Some applications of MCMC perturbations in high-dimensional problems

Anirban Bhattacharya
Department of Statistics, Texas A&M University
Joint work with James Johndrow and Paolo Orenstein

April 10, 2019
BIRS workshop on "New and Evolving Roles of Shrinkage in Large-Scale Prediction and Inference"

Introduction

- Aim: develop scalable MCMC algorithms for large (N, p) regression with continuous shrinkage priors
- Compute the posterior expectation & marginal posterior densities for the coefficients
- We won't get this from optimization, also not a convex problem in many cases
- For concreteness, we focus on the horseshoe prior of Carvalho et al. (2010) - theoretical support + empirical performance
- Basic ingredients extend to more general Gaussian variance mixtures as well as two-component mixtures like the spike-and-slab lasso (Rockova & George, 2014)

Approximations in MCMC

- Our proposed algorithm introduces certain approximations at each MCMC step - approximate certain expensive matrix multiplications
- Leads to substantial computational advantages
- How to quantify the effect of such approximations?
- Perturbation theory for MCMC algorithms (Alquier et al. 2014, Rudolf & Schweizer (2018), Johndrow & Mattingley (2018)...)
- A new general result + bounds on approximation error for our algorithm

Other applications

- Similar ideas applicable to a host of other high-dimensional problems
- Ongoing work: approximate sampling from truncated multivariate normals with applications to problems with constrained parameters
- Replace the hard constraints with "soft" versions

Bayesian shrinkage: motivation and background

"Global-local" shrinkage priors

- Consider a Gaussian linear model

$$z = W\beta + \varepsilon, \quad \varepsilon \sim \mathrm{N}(0, \sigma^2 I_N)$$

  where W is $N \times p$, with N, p both possibly large

- The basic form of the prior is

$$\beta_j \mid \sigma, \xi, \eta \overset{ind}{\sim} N(0, \sigma^2 \xi^{-1} \eta_j^{-1})$$

- The $\eta_j^{-1/2}$ are the "local scales" and $\xi^{-1/2}$ the "global scale"
- A popular choice for $\pi(\xi, \eta)$ is the "Horseshoe" (Carvalho et al. 2010)

$$\eta_j^{-1/2} \overset{ind.}{\sim} \mathrm{Cauchy}_+(0, 1), \quad \xi^{-1/2} \sim \mathrm{Cauchy}_+(0, 1)$$

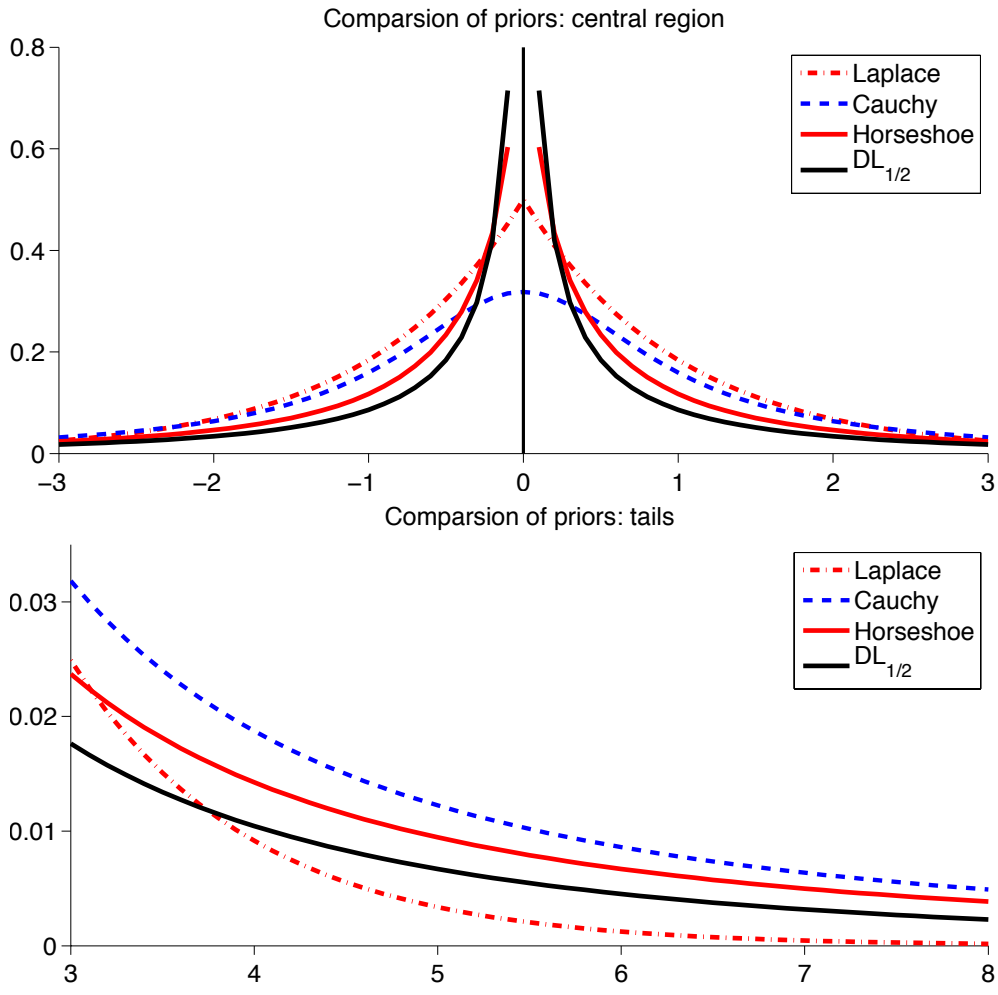
"Global-local" shrinkage priors

- The basic form of the prior is

$$\beta_j \mid \sigma, \xi, \eta \overset{ind}{\sim} N(0, \sigma^2 \xi^{-1} \eta_j^{-1})$$

- The $\eta_j^{-1/2}$ are the "local scales" and $\xi^{-1/2}$ the "global scale"
- Only global scale $\Rightarrow$ ridge type shrinkage
- Local scales help adapt to sparsity
- The global scale $\xi^{-1/2}$ controls **how many** β_j are signals, and $\eta_j^{-1/2}$ control **their identities**

Continuous shrinkage via one group models



Computational challenges

MCMC review

- Basic idea of MCMC: construct a Markov transition kernel $\mathcal{P}$ with invariant measure the posterior, i.e. $\mu\mathcal{P} = \mu$ where μ is the posterior measure

- Then approximate

$$\mu\varphi \equiv \int \varphi(x)\mu(dx) \approx n^{-1} \sum_{k=0}^{n-1} \varphi(X_k)$$

for $X_k \sim \nu\mathcal{P}^{k-1}$

Computational cost

- What is the computational cost? Two factors
  1. The cost of taking one step with $\mathcal{P}$
  2. How long the Markov chain needs to be to make approximation "good"

Computational cost per step

- Perform various matrix operations - multiplication, solving linear systems, Cholesky etc
- Sample from complicated distributions (such as truncated MVNs)

Length of path required

- How long of a Markov chain do we need to approximate the posterior well?
- Informally, the higher the autocorrelations, the longer the path we will need
- Another performance metric: **effective sample size**, the equivalent number of **independent** samples (larger is better)

Computational cost

- What is the computational cost? Two factors
  1. The cost of taking one step with $\mathcal{P}$
  2. How long the Markov chain needs to be to make approximation "good"
- For the horseshoe, **both of these present challenges**
- Linear algebra with large matrices
- High autocorrelation

Algorithmic developments

Gibbs sampling for the horseshoe

State space $\mathbf{X} = \mathbb{R}^p \times \mathbb{R}_+^p \times \mathbb{R}_+ \times \mathbb{R}_+$ with state-vector $x = (\beta, \eta, \xi, \sigma^2)$. Let $D = \text{diag}(\eta_j^{-1})$

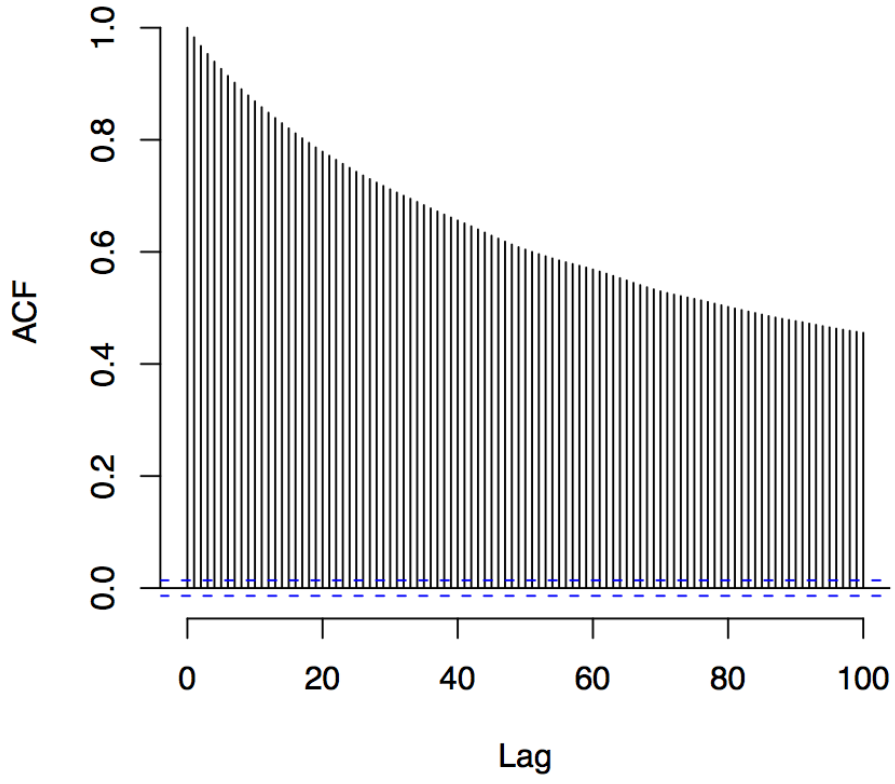
Typical computational approach: blocked Gibbs sampling (Polson et al (2012))

$$\begin{aligned} \eta \mid \beta, \sigma^2, \xi, z \\ \xi \mid \beta, \sigma^2, \eta, z \\ (\beta, \sigma^2) \mid \eta, \xi, z \end{aligned}$$

The algorithm is known to exhibit poor mixing for ξ (Polson et al. (2012))

Mixing issues

Evidence of poor mixing for ξ



Remedy: Johndrow, Orenstein, B. (2018+)

- Our approach: more blocking

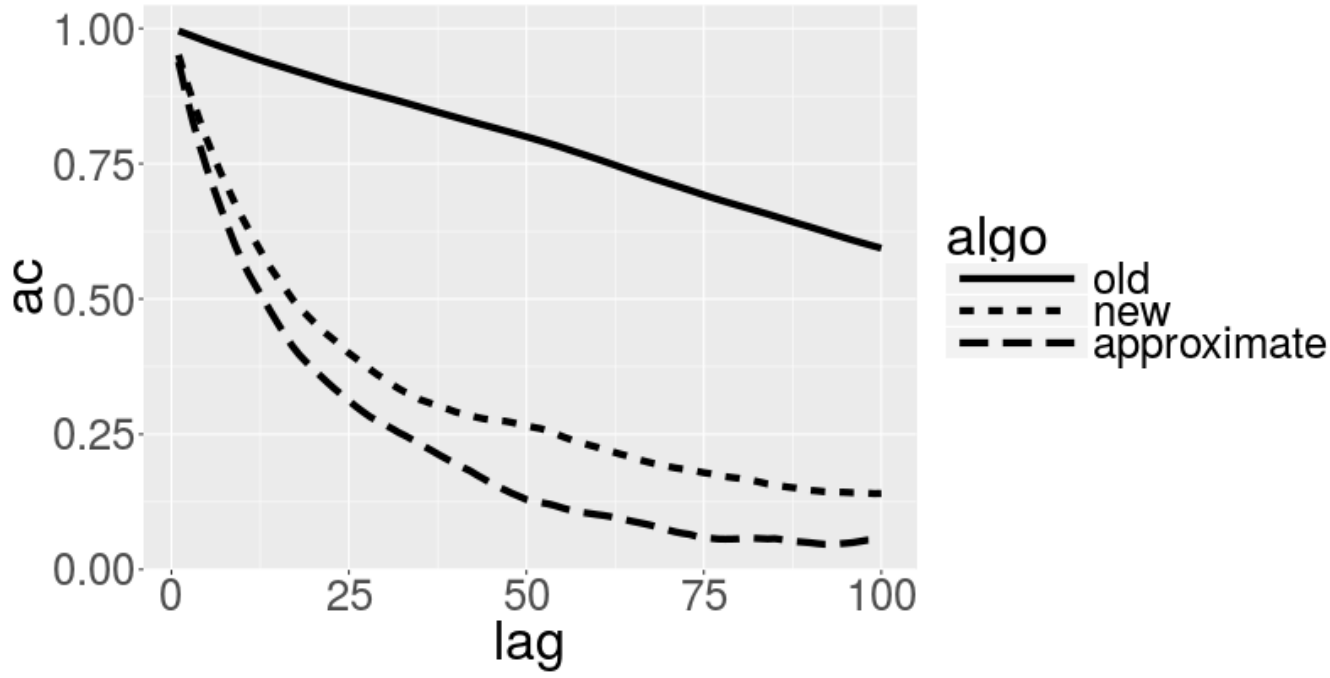
$$(\beta, \sigma^2, \xi) \mid \eta, z$$
$$\eta \mid (\beta, \sigma^2, \xi), z$$

- The first step is done by sampling

$$\xi \mid \eta, z \Rightarrow \text{Metropolis-within-Gibbs}$$
$$\sigma^2 \mid \eta, \xi, z \Rightarrow \text{sample from Inverse-Gamma}$$
$$\beta \mid \eta, \sigma^2, \xi, z \Rightarrow \text{sample from MVN}$$

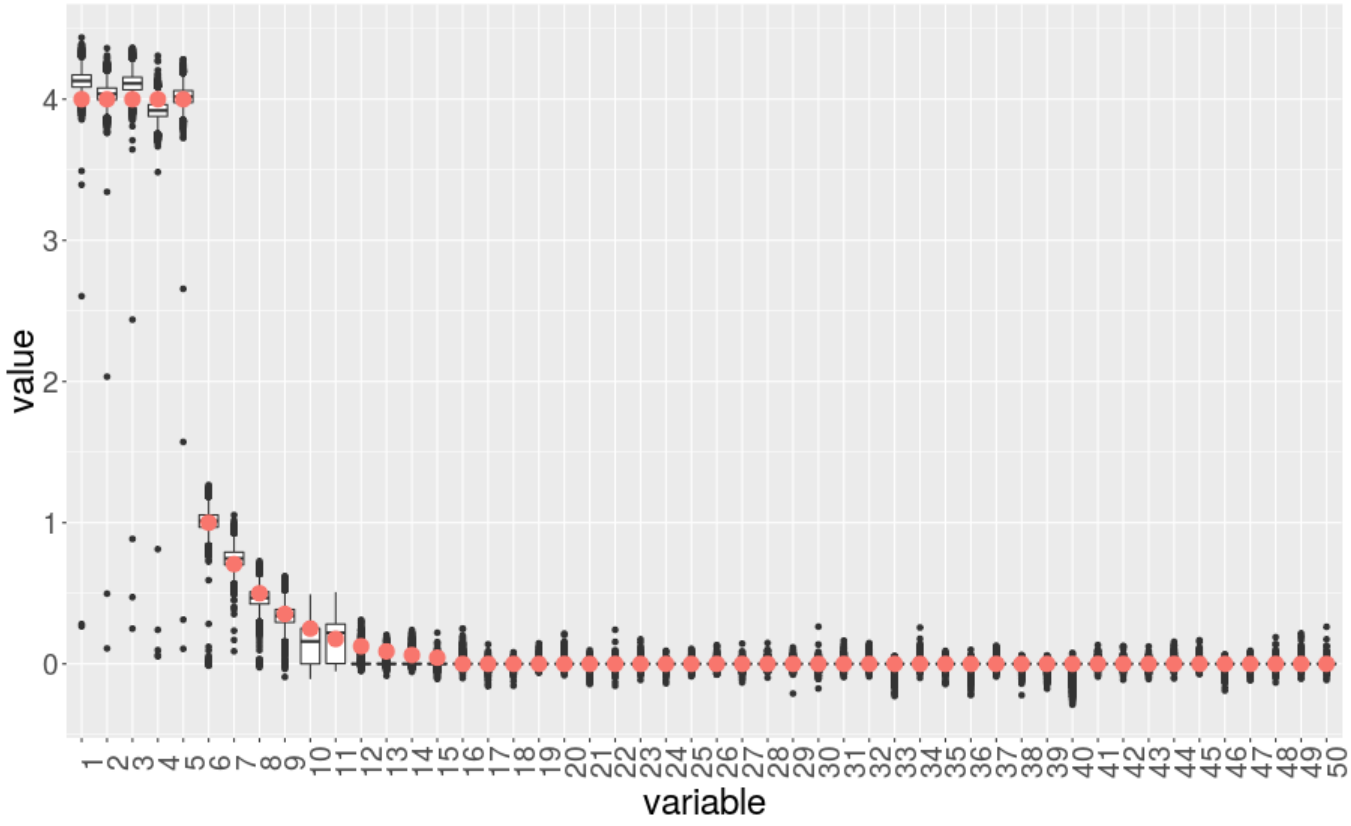
- The second step is done by sampling η_js independently using an accurate rejection sampler

Results: Autocorrelations for ξ

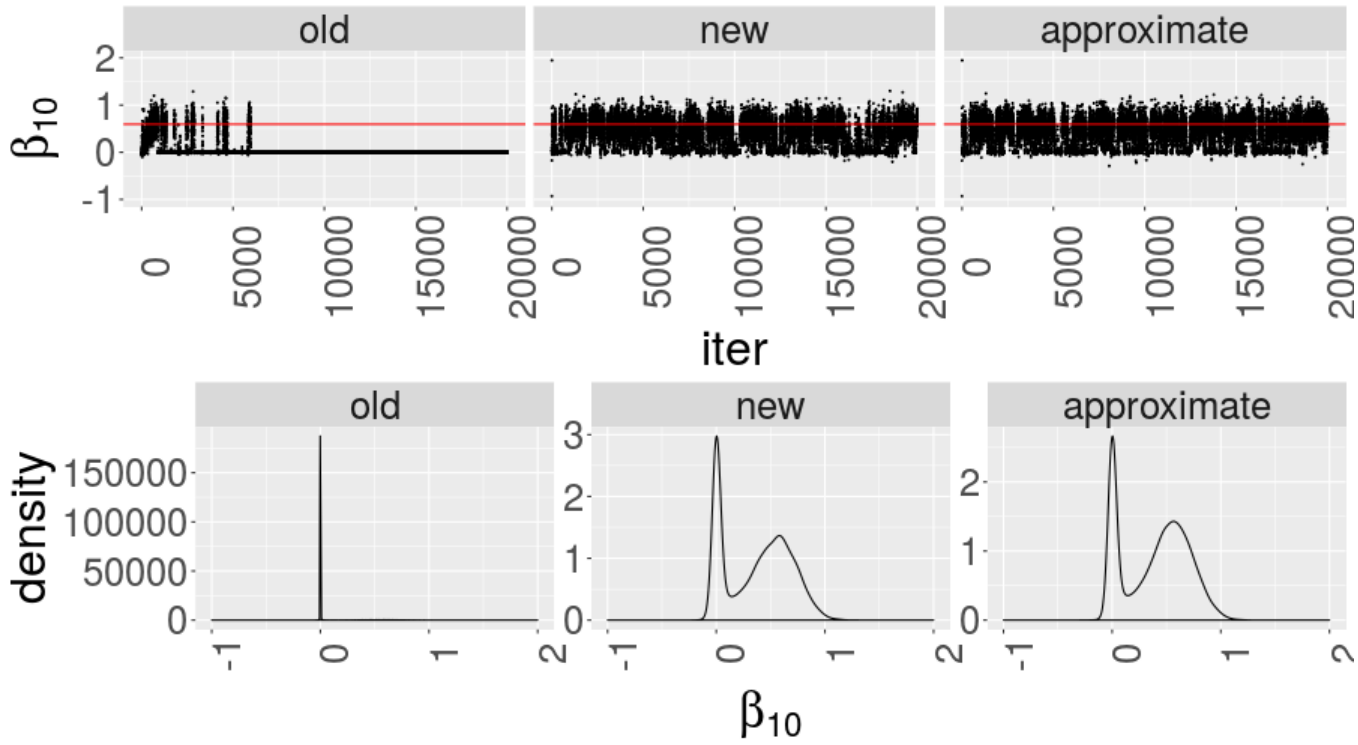


Example

Simulation with $N = 2,000$ and $p = 20,000$: first 50 β_j (rest are zero). Posterior medians, 95 percent credible intervals, along with the truth.



Results: Accuracy



The old algorithm often failed to identify components of β with bimodal marginals

This conveys uncertainty about whether β_j is a true signal, which is one of the nice features of taking a Bayesian approach to multiple testing

Geometric ergodicity

Theorem. The blocked sampler above is **geometrically ergodic**.

Verify standard drift + minorization condition

1. **Foster–Lyapunov condition.** There exists a function $V : \mathbf{X} \to [0, \infty)$ and constants $0 < \gamma < 1$ and $K > 0$ such that

$$(\mathcal{P}V)(x) \equiv \int V(y)\mathcal{P}(x, dy) \leq \gamma V(x) + K.$$

2. **Minorization.** For every $R > 0$ there exists $\alpha \in (0, 1)$ (depending on R) such that, for $\mathcal{S}(R) = \{x : V(x) < R\}$,

$$\sup_{x,y \in \mathcal{S}(R)} \|\mathcal{P}(x, \cdot) - \mathcal{P}(y, \cdot)\|_{TV} \leq 2(1 - \alpha).$$

Geometric ergodicity

Harris' Theorem (Meyn & Tweedie; Hairer & Mattingley).
Let $x = (\eta, \xi, \sigma^2, \beta)$ and $\mathcal{P}$ the transition kernel. Also, let μ be the invariant measure, i.e., the posterior.

Together, (1) and (2) imply,

$$\sup_{|\varphi|<1+V} \int \varphi(y)(\mathcal{P}^n(x,y) - \mu(y))\, dy \leq C\bar{\alpha}^n V(x),$$

for some $\bar{\alpha} \in (0,1)$.

Geometric convergence in a weighted total variation norm

$(1-\bar{\alpha})$ the spectral gap - larger implies faster convergence

The exact algorithm

Blocking improves mixing, plus provably geometrically ergodic.

But what about the cost-per-step?

Cost-per-iteration

Let's focus on the update of β:

$$\beta \mid \sigma^2, \xi, \eta, z \sim \mathsf{N}\left(\left(W'W + (\xi^{-1}D)^{-1}\right)^{-1} W'z,\ \sigma^2 \left(W^T W + (\xi^{-1}D)^{-1}\right)^{-1} \right)$$

where $D = \mathrm{diag}(\eta_j^{-1})$.

Usual Cholesky based sampler (Rue, 2001) for $\mathsf{N}(Q^{-1}b, Q^{-1})$ requires $O(p^3)$ computation for non-sparse Q.

Highly prohibitive $O(p^3)$ complexity per iteration when $p \gg N$.

(Partial) Remedy

In B., Chakraborty, Mallick (2016), we propose an alternative exact sampler with $O(N^2 p)$ complexity.

(i) Sample $u \sim \mathrm{N}(0, \xi^{-1} D)$ and $f \sim \mathrm{N}(0, I_N)$ indep.
(ii) Set $v = Wu + f$
(iii) Solve $M_\xi v^* = (z/\sigma - v)$ where $M_\xi = I_N + \xi^{-1} WDW'$
(iv) Set $\beta = \sigma(u + \xi^{-1} DW' v^*)$

(iii) is the costliest step taking $\max\{O(N^2 p), O(N^3)\}$ steps. Significant savings when $p \gg N$.

Cost-per-iteration

However, still $O(N^2 p)$ computation. N can be in the order of tens of thousands in GWAS studies.

The remaining bottleneck is only in calculating

$$M_\xi = I_N + \xi^{-1} WDW'$$

which is needed by the updates for β, σ^2, and ξ

Our proposal: replace WDW' with a cheaper and accurate approximation

Approximations in MCMC

Approximation

- Horseshoe is designed to shrink most coordinates of β toward zero... So many of the $(\xi\eta_j)^{-1}$ will typically be tiny at any iteration
- Choose a "small" threshold δ, approximate M_ξ by

$$M_{\xi,\delta} = I_N + \xi^{-1} W_S D_S W_S', \quad S = \{j : \xi^{-1}\eta_j^{-1} > \delta\}$$

  where W_S is the sub-matrix consisting of columns in the set S, etc
- Carefully replace all calculations involving M_ξ with $M_{\xi,\delta}$
- Reduces cost per step to $Ns^2 \vee Np$, where $s = |S|$

Note: this is different from setting some $\beta_j = 0$ at each scan. β is still being drawn from a non-singular MVN.

Perturbations in MCMC

- A general strategy to reduce cost-per-step is to replace the exact transition kernel $\mathcal{P}$ with an "approximation" $\mathcal{P}_\epsilon$
- Some other examples - replace a non-standard density with its best approximation from a standard family, divide-conquer...
- $\mathcal{P}_\epsilon$ still a Markov chain
- Question: what can we say about finite-time averages from the approximate chain? In other words, is

$$\mu\varphi \approx n^{-1} \sum_{k=0}^{n-1} \varphi(X_k^\epsilon)$$

for $X_k^\epsilon \sim \nu \mathcal{P}_\epsilon^k$?

Literature review

- Early reference on perturbation bounds: Mitrophanov (2005), for uniformly ergodic chains
- Renewed interest in recent years (Alquier et al. 2014, Pillai & Smith (2015), Rudolf & Schweizer (2018), Johndrow & Mattingley (2018)) - extensions to unbounded state-spaces
- Most applications pertain to "tall data", i.e., lots of samples (Bardenet, Doucet, Holmes (2017))
- Ours is one of the first applications for large N and p with potentially $p \gg N$

A new general perturbation bound

We show that

$$\mathbf{E}\left(\frac{1}{n}\sum_{k=0}^{n-1}\varphi(X_k^\epsilon)-\mu\varphi\right)^2$$

can be "controlled" (skipping exact bounds) if

1. There exists $K_\epsilon > 0$ and $\gamma_\epsilon \in (0,1)$ such that

$$(\mathcal{P}_\epsilon V)(x) \le \gamma_\epsilon V(x) + K_\epsilon,$$

that is V is also Lyapunov for $\mathcal{P}_\epsilon$.

2. The approximate kernel $\mathcal{P}_\epsilon$ satisfies

$$\sup_{x\in\mathbf{X}} \|\mathcal{P}(x,\cdot)-\mathcal{P}_\epsilon(x,\cdot)\|_{TV} \le \frac{\epsilon}{2}.$$

Application to Horseshoe sampler

Recall our approximation step replaces $M_\xi = I_N + \xi^{-1} WDW'$ with $M_{\xi,\delta} = I_N + \xi^{-1} WD_\delta W'$.

We show that this approximation achieves

$$\sup_x \|\mathcal{P}(x,\cdot) - \mathcal{P}_\delta(x,\cdot)\|_{TV}^2 \leq \delta \|W\|^2 \left[4N(\|z\|^2/b_0) + 9\right] + \mathcal{O}(\delta^2)$$

for any small fixed threshold δ.

Satisfies conditions of our general theorem.

Application to Horseshoe sampler

Practically: we recommend $\delta = 10^{-4}$ or 10^{-5} and have observed no advantages from smaller values.

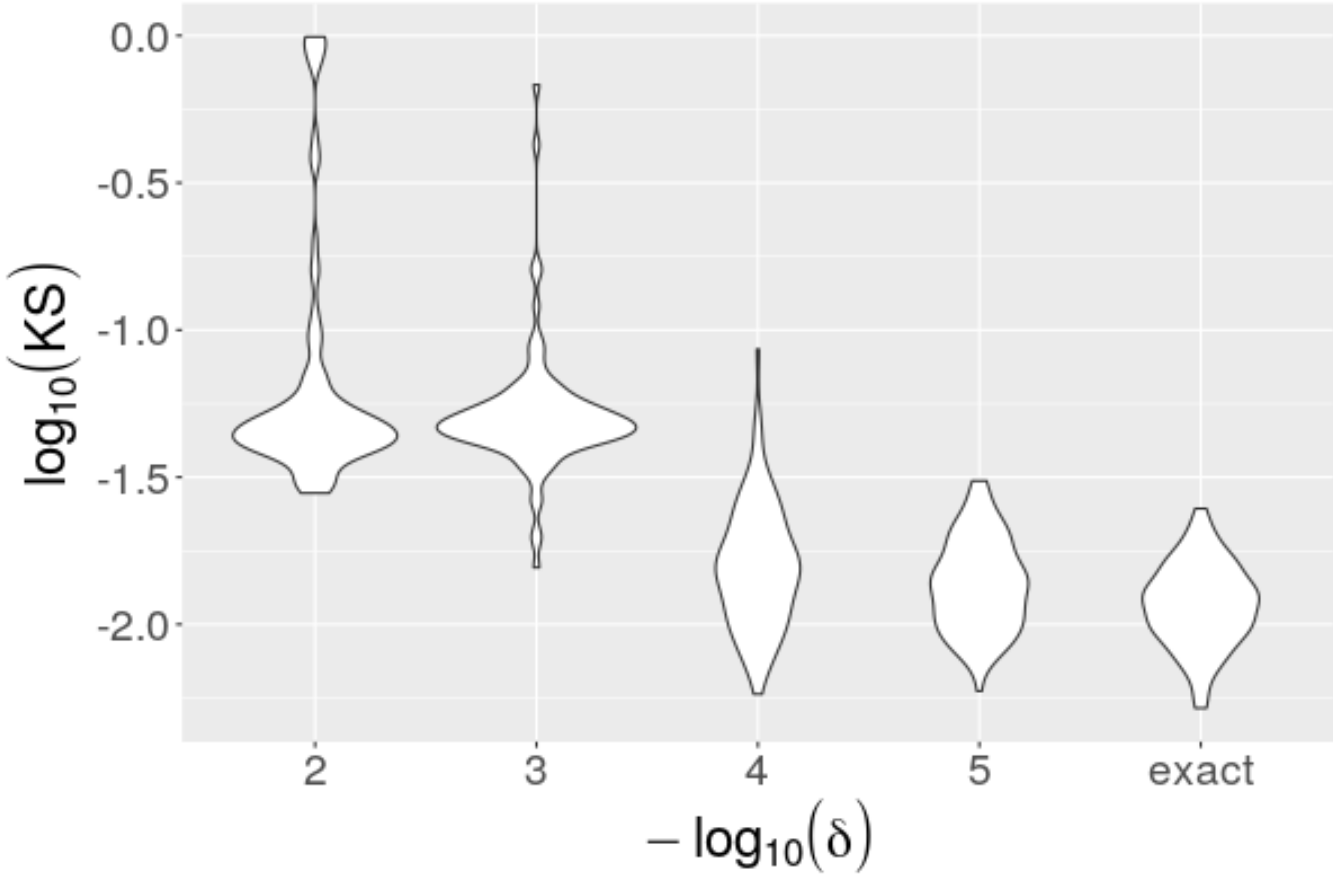


Figure: Average KS distance between the marginals of 100 entries of β from the exact and approximate algorithm for $N = 1000$ and $p = 10000$

Varying threshold

- Using a fixed threshold ϵ results in an asymptotic bias proportional to $\sqrt{\epsilon}/(1-\bar{\alpha})$, where recall $\bar{\alpha}$ quantifies rate of convergence of the exact chain
- More room to use approximations when the exact chain mixes rapidly, i.e., $\sqrt{\epsilon}$ is small compared to the spectral gap $(1-\bar{\alpha})$ of the exact chain
- The asymptotic bias can be eliminated by using a decreasing schedule of approximation parameters (ϵ_k) - need to satisfy $\epsilon_k \to 0$ "sufficiently fast" (summability condition)
- Reminiscent of conditions for stochastic gradient or Langevin dynamics

Simulation studies

The results that follow use a common simulation structure

$$
\begin{aligned}
w_i &\overset{iid}{\sim} \mathrm{N}_p(0, \Sigma) \\
z_i &\sim \mathrm{N}(w_i \beta, 4) \\
\beta_j &= \begin{cases} 2^{-(j/4-9/4)} & j < 24 \\ 0 & j > 23 \end{cases},
\end{aligned}
$$

So there are always "small" and "large" signals, and true nulls
We consider both $\Sigma = I$ (**independent design**) and $\Sigma_{ij} = 0.9^{|i-j|}$ (**correlated design**)

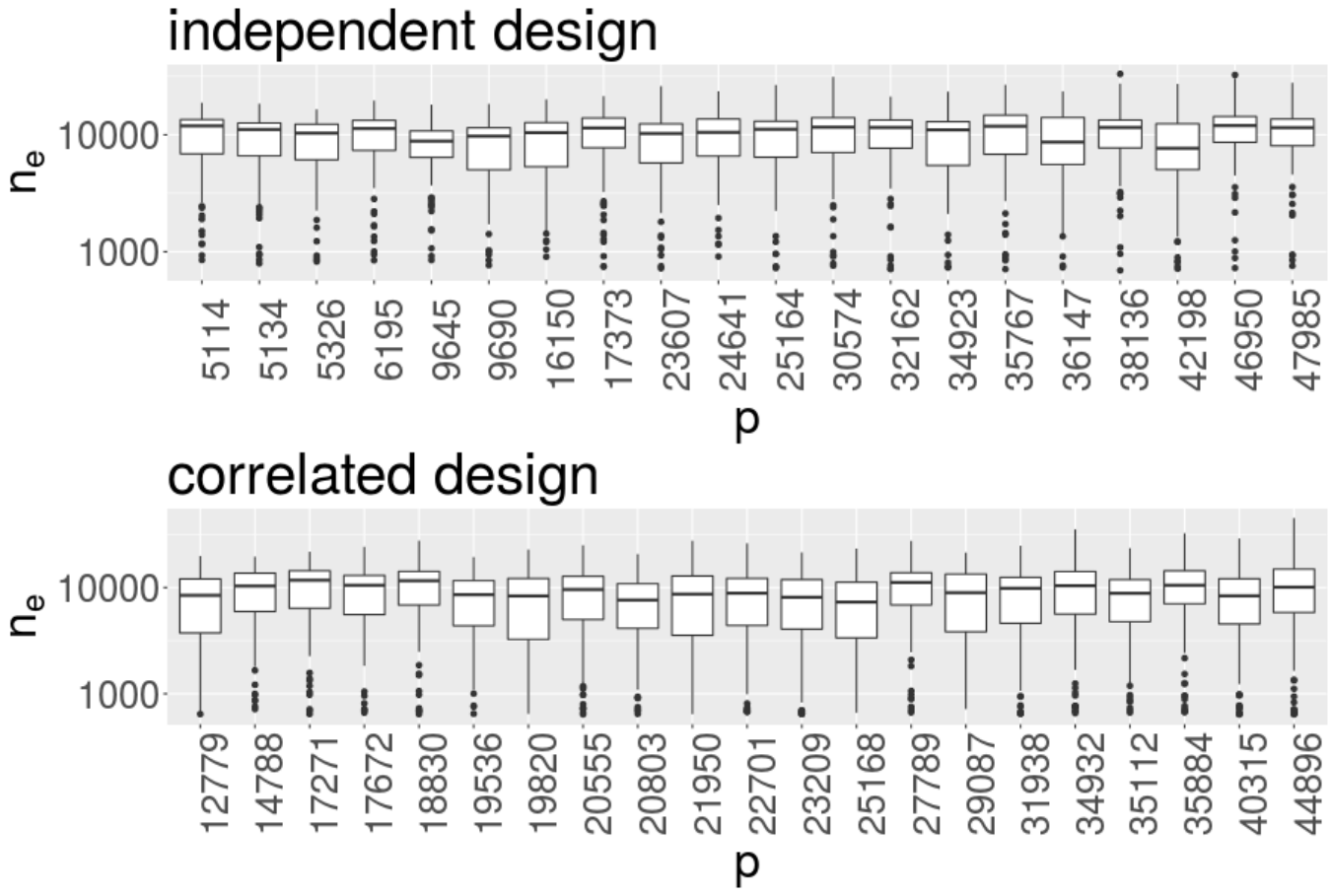
Effective sample size

Recall **effective sample size** n_e, a measure of the number of **independent** samples your Markov path is "worth"

If $n_e = n$ then your MCMC is giving essentially independent samples (like vanilla Monte Carlo)

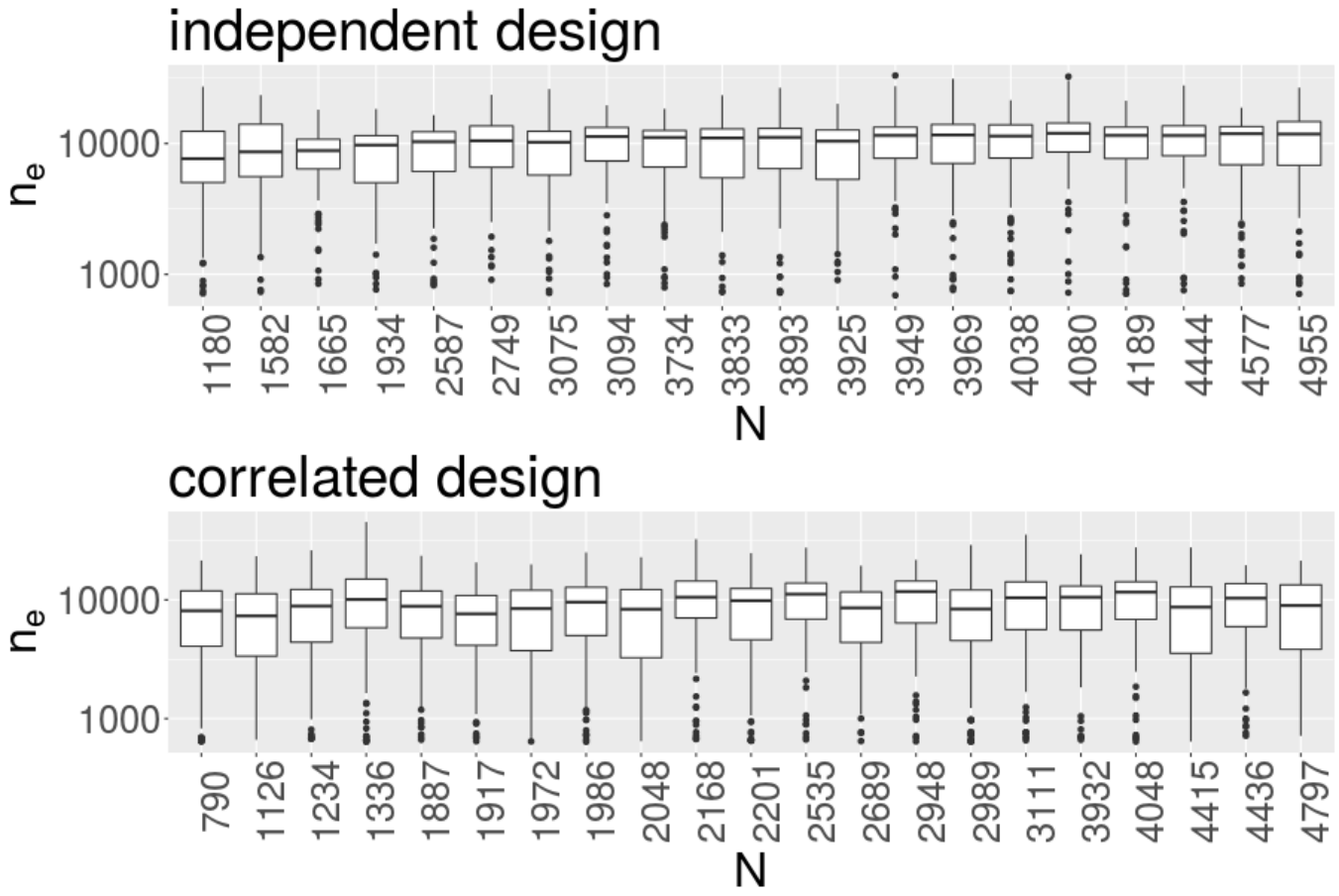
If $n_e \ll n$ then your MCMC has very high autocorrelations, need very long path to get good approximation to posterior

Mixing as p increases



Effective sample sizes are essentially independent of p, even when the design matrix is highly correlated

Mixing as N increases



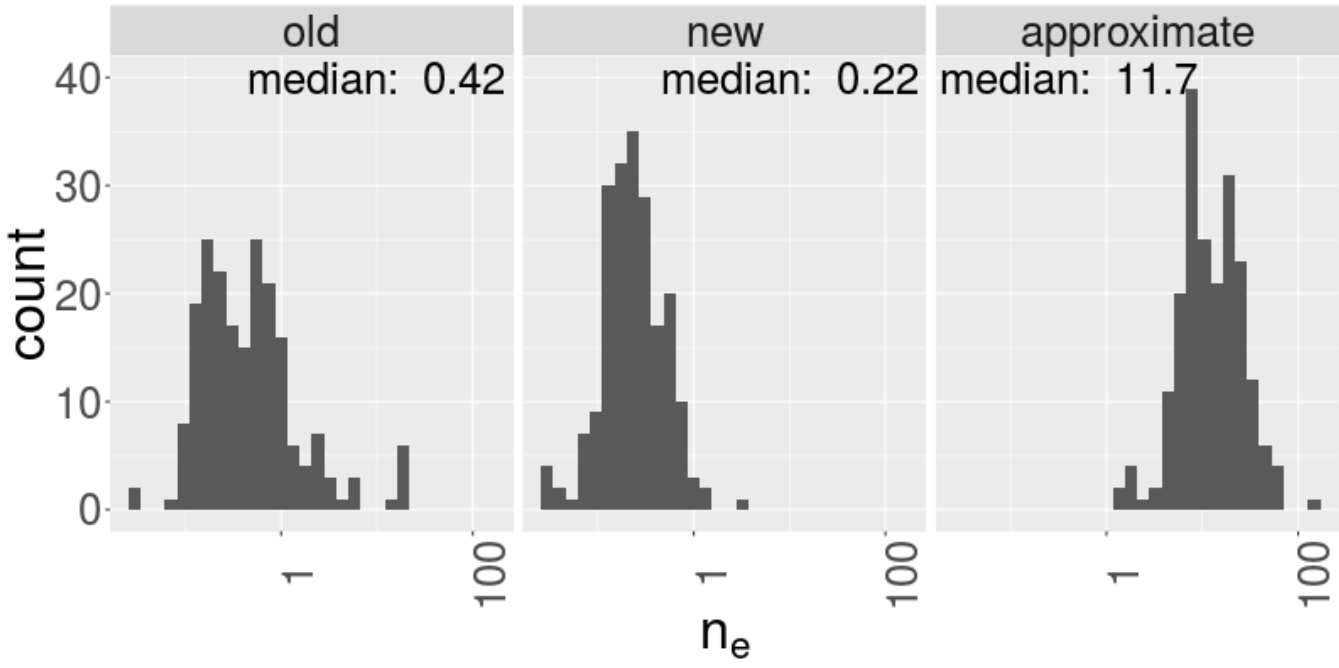
Effective sample sizes are essentially independent of N, even when the design matrix is highly correlated

Effective samples per second

Recall **effective sample size** n_e, a measure of the number of **independent** samples your Markov path is "worth"

So if t is computation time in seconds, **effective samples per second** n_e/t is an empirical measurement of overall computational efficiency

Results: Effective samples per second



The approximate algorithm is **fifty times** more efficient when $N = 2,000$ and $p = 20,000$

Conclusion

Computational cost for MCMC shouldn't massively differ from alternatives designed for the same problem

But making the algorithm fast takes work, often problem-specific

More thrust on "computing" posteriors that we know have "nice" properties

Approximations in MCMC seem a promising direction to speed-up computation

A step towards rigorous quantification of approximation error

References

- Bhattacharya, A., Chakraborty, A., Mallick, B. (2016). Fast sampling with Gaussian scale-mixture priors in high-dimensional regression. *Biometrika* arXiv preprint arXiv:1506.04778
- Johndrow, J. E., Orenstein, P., & Bhattacharya, A. (2018). Bayes Shrinkage at GWAS scale: Convergence and Approximation Theory of a Scalable MCMC Algorithm for the Horseshoe Prior arXiv preprint arXiv:1705.00841.
- Pallavi Ray, Anirban Bhattacharya, and Debdeep Pati. https://arxiv.org/abs/1902.04701

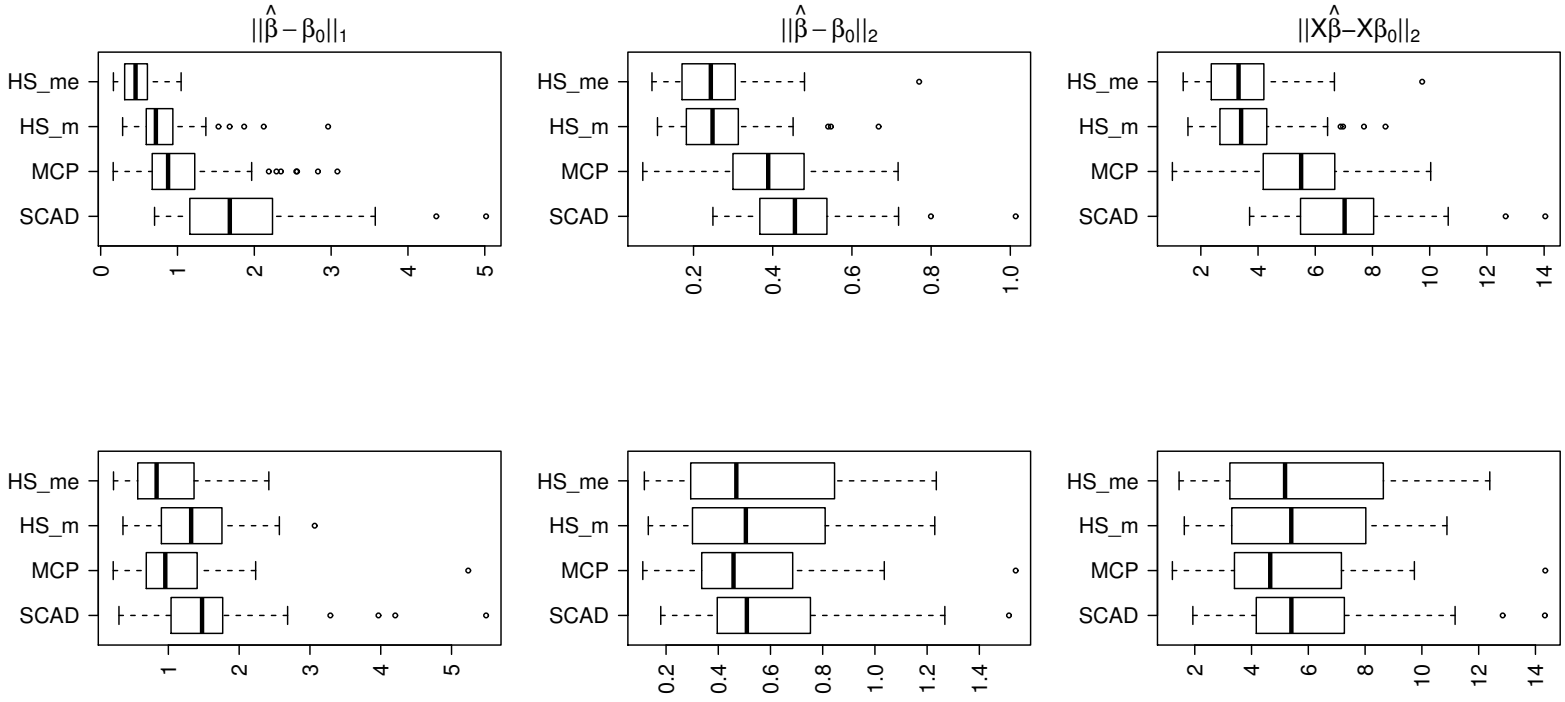
Thank You

Performance in $p \gg n$ settings

Data generation

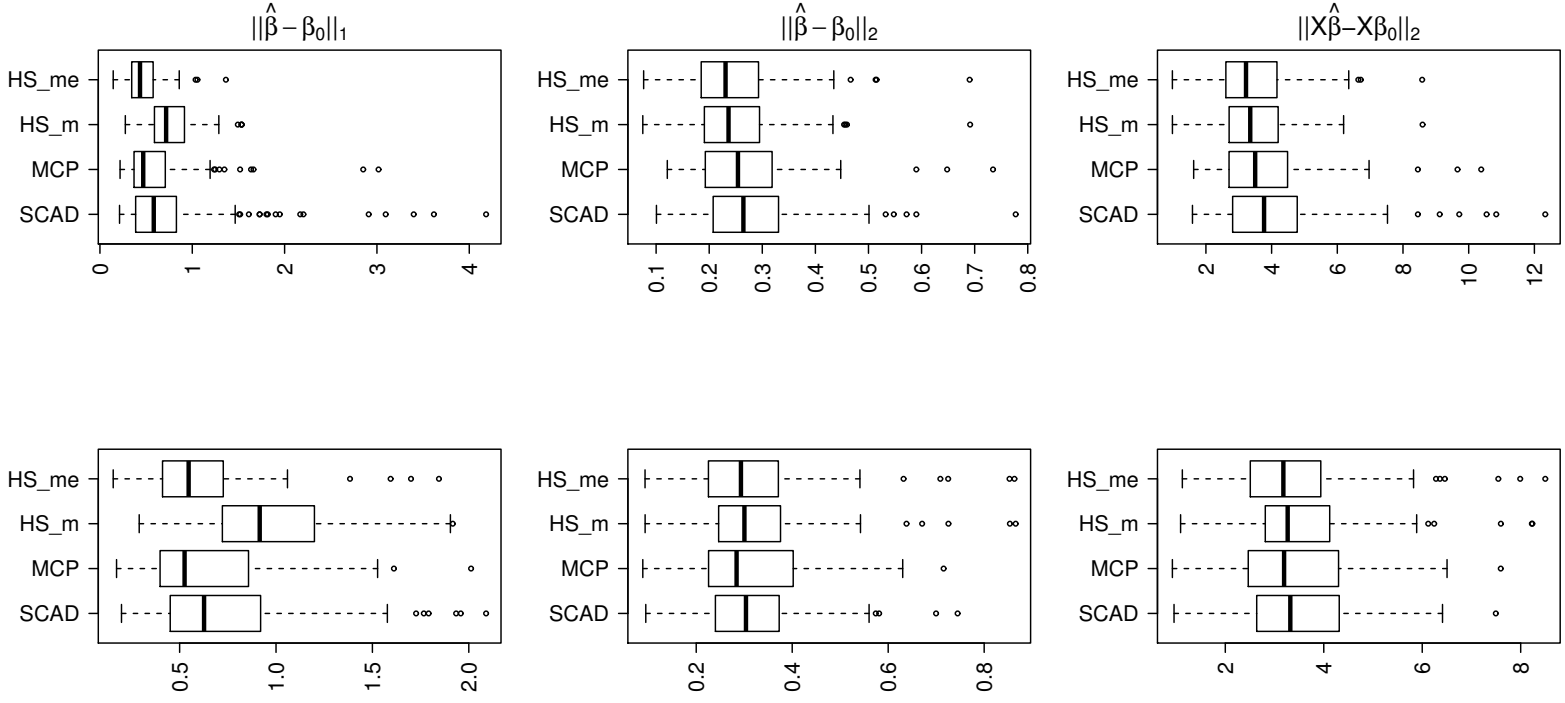
- Replicated simulation study with horseshoe prior (Carvalho et al. (2010)
- $n = 200$ & $p = 5000$. True β_0 has 5 non-zero entries and $\sigma = 1.5$
- Two signal strengths:
  (i) weak - $\beta_{0S} = \pm(0.75, 1, 1.25, 1.5, 1.75)^{\mathrm{T}}$
  (ii) moderate - $\beta_{0S} = \pm(1.5, 1.75, 2, 2.25, 2.5)^{\mathrm{T}}$
- Two types of design matrix:
  (i) Independent - X_j i.i.d. $\mathrm{N}(0, \mathrm{I}_p)$
  (ii) compound symmetry - X_j i.i.d. $\mathrm{N}(0, \Sigma)$, $\Sigma_{jj'} = 0.5 + 0.5\delta_{jj'}$
- Summary over 100 datasets

Weak signal case



Estimation performance: Boxplots of ℓ_1, ℓ_2 and prediction error across 100 simulation replicates. HS_{me} and HS_m are posterior point wise median and mean for the horeshoe prior. Top row: Independent covariates, Bottom row: Compound symmetry

Moderate signal case



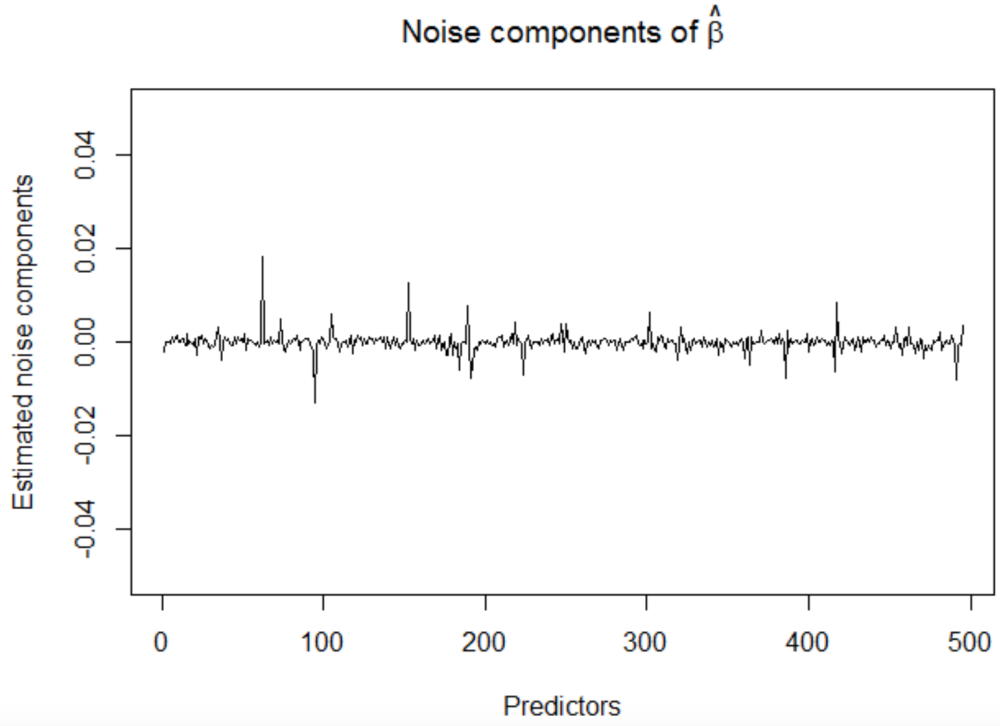
Estimation performance: Boxplots of ℓ_1, ℓ_2 and prediction error across 100 simulation replicates. HS_{me} and HS_m are posterior point wise median and mean for the horeshoe prior. Top row: Independent covariates, Bottom row: Compound symmetry

Frequentist coverage of 95% credible intervals

p	500								
Design	Independent			Comp Symm			Toeplitz		
	HS	LASSO	SS	HS	LASSO	SS	HS	LASSO	SS
Signal Coverage	$93_{1.0}$	$75_{12.0}$	$82_{3.7}$	$95_{0.9}$	$73_{4.0}$	$80_{4.0}$	$94_{4.0}$	$80_{7.0}$	$79_{5.6}$
Signal Length	42	46	41	85	71	75	86	79	74
Noise Coverage	$100_{0.0}$	$99_{0.8}$	$99_{1.0}$	$100_{0.0}$	$98_{1.0}$	$99_{0.8}$	98_{1}	$98_{1.0}$	$99_{0.6}$
Noise Length	2	43	40	4	69	73	5	78	73

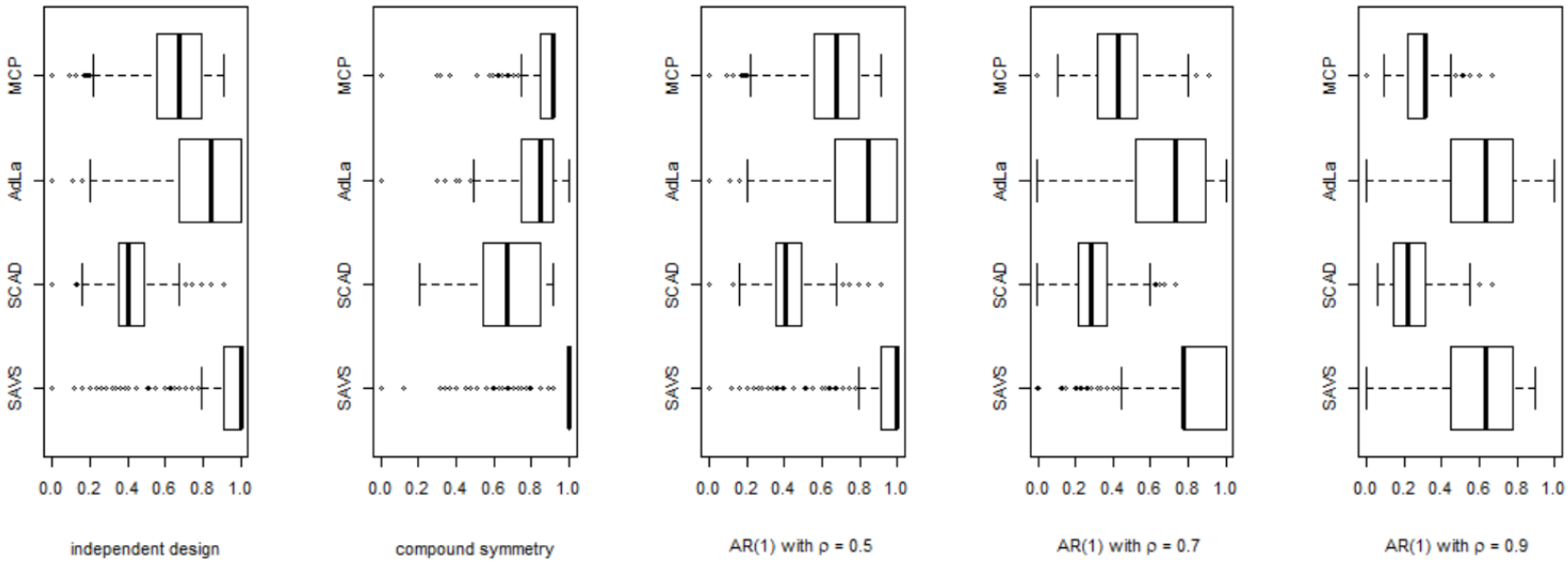
Frequentist coverages (%) and 100×lengths of point wise 95% intervals. Average coverages and lengths are reported after averaging across all signal variables (rows 1 and 2) and noise variables (rows 3 and 4). Subscripts denote 100×standard errors for coverages. LASSO and SS respectively stand for the methods in van de Geer et al. (2014) and Javanmard & Montanari (2014). The intervals for the horseshoe (HS) are the symmetric posterior credible intervals.

Variable selection by postprocessing



$$Q(\beta) = \frac{1}{2}\|X\hat{\beta} - X\beta\|_2^2 + \sum_{j=1}^{p} \mu_j |\beta_j|, \quad \mu_j = |\hat{\beta}_j|^{-2}.$$

Variable selection performance



SAVS: Variable selection by post-processing the posterior mean from the HS prior. Plot of Mathew's correlation coefficient (MCC) over 1000 simulations for various methods. MCC values closer to 1 indicate better variable selection performance.