Scalable Importance Tempering and Bayesian Variable Selection

Giacomo Zanella
joint work with Gareth Roberts

Department of Decision Sciences, BIDSA and IGIER
Bocconi University

Masterclass in Bayesian Statistics, CIRM, Marseille Luminy
22-26 October 2018

Introduction

Bayesian Computation

- Computational scalability is crucial to Bayesian Statistics' applicability
- Here we focus on scalability with the number of parameters p, for example Variable Selection problems with large p

Outline of the talk

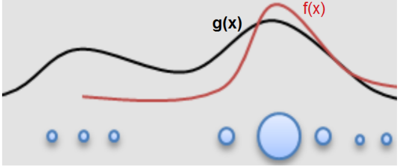
1. Introduction
2. Combining Importance Sampling and MCMC in the context of Gibbs Sampling
3. Analysis of the algorithm
4. Application to Bayesian Variable Selection

Classical approaches to Bayesian computation

Aim: sampling from the posterior distribution $f(\mathbf{x})$

Importance Sampling (IS)

1. Sample $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots \overset{iid}{\sim} g(\mathbf{x})$
2. Weight samples with $w(\mathbf{x}) = \frac{f(\mathbf{x})}{g(\mathbf{x})}$



IS estimators are consistent:

$$\hat{h}_n^{(IS)} = \frac{\sum_{t=1}^{n} w(\mathbf{x}^{(t)}) h(\mathbf{x}^{(t)})}{\sum_{t=1}^{n} w(\mathbf{x}^{(t)})} \xrightarrow{n\to\infty} \mathbb{E}_f[h] = \int h(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} \,.$$

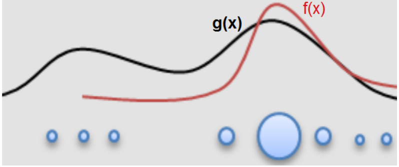
$\bar{h}(\mathbf{x}) = h(\mathbf{x}) - \mathbb{E}_f[h]$

Classical approaches to Bayesian computation

Aim: sampling from the posterior distribution $f(\mathbf{x})$

Importance Sampling (IS)

1. Sample $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots \overset{iid}{\sim} g(\mathbf{x})$
2. Weight samples with $w(\mathbf{x}) = \frac{f(\mathbf{x})}{g(\mathbf{x})}$



IS estimators are consistent:

$$\hat{h}_n^{(IS)} = \frac{\sum_{t=1}^n w(\mathbf{x}^{(t)})h(\mathbf{x}^{(t)})}{\sum_{t=1}^n w(\mathbf{x}^{(t)})} \xrightarrow{n\to\infty} \mathbb{E}_f[h] = \int h(\mathbf{x})f(\mathbf{x})d\mathbf{x}\,.$$

Main weakness

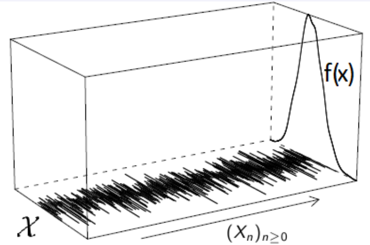
Naive IS is fragile in high dimensions. In particular $\text{var}(h, IS) := \lim_{n\to\infty} n\,\text{var}\left(\hat{h}_n^{(IS)}\right) = \mathbb{E}_f[\bar{h}^2 w]$ can grow as $\exp(d)$ with dimension d.

$\bar{h}(\mathbf{x}) = h(\mathbf{x}) - \mathbb{E}_f[h]$

Markov chain Monte Carlo

Simulate an ergodic Markov chain $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots$ with stationary distribution $f(\mathbf{x})$.Then

$$\frac{1}{n}\sum_{t=1}^{n} h(\mathbf{x}^{(t)}) \overset{n\to\infty}{\longrightarrow} \mathbb{E}_f[h]\,.$$



Main weakness

Exposed to slow mixing. In particular

$$n\,\mathrm{var}\left(\frac{1}{n}\sum_{t=1}^{n} h(\mathbf{x}^{(t)})\right) \overset{n\to\infty}{\longrightarrow} \mathrm{var}_f(h)\left(1+2\sum_{t=1}^{\infty}\rho_t\right)$$

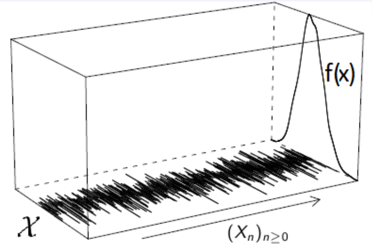
where $\rho_t = \mathrm{Corr}(h(\mathbf{x}^{(s)}), h(\mathbf{x}^{(s+t)}))$ $\rightsquigarrow$ MCMC gets bad if $\sum_{t=1}^{\infty}\rho_t$ is large

Figure from Johansen,Evers,Whiteley(2010)

Markov chain Monte Carlo

Simulate an ergodic Markov chain $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots$ with stationary distribution $f(\mathbf{x})$.Then

$$\frac{1}{n}\sum_{t=1}^{n} h(\mathbf{x}^{(t)}) \overset{n\to\infty}{\longrightarrow} \mathbb{E}_f[h]\,.$$



Main weakness

Exposed to slow mixing. In particular

$$n\,\mathrm{var}\left(\frac{1}{n}\sum_{t=1}^{n} h(\mathbf{x}^{(t)})\right) \overset{n\to\infty}{\longrightarrow} \mathrm{var}_f(h)\left(1+2\sum_{t=1}^{\infty}\rho_t\right)$$

where $\rho_t = \mathrm{Corr}(h(\mathbf{x}^{(s)}), h(\mathbf{x}^{(s+t)}))$ $\rightsquigarrow$ MCMC gets bad if $\sum_{t=1}^{\infty}\rho_t$ is large

"Importance tempering" is a way of combining Importance Sampling and MCMC.

Figure from Johansen,Evers,Whiteley(2010)

Classical Gibbs Sampling

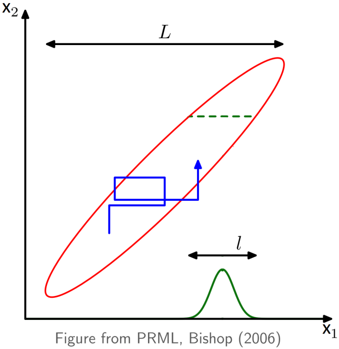


Figure from PRML, Bishop (2006)

$f(\mathbf{x})$ is d-dimensional, $\mathbf{x} = (x_1, \dots, x_d) \in \mathcal{X}^d$

Gibbs Sampling (GS)

At each iteration:

1. Sample i from $\{1, \dots, d\}$ uniformly
2. Update $x_i \sim f(x_i | \mathbf{x}_{-i})$

Main limitation: correlation in the posterior induces slow mixing

Plan: develop an importance tempering version of GS to alleviate slow mixing, and use the one-dimensional nature of GS to have robustness to high-dimensions.

Importance Tempering for the Gibbs Sampler

Classical importance tempering

$\beta \in (0,1]$

$$g(\mathbf{x}) = f^{(\beta)}(\mathbf{x}) = \frac{f(\mathbf{x})^\beta}{\int f(\mathbf{x})^\beta dx}$$

Tempered Gibbs Sampling

Intuition: temper only the coordinate that is being updated.
Consider augmented state space: $(\mathbf{x}, i) \in \mathcal{X}^d \times \{1, \dots, d\}$ and

$$\tilde{f}(\mathbf{x}, i) = \frac{1}{d} f(\mathbf{x}_{-i}) f^{(\beta)}(x_i|\mathbf{x}_{-i})$$

- target $\tilde{f}(\mathbf{x}, i)$ by updating $i \sim \tilde{f}(i|\mathbf{x})$ and $x_i \sim \tilde{f}(x_i|\mathbf{x}_{-i}, i)$.
- Marginal distribution of $\mathbf{x}$ is $\frac{1}{d}\sum_{i=1}^d f(\mathbf{x}_{-i}) f^{(\beta)}(x_i|\mathbf{x}_{-i})$

Tempered Gibbs Sampling

$f^{(\beta)}(x_i|\mathbf{x}_{-i})$ can be replaced with any $g(x_i|\mathbf{x}_{-i})$

Tempered Gibbs Sampling (TGS)

At each iteration:

1. Sample i from $\{1,\dots,d\}$ proportionally to $p_i(\mathbf{x}) = \frac{g(x_i|\mathbf{x}_{-i})}{f(x_i|\mathbf{x}_{-i})}$
2. Update $x_i \sim g(x_i|\mathbf{x}_{-i})$
3. Weight the new state $\mathbf{x}$ with $w(\mathbf{x}) = Z(\mathbf{x})^{-1}$, where $Z(\mathbf{x}) = \frac{1}{d}\sum_{i=1}^{d} p_i(\mathbf{x})$

Induced $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ is invariant w.r.t. $fZ(\mathbf{x}) = \frac{1}{d}\sum_{i=1}^{d} f(\mathbf{x}_{-i})g(x_i|\mathbf{x}_{-i})$. Thus

$$\frac{\sum_{t=1}^{n} w(\mathbf{x}^{(t)})h(\mathbf{x}^{(t)})}{\sum_{t=1}^{n} w(\mathbf{x}^{(t)})} \xrightarrow{n\to\infty} \mathbb{E}_f[h]\,,$$

NB: $g(x_i|\mathbf{x}_{-i}) = f(x_i|\mathbf{x}_{-i})$ corresponds to standard GS

Tempered Gibbs Sampling

Simplest version: $g(x_i|\mathbf{x}_{-i}) = f^{(\beta)}(x_i|\mathbf{x}_{-i})$ for $\beta \in (0,1]$.
At each iteration:

1. Sample i from $\{1,\dots,d\}$ proportionally to $p_i(\mathbf{x}) = \frac{1}{f^{(1-\beta)}(x_i|\mathbf{x}_{-i})}$
2. Update $x_i \sim f^{(\beta)}(x_i|\mathbf{x}_{-i})$
3. Assign to the new state x a weight $w(\mathbf{x}) = Z(\mathbf{x})^{-1}$

Intuition

- Step 1 chooses the "best" coordinate to update at each iteration ("greedy" behavior)
- Step 2 tempers the conditional distribution of the updated variable to make longer moves and overcome correlation
- Modifications in Steps 1&2 compensate each other and keep $Var(w(\mathbf{x}))$ low.

TGS and correlation

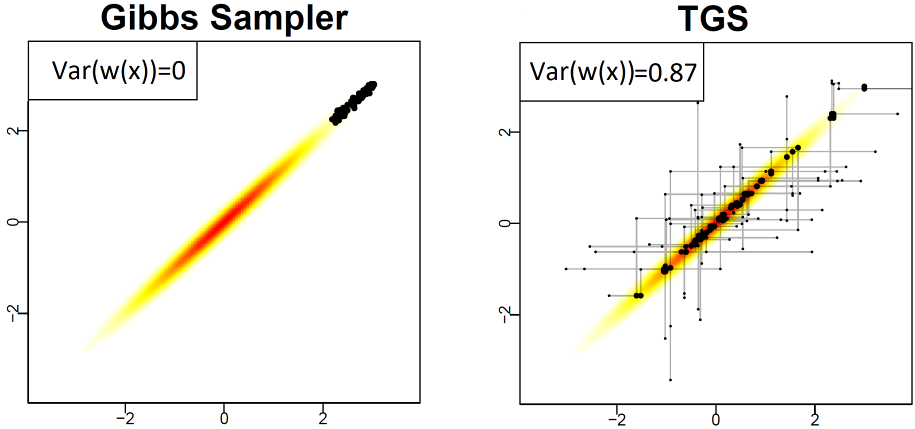


Figure: GS&TGS on a correlated Gaussian. Dots are proportional to importance weights.

⇝ Improving mixing by allowing some variance of the importance weights

TGS can mix faster because the importance distribution has weaker correlation than the original one.

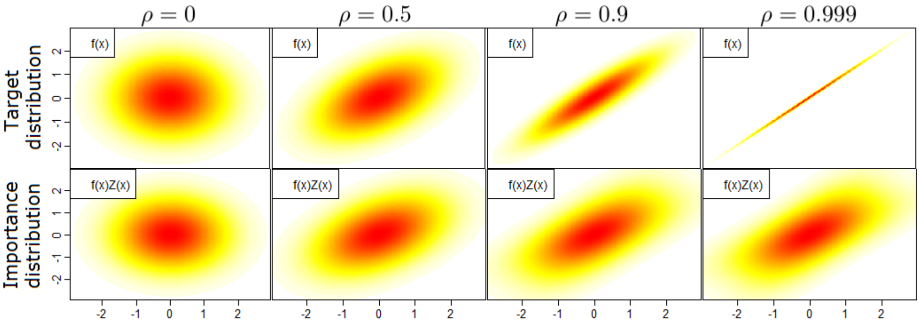
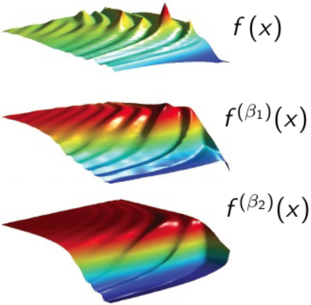


Figure: Target $f(\mathbf{x})$ and importance distribution $f(\mathbf{x})Z(\mathbf{x})$, for increasing correlation.

NB: standard tempering would not reduce correlation here!

Remark: difference from classical tempering

- Most MCMC schemes try to sample *exactly* from f
- "Importance Tempering": run Markov chain on g and reweight samples with $w(\mathbf{x})$. However, plain importance tempering rarely used!
- More common tempering schemes (simulated tempering, parallel tempering, SMC samplers,...) build a sequence $f^{(\beta_0)}(\mathbf{x}), \dots, f^{(\beta_k)}(\mathbf{x})$ and keep samples from $f^{(\beta_0)} = f$.
- Very different in spirit from TGS



Theoretical guarantees?

Measure of efficiency: asymptotic variances

$$var(h, TGS) := \lim_{n\to\infty} n\,\mathrm{var}\left(\frac{\sum_{t=1}^{n} w(\mathbf{x}^{(t)})h(\mathbf{x}^{(t)})}{\sum_{t=1}^{n} w(\mathbf{x}^{(t)})}\right)$$

where $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ is the Markov chain generated by TGS.

Importance sampling & MCMC contribution

We have

$$\mathrm{var}(h, TGS) = \mathrm{var}(h, IS)\left(1 + 2\sum_{t=1}^{\infty} \rho_t\right)$$

$\mathrm{var}(h, IS)$ is the asymptotic variance of importance sampling with proposal fZ
ρ_t is the lag t autocorrelation of $(w(\mathbf{x}^{(i)})h(\mathbf{x}^{(i)}))_{i=1}^{\infty}$

Theoretical guarantees (IS)

- Concern with classical IS is that var(h, IS) could grow as exp(d)
- In TGS we are tempering one coordinate at a time
  $\rightsquigarrow$ we don't pay a dimensionality price in var(h, IS).

Robustness to high-dimensionality

Given the importance distribution $fZ(\mathbf{x}) = \frac{1}{d}\sum_{i=1}^{d} f(\mathbf{x}_{-i})g(x_i|\mathbf{x}_{-i})$

1.
$$Var\,(h, IS) \leq c\,,$$
   where c is a constant independent of d. In applications $c = 2$.
2. For "nice" targets $Var\,(w(\mathbf{x})) \to 0$ as $d \to \infty$.
   Intuition: $w(\mathbf{x}) = (\frac{1}{d}\sum_{i=1}^{d} p_i(\mathbf{x}))^{-1}$ is an average and stabilizes for large d.

$\rightsquigarrow$ IS variance does not harm here.

Theoretical Guarantees (MCMC)

Mixing of the Markov chain

1. The mixing of TGS can never be significantly worse than the one of GS

$$\mathrm{var}(h, TGS) \leq c^2 \mathrm{var}(h, GS) + c^2 \mathrm{var}_f(h)$$

In applications $c^2 = 4$. *(Proof involves continuous-time formulation of the chains, Peskun ordering and control on the importance weights.)*

$\rightsquigarrow$ The mixing is never worse, but when is it better?

2. For simple bivariate cases one can show that the mixing time of TGS is uniformly bounded over the correlation $\rho \in (0, 1)$.
(Proof involves notion of "deinitializing" Markov chain.)

When does TGS help? (and when it doesn't?)

Whether or not TGS overcomes correlation depends on the geometry of the target:

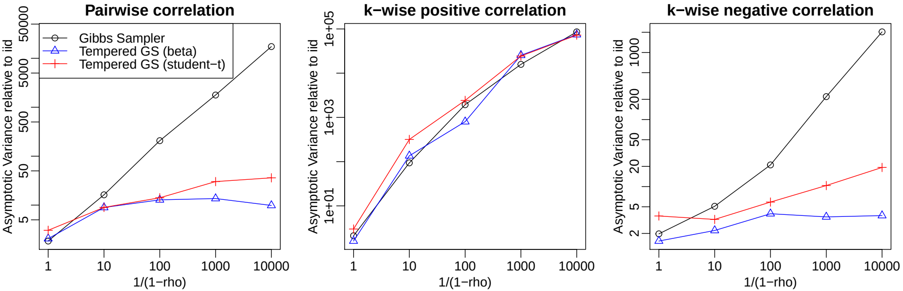


Figure: Log-log plots of var(h, GS) and var(h, TGS) for Gaussian targets with difference covariance structures.

TGS effective for targets with pairwise and high-order negative correlations, but not for high-order positive correlations $\rightsquigarrow$ indication of which models to use it for!

Application to Bayesian Variable Selection

Classical linear regression: given an $n \times p$ design matrix X

$$\begin{aligned} Y|\beta,\sigma^2 \sim & N(X\beta,\sigma^2\mathbb{I}_n) \\ \beta|\sigma^2 \sim N(0,\sigma^2\Sigma)\,, & \qquad p(\sigma^2) \propto \frac{1}{\sigma^2}\,. \end{aligned}$$

Bayesian Variable Selection (BVS)

Introduce binary indicators: $\gamma_i = 1$ if the i-th regressor is "active" and $\gamma_i = 0$ otherwise. Place prior distribution on $\gamma = (\gamma_1, \ldots, \gamma_p) \in \{0,1\}^p$.

$$\begin{aligned} Y|\beta_\gamma,\gamma,\sigma^2 \sim & N(X_\gamma\beta_\gamma,\sigma^2\mathbb{I}_n) \\ \beta_\gamma|\gamma,\sigma^2 \sim N(0,\sigma^2\Sigma_\gamma)\,, & \qquad p(\sigma^2) \propto \frac{1}{\sigma^2}\,. \end{aligned}$$

X_γ is the $n \times |\gamma|$ matrix containing only the columns of the active regressors
β_γ is the $|\gamma| \times 1$ vector containing only the coefficients of the active regressors
Σ_γ is a $|\gamma| \times |\gamma|$ prior covariance matrix. Here $|\gamma| = \sum_{i=1}^{p} \gamma_i$

Bayesian Variable Selection

- Joint posterior distribution $p(\gamma, \beta, \sigma | Y)$. Posterior inclusion probability of i-th variable given by $p(\gamma_i = 1 | Y)$
- BVS has many attractive properties (UQ, interpretability, consistency, good predictions,...) but the bottleneck is posterior computation
- Cost driven by p, not n. Many applications involve $p \gg n$
- After integrating out β and σ analytically you're left with $p(\gamma | Y)$, with $\gamma \in \{0,1\}^p$. Computation done by Gibbs Sampling on $(\gamma_1, \ldots, \gamma_p)|Y$.
- $(\gamma_1, \ldots, \gamma_p)|Y$ is high-dimensional target with only pairwise and negative correlation $\rightsquigarrow$ theory suggests TGS should mix well here!

TGS for Bayesian Variable Selection

Parameter space: $\gamma \in \{0,1\}^p$
Target: $f(\gamma) = p(\gamma|Y)$
Tempered conditionals: $g(\gamma_i|\gamma_{-i}) = \text{Unif}(\{0,1\})$

TGS for Variable Selection

At each iteration

1. Sample i from $\{1,\ldots,p\}$ proportionally to $p_i(\gamma) = \frac{1}{p(\gamma_i|\gamma_{-i},Y)}$
2. Flip γ_i to $1-\gamma_i$
3. Assign to the new state γ a weight $w(\gamma) = Z(\gamma)^{-1}$

Illustrative example

Simulated data with $p = 100$ and variables 1 and 2 strongly correlated.
GS gets stuck in the local modes $(\gamma_1, \gamma_2) = (1, 0)$ and $(\gamma_1, \gamma_2) = (0, 1)$.

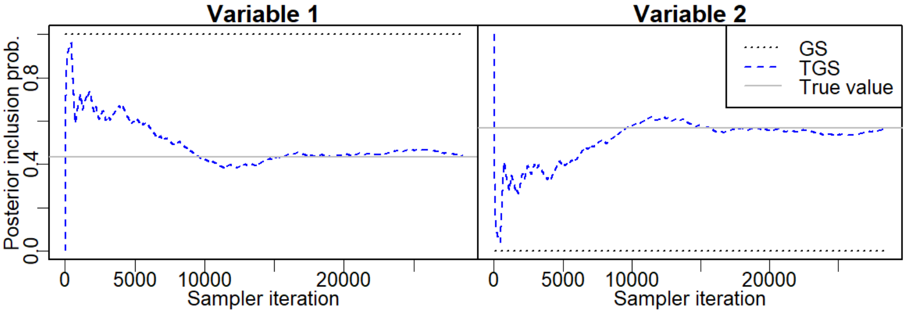


Figure: Running estimates of posterior inclusion probabilities for variables 1 and 2 produced by GS and TGS. Horizontal line is the truth.

Speed-up trick: weighted TGS (wTGS)

Multiply $p_i(\mathbf{x})$ with weight function $\eta_i(\mathbf{x}_{-i})$ without affecting algorithms' validity

1. Sample i from $\{1, \dots, p\}$ proportionally to $p_i(\mathbf{x}) = \eta_i(\mathbf{x}_{-i}) \frac{g(x_i|\mathbf{x_{-i}})}{f(x_i|\mathbf{x_{-i}})}$,
2. Sample $x_i \sim g(x_i|\mathbf{x}_{-i})$,
3. Weight the new state $\mathbf{x}$ with a weight $Z(\mathbf{x})^{-1}$

Now the i-th coordinate gets updated with frequency $\mathbb{E}[\eta_i(\mathbf{x}_{-i})] \neq 1/p$

wTGS for Variable Selection

In BVS, set $\eta_i(\gamma_{-i}) = p(\gamma_i = 1|\gamma_{-i}, Y)$ so that $\mathbb{E}[\eta_i(\gamma_{-i})] \propto p(\gamma_i = 1|Y)$
$\rightsquigarrow$ "focus" computational effort on more important variables.

At each iteration

1. Sample i from $\{1, \dots, p\}$ proportionally to $p_i(\gamma) = \frac{p(\gamma_i=1|\gamma_{-i}, Y)}{p(\gamma_i|\gamma_{-i}, Y)}$
2. Flip γ_i to $1 - \gamma_i$
3. Assign to the new state γ a weight $w(\gamma) = Z(\gamma)^{-1}$

Illustrative example

Simulated data with $p = 1000$ and variables 1 and 2 strongly correlated. GS gets stuck in the local modes $(\gamma_1, \gamma_2) = (1, 0)$ and $(\gamma_1, \gamma_2) = (0, 1)$.

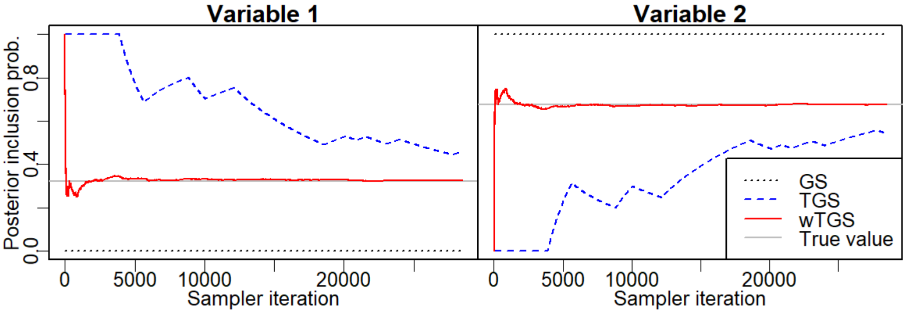


Figure: Running estimates of posterior inclusion probabilities for variables 1 and 2 produced by GS, TGS and wTGS. Horizontal line is the truth.

Computational complexity?

$$\text{Complexity} = (\text{Cost per iteration}) \times (\,\#\text{ iterations})$$

Cost per iteration

- TGS has a higher cost per iteration in computing $\{p_i(\gamma)\}_{i=1}^p$
- For BVS $\{p_i(\gamma)\}_{i=1}^p$ can be computed with single matrix multiplication

  ⇝ GS cost per iteration[1] $\mathcal{O}(|\gamma|^2)$, where $|\gamma| = \sum_{i=1}^p \gamma_i$

  ⇝ TGS cost per iteration[2] $\mathcal{O}(|\gamma| p)$
- Values of $\{p_i(\gamma)\}_{i=1}^p$ can be recycled to compute Rao-Blackwellized estimators.

[1] computing Cholesky decomposition of $|\gamma| \times |\gamma|$ matrix

[2] doing a $|\gamma| \times |\gamma|$ times $|\gamma| \times p$ matrix product

Number of iterations?

# iterations depends on the mixing properties of the Markov chain. We will study the relaxation time. For example, for GS:

$$t_{GS} = Gap(P_{GS})^{-1} \quad \Rightarrow \quad \frac{\mathrm{var}(h, GS)}{\mathrm{var}_f(h)} \leq 2\, t_{GS}$$

Interpretation: one "effective sample" every $2\, t_{GS}$ iterations.

How do t_{GS}, t_{TGS} and t_{wTGS} scale with p?

Computational complexity of GS, TGS and wTGS

Consider two extreme scenarios

1. Uncorrelated variables ($X^T X$ diagonal)

$$t_{GS} = \mathcal{O}(p)\,, \qquad t_{TGS} = \mathcal{O}(p)\,, \qquad t_{wTGS} = \mathcal{O}(s)$$

where s is the average number of *active* variables. Thus

$$\text{Compl}_{GS} = \mathcal{O}(p\,s^2)\,, \qquad \text{Compl}_{TGS} = \mathcal{O}(p^2\,s)\,, \qquad \text{Compl}_{wTGS} = \mathcal{O}(p\,s^2)$$

2. Maximally correlated variables (m collinear, $p - m$ noise)

$$t_{GS} \geq \mathcal{O}(c^{1/2}h^{-1}p) \approx \mathcal{O}(p^3)\,, \qquad t_{TGS} = \mathcal{O}(p)\,, \qquad t_{wTGS} = \mathcal{O}(s)\,.$$

Thus

$$\text{Compl}_{GS} = \mathcal{O}(p^3\,s^2)\,, \qquad \text{Compl}_{TGS} = \mathcal{O}(p^2\,s)\,, \qquad \text{Compl}_{wTGS} = \mathcal{O}(p\,s^2)$$

Simulation study

3 simulated scenarios (varying strength and types of correlation)
Various levels of n, p and signal-to-noise.

		TGS-vs-GS				wTGS-vs-GS			
		SNR				SNR			
	(p,n)	0.5	1	2	3	0.5	1	2	3
scen.1	(100,50)		7.2e1	1.8e1	2.8e2		5.8e2	4.2e2	3.1e3
	(200,200)	4.9e3		6.6e1	1.9e2	1.1e4		1.8e3	1.6e4
	(1000,500)	2.7e2	6.3e2	1.4	8.1e1	8.8e3	2.5e4	5.8e2	1.9e4
scen.2	(100,50)	4.8	1.4e1	3.3	2.0e1	1.3e2	2.4e2	1.8e1	1.4e2
	(200,200)	8.6e1	4.7e1	3.4	2.5e6	2.3e3	2.1e3	6.0e1	4.1e2
	(1000,500)	4.6e1	3.7e1	1.3e1	4.5e2	1.1e4	7.6e3	1.1e3	1.8e4
scen.3	(100,50)	2.7	5.3	9.2		2.5e1	6.7e1	2.1e1	
	(200,200)	1.1e2	6.6e1			1.3e3	4.6e2		
	(1000,500)	1.6e1	6.8e2			1.1e3	9.4e3		

Table: Mean efficiency improvement of TGS and wTGS over GS. Empty values corresponds to large values with no reliable estimate available.

Large p genomic dataset[5]

$p = 10172$. Compare TGS with GS and Hamming Ball Sampler[3] (HBS)

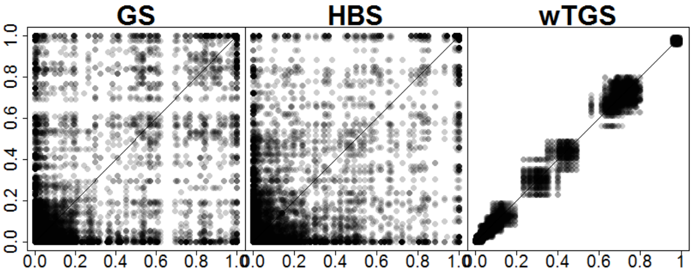


Figure: Points close to the diagonal line indicate estimates agreeing across different runs.

- Runtime less than 2 minutes with pure R on single desktop computer[4]
- $p \approx 10^4$ often considered computationally infeasible for Bayesian approach to Variable Selection (most available *R* packages require hours to fit this model).

[3] Titsias and Yau (2017) The Hamming Ball Sampler. JASA

[4] R code available at `https://github.com/gzanella/TGS`

[5] Human microarray gene expression data in colon cancer patients from Calon et al. (2012)

Discussion

- Proposed a combination of IS&MCMC that is robust to high-dimensionality.
- Theoretical results, e.g. guarantees of improving convergence over GS, but with higher cost per iteration.
- TGS will work well if:
  (a) posterior exhibits negative and/or pairwise correlation;
  (b) computing the selection probabilities $\{p_i(\gamma)\}_{i=1}^{p}$ can be done efficiently.
- Simple and scalable sampler for spike and slab Bayesian Variable Selection. Computational complexity results in simple scenarios.
- Many extensions and variations of the algorithmic scheme possible.

Arxiv preprint: G.Zanella&G.O.Roberts (2018) Scalable Importance Tempering and Bayesian Variable Selection.

Acknowledgements: support by the European Research Council (ERC) through starting grant "N-BNP", PI. Igor Prünster.