0$ called the Stein operator $\mathcal{A}$.

If we pick $f = \mathcal{A}g$ then $\mathbb{E}_\pi[f(\theta)] = \mathbb{E}_\pi[\mathcal{A}g(\theta)] = 0$.

We will use the Langevin Stein operator[1]

$$\mathcal{A}g(\theta) = \nabla_\theta \cdot g(\theta) + \nabla \log \pi(\theta|y) \cdot g(\theta)$$

where $g$ is an $\mathbb{R}^d$-valued function.

[1] Stein (1972), Gorman and Mackey (2015)

# Circumventing $\mathbb{E}_\pi[f(\theta)]$

Other Stein operators can be developed using the generator method of Barbour (1988, 1990):

1. Identify a Markov process $(\theta_t)_{t\geq 0}$ with stationary distribution $\pi$
   E.g. the Langevin diffusion $\mathrm{d}\theta(t) = \nabla_\theta \log \pi(\theta|y)\mathrm{d}t/2 + \mathrm{d}b(t)$ where $b(t)$ is Brownian motion.
2. Under mild conditions, the infinitesimal generator

$$\mathcal{A}g(x) = \lim_{t\to 0} \frac{\mathbb{E}[g(\theta_t|\theta_0 = x] - g(x)}{t}$$

   satisfies $\mathbb{E}_\pi[\mathcal{A}g(\theta)] = 0$.

# Circumventing $\mathbb{E}_\pi[f(\theta)]$

The new discrepancy looks like this:

$$
\begin{aligned}
d_{\mathcal{F}}(Q, \pi) &= \sup_{f \in \mathcal{F}} \left[\mathbb{E}_Q[f(\theta)] - \mathbb{E}_\pi[f(\theta)]\right] \\
&= \sup_{g \in \mathcal{G}} \left[\mathbb{E}_Q[\mathcal{A}g(\theta)] - \mathbb{E}_\pi[\mathcal{A}g(\theta)]\right] \\
&= \sup_{g \in \mathcal{G}} \mathbb{E}_Q[\mathcal{A}g(\theta)] \\
&:= d_{\mathcal{G}}(Q, \mathcal{A}, \pi)
\end{aligned}
$$

Now we need to pick a large enough $\mathcal{G}$ for which we can compute this discrepancy.

# Class Selection (RKHS's)

Gorham and Mackey (2017) use *reproducing kernels* $k : \Theta \times \Theta \to \mathbb{R}$.

- For a kernel to be *reproducing*, it needs to be symmetric, i.e. $k(\theta, \theta') = k(\theta', \theta)$, and the kernel matrix $[K]_{ij} = k(\theta^{(i)}, \theta^{(j)})$ needs to be positive semi-definite. Some examples:
  - Polynomial kernel $k(\theta, \theta') = \sum_{j=1}^{J} P_j(\theta) P_j(\theta')$
  - Gaussian kernel: $k(\theta, \theta') = e^{-\|\theta - \theta'\|_2^2/(2\sigma)}$
- Such kernels induce a unique reproducing kernel Hilbert space $\mathcal{K}_k = \{h : h(\theta) = \sum_{i=1}^{N} c_i k(\theta^{(i)}, \theta)\}$ with norm $\|h\|_{\mathcal{K}_k} = \sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} c_i c_j k(\theta^{(i)}, \theta^{(j)})}$
  - Polynomial kernel $\mathcal{K}_k = \text{span}\{P_j\}_{j=1,\ldots,J}$
  - Gaussian kernel: $\mathcal{K}_k =$ (hard to write down but it has infinite span!)

# Class Selection (RKHS's)

These methods give us the ability to have an infinite span for $\mathcal{G}$ while being computable.

Our functions $g \in \mathcal{G}$ need to be $\mathbb{R}^d$-valued functions and the space needs to be constrained:

$$\mathcal{G} =: \{g = (g_1, \ldots, g_d) \mid \|v\| \leq 1 \text{ for } v_j = \|g_j\|_{\mathcal{K}_k}\}$$

where $g_i \in \mathcal{K}_k$ for $i = 1, \ldots, d$.

This video explains more about RKHS's:
`https://www.youtube.com/watch?v=KZZD5sBwGCA`

# Class Selection (RKHS's)

Under these choices, it can be shown that

$$d_{\mathcal{G}}(Q, \mathcal{A}, \pi) = \sup_{g \in \mathcal{G}} \mathbb{E}_Q[\mathcal{A}_\pi g(\theta)]$$
$$= \sqrt{\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} k_0(\theta^{(i)}, \theta^{(j)})},$$

where

$$k_0(\theta, \theta') = \mathcal{A}_\theta \mathcal{A}_{\theta'} k(\theta, \theta')$$
$$= \text{trace}(\nabla_\theta \nabla_{\theta'} k(\theta, \theta')) + \nabla_\theta \log \pi(\theta|y) \cdot \nabla_{\theta'} k(\theta, \theta')$$
$$+ \nabla_{\theta'} \log \pi(\theta'|y) \cdot \nabla_\theta k(\theta, \theta') + k(\theta, \theta') \nabla_\theta \log \pi(\theta|y) \cdot \nabla_{\theta'} \log \pi(\theta'|y).$$

This discrepancy is called the kernel Stein discrepancy (KSD).

# Convergence Determining?

The KSD can be computed at $\mathcal{O}(N^2)$ complexity given $\{\theta^{(i)}, \nabla_\theta \log \pi(\theta^{(i)}|y)\}_{i=1}^N$.

Now we have a computable discrepancy but when is it convergence determining? Stein's method requires thats we

- Lower bound the KSD to establish that $d_{\mathcal{G}}(Q, \mathcal{A}, \pi) \to 0$ only if $Q \to \pi$
- Upper bound the KSD to establish that $d_{\mathcal{G}}(Q, \mathcal{A}, \pi) \to 0$ if $Q \to \pi$

Both are needed. The quick answer is that it depends on the kernel choice, $k(\theta, \theta')$, and on the target, $\pi$.

# Convergence Determining?

The recommended kernel for detecting non-convergence is the inverse multiquadric kernel (IMQ),

$$k(\theta, \theta') = (c^2 + \|\theta - \theta'\|_2^2)^{\beta},$$

with $\beta = -0.5$ and $c = 1$.

# An Example

Problems with Gaussian & Matérn kernels: Figure 2 from Gorham & Mackey (2017)[1]:

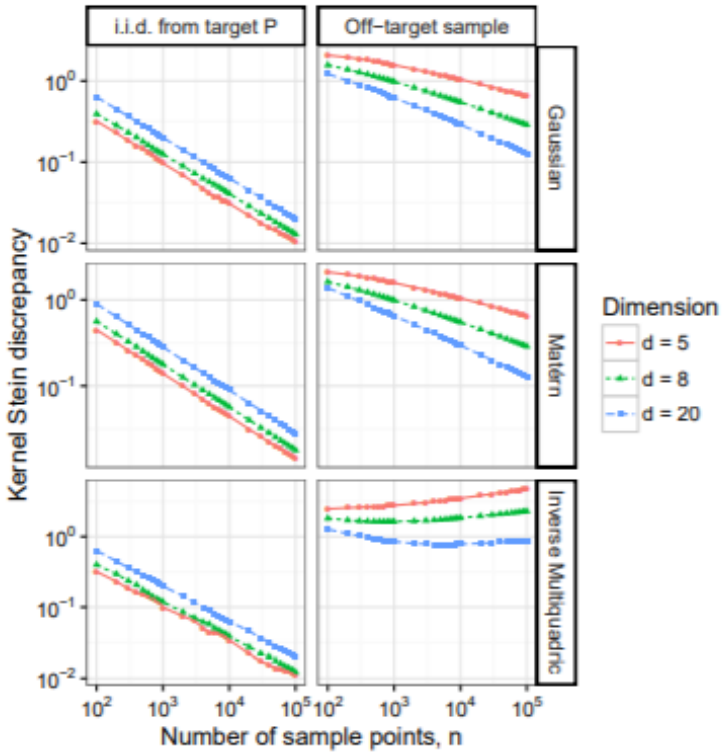



*Figure 2.* Gaussian and Matérn KSDs are driven to 0 by an off-target sequence that does not converge to the target $P = \mathcal{N}(0, I_d)$ (see Section 4.2). The IMQ KSD does not share this deficiency.

# An Example

Figure 1 from Nemeth & Fearnhead (2021)[1]:

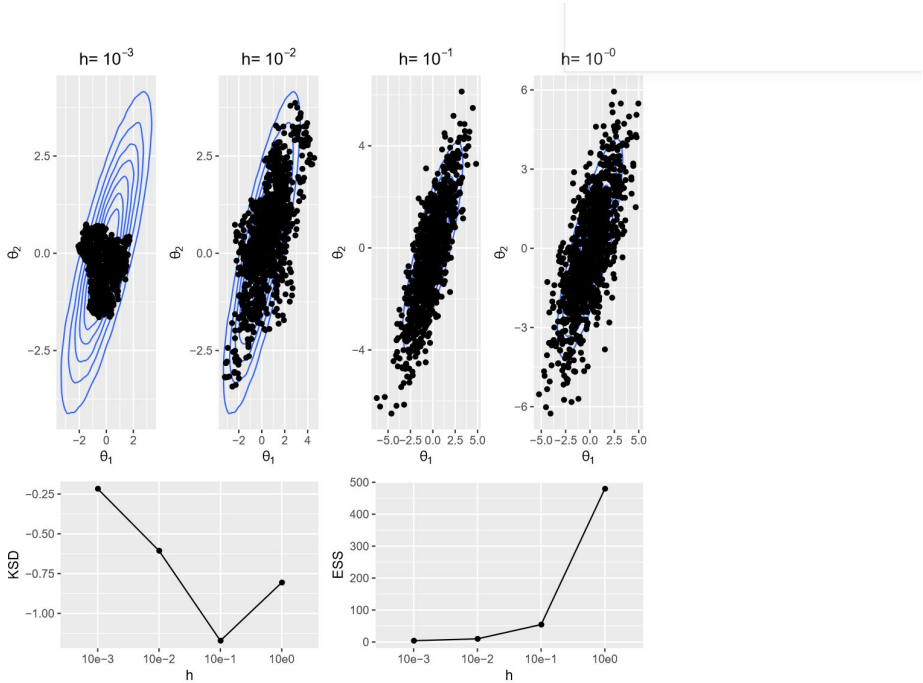



Figure 1: Top: Samples generated from the Langevin dynamics (7) are plotted over the bivariate Gaussian target. The samples are thinned to 1,000 for the ease of visualisation. Bottom: The kernel Stein discrepancy (log10) and effective sample size are calculated for each Markov chain with varying step size parameter $h$.

[1] Nemeth, C., & Fearnhead, P. (2021). Stochastic gradient Markov chain Monte Carlo. Journal of the American Statistical Association, 116(533), 433-450.

# Running Example

Let's take a look at the radiata pine example.

# Summary

- KSD can be used to compare the quality of samples from various methods
- This includes standard MCMC, biased MCMC, deterministic points, ...
- There is theory that tells us it's convergence-determining for some settings
  - Consider looking at the paper if you're not sure about your example
  - Even if the theory doesn't hold, it may still be practically useful.

# Testing if $\{\theta^{(i)}\}_{i=1}^{N} \sim \pi$
## with kernel Stein goodness of fit (GoF) tests

# A goodness of fit test

Chwialkowski et al (2016)[1] designed a method to test if $\{\theta^{(i)}\}_{i=1}^N$ are generated from $\pi$ using similar ideas to the KSD method.

We now think of the samples $\{\theta^{(i)}\}_{i=1}^N$ as coming from some unknown distribution $Q$. Therefore KSD is a random variable.

Applications: - biased MCMC (are the samples of the desired quality?) - standard MCMC - an alternative to standard normality tests - any time we want to test if $\{\theta^{(i)}\}_{i=1}^N \sim \pi$.

[1] Chwialkowski, K., Strathmann, H., & Gretton, A. (2016). A kernel test of goodness of fit. ICML

# The test

Our hypotheses are:

$$H_0 : \{\theta^{(i)}\}_{i=1}^N \sim \pi$$
$$H_a : \{\theta^{(i)}\}_{i=1}^N \not\sim \pi$$

The test statistic, its distribution and the procedure are slightly different depending on whether we do[1] or don't[2] have autocorrelation.

We estimate quantiles of the test statistic under $H_0$ using a form of bootstrapping.

[1] Liu, Q., Lee, J., & Jordan, M. (2016). A kernelized Stein discrepancy for goodness-of-fit tests. ICML

[2] Chwialkowski, K., Strathmann, H., & Gretton, A. (2016). A kernel test of goodness of fit. ICML

# Without Correlation

Our test statistic[3] is $\widetilde{\text{KSD}^2}$.

The kernel test of goodness of fit with significance level $\alpha$:

1. Calculate the test statistic $\text{KSD}^2$
2. Obtain $M$ sets of bootstrap samples $\{B_n^{(m)}\}$.
3. If $\text{KSD}^2$ exceeds the $1-\alpha$ empirical quantile of these samples, reject $H_0$.

[3] The tilde denotes a slight difference compared to KSD: we don't include $k(\theta, \theta)$ terms.

# With Correlation: Wild Bootstrap

To take into account correlation, the procedure involves using a Markov chain taking values in $\{-1, 1\}$ starting from $W_1 = 1$ and following

$$W_t = 1(U_t > a)W_{t-1} - 1(U_t < a)W_{t-1}$$

where $U_t \sim U(0, 1)$ and $a$ is the probability of $W_t$ changing sign. The tuning parameter $a$ should be chosen based on the level of autocorrelation ($a = 0.5$ for iid data).

The bootstrapped V-statistic is

$$B_n = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} W_i W_j k_0(x_i, x_j).$$

# With Correlation: Wild Bootstrap

The kernel test of goodness of fit with significance level $\alpha$:

1. Calculate the test statistic $\text{KSD}^2$
2. Obtain $M$ sets of wild bootstrap samples $\{B_n^{(m)}\}$.
3. If $KSD^2$ exceeds the $1-\alpha$ empirical quantile of these samples, reject $H_0$.

Intuition: There is some theory[1] that says $nB_n$ is a good approximation of $n\text{KSD}^2$ under the null. Under the alternative, $\text{KSD}^2 \to c$ for some $c > 0$ while $B_n \to 0$, resulting in almost sure rejection of the null.

[1] The theory assumes a basic regularity condition and holds for iid observations and geometrically ergodic Markov chains.

# In practice

In practice,

- The kernel $k(\theta, \theta')$ can affect the statistical power (prob. of rejecting $H_0$ when the alternative is true).
- Choosing the tuning parameter $a$ too high (underestimating the correlation) leads to overly conservative p-values but choosing it too low decreases power.

# In practice

Chwialkowski et al (2016) recommend a mixture of thinning and adjusting the tuning parameter *a*...

hypothesis. In a general, we recommend to thin a chain so that $Cor(X_t, X_{t-1}) < 0.5$, set $a_n = 0.1/k$, and run test with at least $max(500k, d100)$ data points, where $k < 10$, and d is data dimensionality[3].

[3] We recommend men should drink no more than 68 units of alcohol per week, no more than 34 units in any given day, and have at least 1 alcohol-free day.

# Choosing Which Samples to Use
## with Stein thinning

# Stein Thinning

The method of Riabiz et al (2020)[1] develop an approach to select a subset of $m$ samples from our $N$ samples ($m \ll N$) by minimising the KSD.

The goal is to select the subsample which minimises the KSD:

$$\begin{aligned} S &= \underset{S \subset \{1,\dots,N\}, |S|=m}{\arg\min} \; d_{\mathcal{G}}(Q_m, \mathcal{A}, \pi) \\ &= \underset{S \subset \{1,\dots,N\}, |S|=m}{\arg\min} \; d_{\mathcal{G}}(\frac{1}{m}\sum_{i \in S} \delta_{\theta^{(i)}}, \mathcal{A}, \pi) \end{aligned}$$

This is an expensive combinatorial problem. A greedy minimisation based on selecting one sample at a time works well.

[1] Riabiz, M., Chen, W., Cockayne, J., Swietach, P., Niederer, S. A., Mackey, L., & Oates, C. (2020). Optimal thinning of MCMC output. arXiv preprint arXiv:2005.03952.

# Stein Thinning

I recommend going to the website

`http://stein-thinning.org/`

Stein thinning is bias-removing in some contexts (we can get KSD $\to 0$ as $m, n \to \infty$ for certain non $\pi$-invariant Markov chains).

We can also use all of the samples or a weighted set of samples (related to the next section). Rob Salomone (next week) has done some related work!

# Improving Estimates
## using Stein-based control variates

# Setting

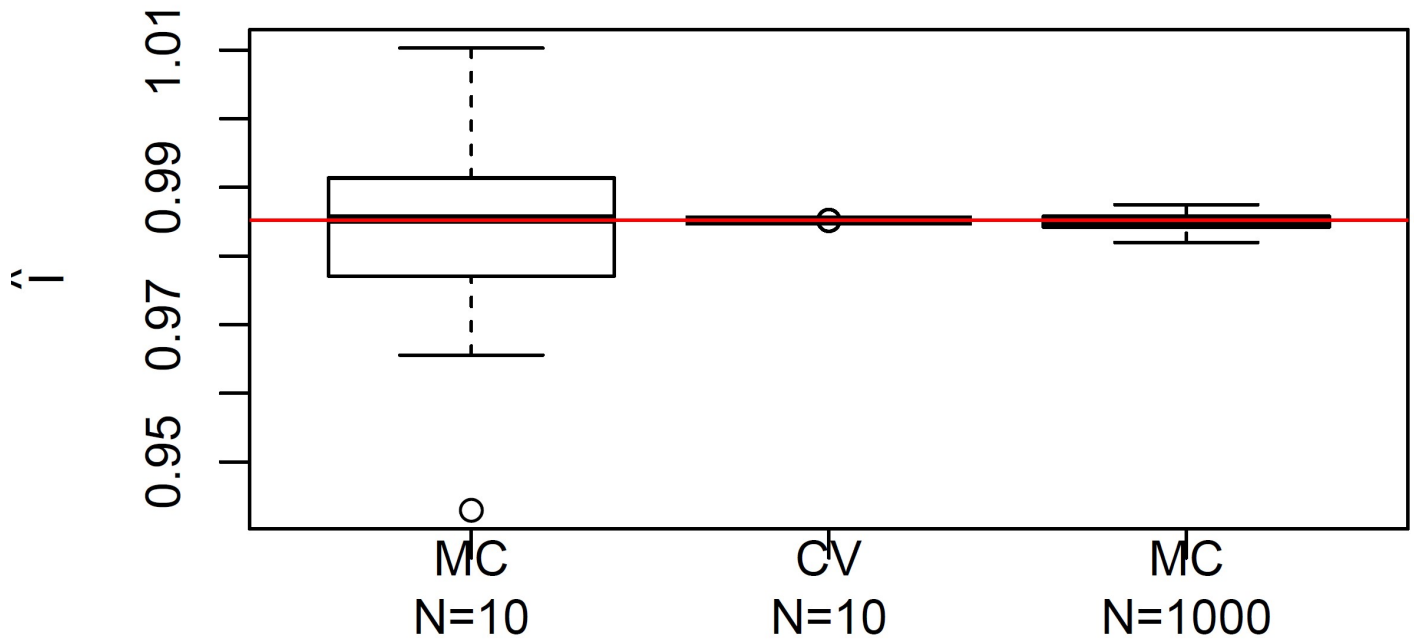
We have samples $\{\theta^{(i)}\}_{i=1}^N$ and gradients $\{\nabla_\theta \log \pi(\theta^{(i)}|y)\}_{i=1}^N$ and we want to evaluate

$$I = \mathbb{E}_\pi[f(\theta)] = \int_\Omega f(\theta)\pi(\theta|y)d\theta.$$

We now focus on improving upon the vanilla estimate $\hat{I} = \frac{1}{N}\sum_{i=1}^N f(\theta^{(i)})$, which often has high variance and/or bias.

# This Section

Example comparisons




All methods I'll talk about are in the R package ZVCV.

# Control Variates

In control variates, we replace the Monte Carlo estimator with

$$\hat{I}_{\text{CV}} = \frac{1}{N}\sum_{i=1}^{N}\left[f(\theta^{(i)}) - \tilde{f}(\theta^{(i)})\right] + \int_{\Omega}\tilde{f}(\theta)\pi(\theta|y)\mathrm{d}\theta.$$

# Control Variates

The intuition for i.i.d. samples from $\pi$:

- $\mathbb{E}_\pi[\hat{I}_{\text{CV}}] = I$
- $\mathbb{V}_\pi[\hat{I}_{\text{CV}}] \ll \mathbb{V}_\pi[\hat{I}_{\text{MC}}]$ because

$$\mathbb{V}_\pi[f(\theta) - \tilde{f}(\theta)] = \mathbb{V}_\pi[f(\theta)] + \mathbb{V}_\pi[\tilde{f}(\theta)] \underline{\underline{-2\mathbb{COV}_\pi[f(\theta), \tilde{f}(\theta)]}}$$

# Stein Control Variates

Desiderata for control variates:

1. Ability to evaluate $\mathbb{E}_\pi[f(\theta)] = \int_\Omega \tilde{f}(\theta)\pi(\theta|y)\mathrm{d}\theta$
2. Easy to compute $\tilde{f}$ at $\{\theta^{(i)}\}_{i=1}^N$
3. Reduced variance: $\sigma^2(f - \tilde{f}) \ll \sigma^2(f)$

# Stein Control Variates

Desiderata for control variates:

1. Ability to evaluate $\mathbb{E}_\pi[f(\theta)] = \int_\Omega \tilde{f}(\theta)\pi(\theta|y)\mathrm{d}\theta$
2. Easy to compute $\tilde{f}$ at $\{\theta^{(i)}\}_{i=1}^N$
3. Reduced variance: $\sigma^2(f - \tilde{f}) \ll \sigma^2(f)$

Control variates based on the Langevin Stein operator can satisfy these.

# Stein Control Variates

Oates et al (2017) propose to use

$$\tilde{f}(\theta) = a + \mathcal{A}g(\theta),$$

where $a \in \mathbb{R}$, $g$ is a user-specified function and $\mathcal{A}$ is a Stein operator:

$$\mathcal{A}g(\theta) = \nabla_\theta \cdot g(\theta) + \nabla \log \pi(\theta|y) \cdot g(\theta) \text{ for } g : \Theta \to \mathbb{R}^d$$
$$\mathcal{A}g(\theta) = \Delta_\theta g(\theta) + \nabla \log \pi(\theta|y) \cdot \nabla_\theta g(\theta) \text{ for } g : \Theta \to \mathbb{R}$$

which signifies that $\mathbb{E}_\pi[\mathcal{A}g(\theta)] = 0$ under regularity conditions on $g$ and $\pi$.

The challenge is to choose $g$ so that $\sigma^2(f - \tilde{f}) \ll \sigma^2(f)$. The key is to select $g \in \mathcal{G}$ by optimising some criterion.

# Zero-Variance Control Variates: Polynomial $\mathcal{G}$

Assaraf and Caffarel (1999) and Mira et al. (2013) choose $\mathcal{G}$ to be the class of $r$th order polynomials.

# Zero-Variance Control Variates: Polynomial $\mathcal{G}$

Assaraf and Caffarel (1999) and Mira et al. (2013) choose $\mathcal{G}$ to be the class of $r$th order polynomials.

This leads to

$$\begin{aligned}\tilde{f}(\theta) &= a + \mathcal{A}P(\theta) \\ &= a + \sum_{j=1}^{J} \beta_j \mathcal{A}P_j(\theta), \\ &= a + \beta^T x(\theta),\end{aligned}$$

where $\beta \in \mathbb{R}^J$ is the polynomial coefficients and $x(\theta) \in \mathbb{R}^J$ is a vector of terms involving $\theta$ and $\nabla_\theta \log \pi(\theta|y)$.

# Zero-Variance Control Variates: Polynomial $\mathcal{G}$

A common approach to estimating $(a, \beta)$ is OLS,

$$(\hat{a}, \hat{\beta}) \in \underset{\substack{a \in \mathbb{R} \\ \beta \in \mathbb{R}^J}}{\arg\min} \sum_{i=1}^{N} \left[ f(\theta^{(i)}) - a - \beta^\top x(\theta^{(i)}) \right]^2,$$

so $(\hat{a}, \hat{\beta})^\top = (X^\top X)^{-1} X^\top f$ and the resulting estimator is

$$\hat{I}_{\text{ZVCV}} = \frac{1}{N} \sum_{i=1}^{N} \left[ f(\theta^{(i)}) - \hat{\beta}^\top x(\theta^{(i)}) \right] = \hat{a}.$$

# Zero-Variance Control Variates: Polynomial $\mathcal{G}$

Let's implement this to estimate some expectations for a unit normal distribution $\pi(\theta) = \frac{1}{\sqrt{2\pi}} e^{-(\theta-5)^2/2}$ where $\{\theta^{(i)}\}_{i=1}^{10} \overset{iid}{\sim} \pi$.

# Control Functionals: RKHS $\mathcal{G}$

Oates et al. (2017) set $\mathcal{G}$ to be a reproducing kernel hilbert space $\mathcal{K}_k$.

$$(\hat{a}, \hat{g}) \in \underset{\substack{a \in \mathbb{R} \\ g \in \mathcal{K}_k}}{\arg\inf} \frac{1}{N} \sum_{i=1}^{N} \left[ f(\theta^{(i)}) - a - \mathcal{A}g(\theta^{(i)}) \right]^2 + \lambda \|g\|^2_{\mathcal{K}_k}$$

In practice, performing CF amounts to choosing a kernel $k(\theta, \theta')$, estimating its parameters and solving the $N$ by $N$ linear system:

$$\hat{I}_{\text{CF}} = \frac{1^T (K_0 + \lambda N I)^{-1} f}{1^T (K_0 + \lambda N I)^{-1} 1}$$

where $f_i = f(\theta^{(i)})$ and $[K_0]_{i,j} = \mathcal{A}_x \mathcal{A}_y k(x, y)|_{x=\theta^{(i)}, y=\theta^{(j)}}$.

The estimator can also be thought of as the posterior mean with a Gaussian process prior $f \sim \mathcal{GP}(0, K_0)$.

# Semi-Exact Control Functionals

The interpolant takes the form

$$\tilde{f}(\theta) = b_0 + \underbrace{\sum_{j=1}^{J-1} b_j \mathcal{A} P_j(\theta)}_{\text{related to ZV-CV}} + \underbrace{\sum_{i=1}^{N} a_i k_0(\theta, \theta^{(i)})}_{\text{related to CF}} .$$

where $\{P_j\}_{j=1}^{J-1}$ is a polynomial basis

# Semi-Exact Control Functionals

The interpolant takes the form

$$\tilde{f}(\theta) = b_0 + \underbrace{\sum_{j=1}^{J-1} b_j \mathcal{A}P_j(\theta)}_{\text{related to ZV-CV}} + \underbrace{\sum_{i=1}^{N} a_i k_0(\theta, \theta^{(i)})}_{\text{related to CF}}.$$

where $\{P_j\}_{j=1}^{J-1}$ is a polynomial basis and the coefficients $a$ and $b$ are selected such that

1. $\tilde{f} = f$ whenever $f \in \text{span}(1, \mathcal{A}P_1, \ldots, \mathcal{A}P_{J-1})$
2. $\tilde{f}(\theta^{(i)}) = f(\theta^{(i)})$ for $i = 1, \ldots, N$

This is estimating the posterior mean using a GP prior $f \sim \mathcal{GP}(\sum_{j=1}^{J-1} b_j \mathcal{A}P_j, K_0)$

# Semi-Exact Control Functionals

Estimation of $a$ and $b$ amounts to solving the linear system

$$\begin{bmatrix} K_0 & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix},$$

where

$$P = \begin{bmatrix} 1 & \mathcal{A}P_1(\theta^{(1)}) & \cdots & \mathcal{A}P_J(\theta^{(1)}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \mathcal{A}P_1(\theta^{(N)}) & \cdots & \mathcal{A}P_J(\theta^{(N)}) \end{bmatrix} \text{ and } \quad f = \begin{bmatrix} f(\theta^{(1)}) \\ \vdots \\ f(\theta^{(N)}) \end{bmatrix}.$$

# Semi-Exact Control Functionals

The estimator is

$$\begin{aligned}\hat{I}_{\text{SECF}} &= \frac{1}{N}\sum_{i=1}^{N}\left[f(\theta^{(i)}) - \tilde{f}(\theta^{(i)})\right] + \int_{\Omega}\tilde{f}(\theta)\pi(\theta|y)\mathrm{d}\theta \\ &= \int_{\Omega}\tilde{f}(\theta)\pi(\theta|y)d\theta \\ &= \hat{b}_0 \\ &= e_1^T(P^T K_0^{-1} P)^{-1} P^T K_0^{-1} f\end{aligned}$$

# Summary of Theory

- The polynomial approaches (ZV-CV and SECF) are exact for all $r$th order polynomials in the Bernstein-von-Mises (Bayesian big data) limit
- The kernel-based approaches (CF and SECF) offer consistency:

$$\hat{I}_{\text{SECF}} \to I \text{ in probability as } N \to \infty,$$

under certain conditions even in the biased sampling setting where the Markov chain is not $\pi$-invariant and the potential for improved convergence rates.

# Example Bias-Correction

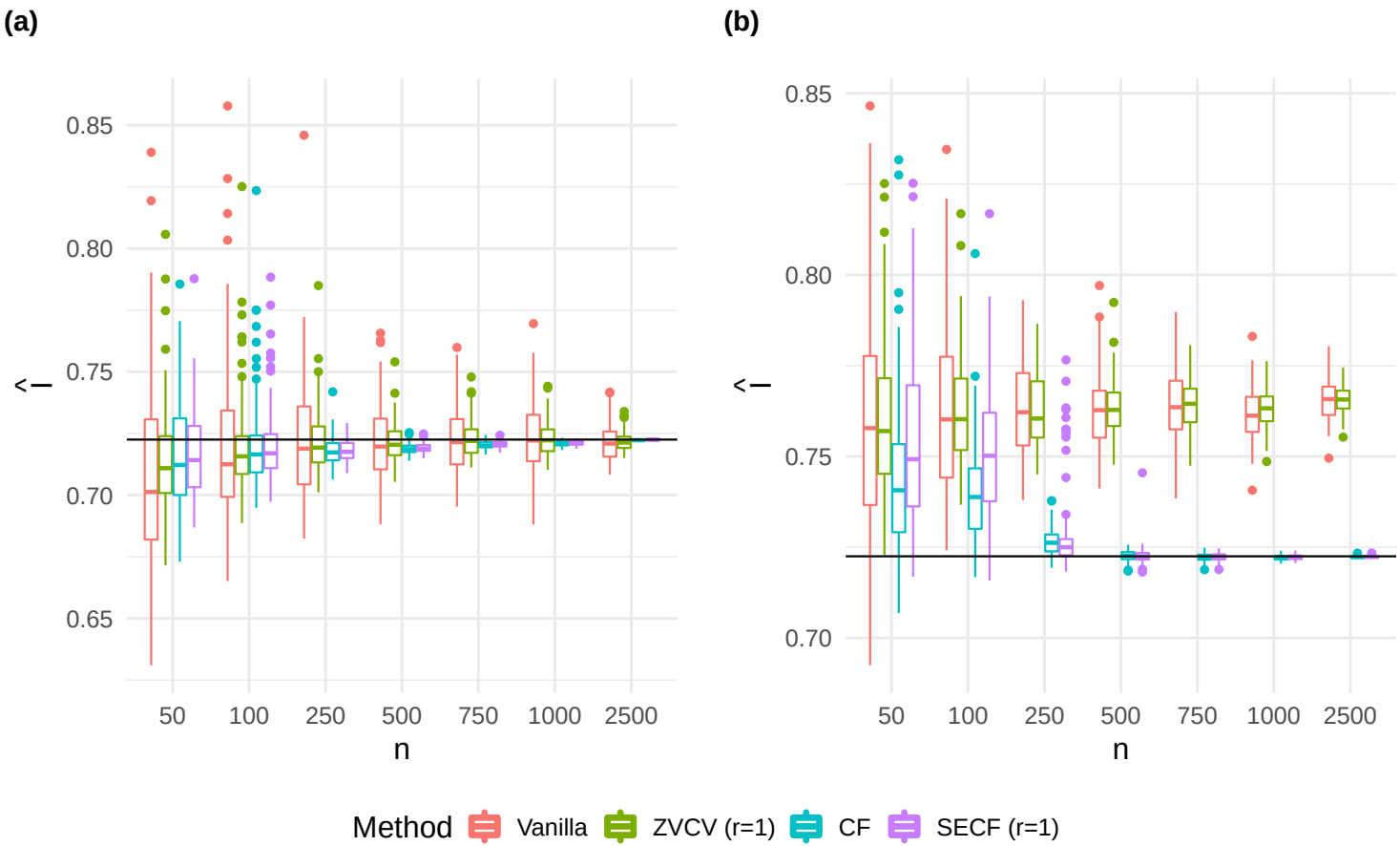



Figure: Boxplots of 100 estimates of an expectation when (a) MALA and (b) ULA are used for sampling. The black horizontal line is the gold standard.

# R package

ZVCV is available on CRAN.

ZVCV_package {ZVCV} R Documentation

## Zero-Variance Control Variates

### Description

This package can be used to perform post-hoc variance reduction of Monte Carlo estimators when the derivatives of the log target are available. The main functionality is available through the following functions. All of these use a set of $N$ $d$-dimensional samples along with the associated derivatives of the log target. You can evaluate posterior expectations of $k$ functions.

- `zvcv`: For estimating expectations using (regularised) zero-variance control variates (ZV-CV, Mira et al, 2013; South et al, 2018). This function can also be used to choose between various versions of ZV-CV using cross-validation.
- `CF`: For estimating expectations using control functionals (CF, Oates et al, 2017).
- `SECF`: For estimating expectations using semi-exact control functionals (SECF, South et al, 2020).
- `aSECF`: For estimating expectations using approximate semi-exact control functionals (aSECF, South et al, 2020).
- `CF_crossval`: CF with cross-validation tuning.
- `SECF_crossval`: SECF with cross-validation tuning.
- `aSECF_crossval`: aSECF with cross-validation tuning.

ZV-CV is exact for polynomials of order at most `polyorder` under Gaussian targets and is fast for large $N$ (although setting a limit on `polyorder` through `polyorder_max` is recommended for large $N$). CF is a non-parametric approach that offers better than the standard Monte Carlo convergence rates. SECF has both a parametric and a non-parametric component and it offers the advantages of both for an additional computational cost. The cost of SECF is reduced in aSECF using nystrom approximations and conjugate gradient.

### Helper functions

# Running Example

Let's take a look at the radiata pine example.

# Summary

# Summary

We have covered

- **Sample quality:**
  - Measuring sample quality with the kernel Stein discrepancy
  - Testing if $\{\theta^{(i)}\}_{i=1}^N \sim \pi$ with kernel Stein goodness of fit tests
- **Choosing samples:**
  - Selecting a subset of samples with Stein thinning
- **Improving estimates:**
  - Improving upon $\widehat{\mathbb{E}_\pi[f(\theta)]} = \frac{1}{N}\sum_{i=1}^N f(\theta^{(i)})$ with Stein-based control variates.

All methods used $\{\theta^{(i)}, \nabla_\theta \log \pi(\theta^{(i)}|y)\}_{i=1}^N$.

# Advert

Do you want 80 hours of paid RA work?[1]

- Write a vignette on ZVCV + RStan
- Try out some of these methods
- ...

I'm also doing research on a fast alternative to KSD.

[1] QUT may make this difficult for external applicants, sorry.

# Additional Resources

- This talk by the creator of the KSD: `https://www.youtube.com/watch?v=nVTL2yH_VOk`
- The review paper and references within: South, L. F., Riabiz, M., Teymur, O., & Oates, C. (2021). Post-Processing of MCMC. *arXiv preprint arXiv:2103.16048*.
- The review paper and references within: Anastasiou, A., Barp, A., Briol, F. X., Ebner, B., Gaunt, R. E., Ghaderinezhad, F., Gorham, J., Gretton, A., Ley, C., Liu, Q., Mackey, L., Oates, C. J., Reinert, G. & Swan, Y. (2021). Stein's Method Meets Statistics: A Review of Some Recent Developments. *arXiv preprint arXiv:2105.03481*.

# Boundary/tail condition

$$\begin{aligned}
\mathbb{E}_\pi[\mathcal{A}g(\theta)] &= \int_\Omega \mathcal{A}g(\theta)\pi(\theta|y)\mathrm{d}\theta \\
&= \int_\Omega \nabla_\theta \cdot (\pi(\theta|y)\nabla_\theta g(\theta))\,\mathrm{d}\theta \\
&= \oint_{\delta\Omega} \pi(\theta|y)\nabla_\theta g(\theta) \cdot n(\theta) S(\mathrm{d}\theta)
\end{aligned}$$

where the last line is by the divergence theorem.

# Convergence Determining?

**Upper bound**: Under regularity conditions, we have that $d_{\mathcal{G}}(Q, \mathcal{A}, \pi) \to 0$ when the Wasserstein distance between $Q$ and $\pi$ goes to zero.
**Lower bound**: Kernels with rapidly decaying tails can lead to $d_{\mathcal{G}}(Q, \mathcal{A}, \pi) \to 0$ even when $Q \not\to \pi$.

The conditions required for $d_{\mathcal{G}}(Q, \mathcal{A}, \pi) \to 0$ only if $Q \to \pi$ are:

- $d = 1$: $\pi$ is "distantly dissipative"[1] and $k$ is sufficiently regular[2]
- $d > 1$: All of the above plus $k$ does not decay more rapidly than the score function grows.

KSD fails to detect non-convergence for bounded scores.

[1] A relaxation of log-concavity covering many distributions of interest including Bayesian logistic & student's $t$ regression with Gaussian priors, Gaussian mixtures with common variance and more

[2] $k(\theta, \theta') = \Phi(\theta - \theta')$ for $\Phi$ with continuous second order derivatives and a non-vanishing generalised Fourier transform.