A LEVEL-SET HIT-AND-RUN SAMPLER FOR QUASI-CONCAVE DISTRIBUTIONS

Supplementary Materials

1 The LSHR1 Algorithm

In this appendix, we give pseudo-code for our LSHR1 level-set hit-and-run sampler described in our manuscript. We must pre-specify a minimum value K of the probability density $f(\cdot)$ as our stopping threshold. The first threshold t_1 must be pre-specified, and we arbitrarily use $t_1 = 0.95 * f(\boldsymbol{x}_{\max})$. Finally, the number of hit-and-run iterations per level set, m, must be chosen. The value of m influences the accuracy of our volume ratio estimates. In Section 4, we explore appropriate choices for the number of hit-and-run iterations m. We find that a value of $m = 1000$ is a reasonable trade-off between accurate estimation of our volume ratios and the computational cost of more hit-and-run steps.

When considering the proposal for level set $k+1$, the initial t_{prop} is set to $t_k - (t_{k-1} - t_k)$, so that our proposal moves the same distance in t as the previous successful move. If this initial t_{prop} is rejected for not being warm enough, we choose a new proposal t_{prop} as the midpoint of the old proposal and t_k, and so on.

Algorithm 1 LSHR1: Level-set Hit-and-run Sampler

initialize $\boldsymbol{x} = \boldsymbol{x}_{\max}$ and $\Sigma_1 = \mathrm{I}$
initialize t_1 and set $k = 1$
define current level set C_1 as $\{\boldsymbol{x} : f(\boldsymbol{x}) > t_1\}$
for m iterations **do**
    sample random direction $\boldsymbol{d}$
    calculate boundaries $\boldsymbol{a}$ and $\boldsymbol{b}$ of C_1 along direction $\boldsymbol{d}$
    sample new $\boldsymbol{x}$ uniformly between $\boldsymbol{a}$ and $\boldsymbol{b}$
    store all samples $\boldsymbol{x}$ in $\{\boldsymbol{x}\}_1$
while $t_k > K$ **do**
    propose new $t_{\mathrm{prop}} < t_k$
    define proposed level set C_{prop} as $\{\boldsymbol{x} : f(\boldsymbol{x}) > t_{\mathrm{prop}}\}$
    for m iterations **do**
        sample random direction $\boldsymbol{d}^\star$ and set $\boldsymbol{d} = \Sigma_k^{1/2}\boldsymbol{d}^\star$
        calculate boundaries $\boldsymbol{a}$ and $\boldsymbol{b}$ of C_1 along direction $\boldsymbol{d}$
        sample new $\boldsymbol{x}$ uniformly between $\boldsymbol{a}$ and $\boldsymbol{b}$
        if $f(\boldsymbol{x}) > t_k$ **then** $\hat{R}_{k:\mathrm{prop}} = \hat{R}_{k:\mathrm{prop}} + 1$
    $\hat{R}_{k:\mathrm{prop}} = \hat{R}_{k:\mathrm{prop}}/m$
    if $0.55 \leq \hat{R}_{1:\mathrm{prop}} \leq 0.8$ **then**
        $k = k + 1$
        $t_k = t_{\mathrm{prop}}$; $C_{\mathrm{k}} = C_{\mathrm{prop}}$; $\hat{R}_{k:k+1} = \hat{R}_{k:\mathrm{prop}}$; $\Sigma_{k+1} = \mathrm{Cov}(\boldsymbol{x})$
        store all samples $\boldsymbol{x}$ in $\{\boldsymbol{x}\}_k$
define n = number of level sets
for $i = 1, \ldots, n$ **do**
    $q_i = (t_{i-1} - t_i) \times \prod_{k=i}^{n} \hat{R}_{k:k+1}$
    where t_0 is the maximum of $f(\cdot)$ and $\hat{R}_{n:n+1} = 1$
for $l = 1, \ldots, L$ **do**
    Sample level set s w.p. $p_s = q_s / \sum q_s$
    Sample point $\boldsymbol{x}_l$ randomly from level set collection $\{\boldsymbol{x}\}_s$

2 The LSHR2 Algorithm

In this appendix, we give pseudo-code for LSHR2, our exponentially-weighted level-set hit-and-run sampler procedure described in our manuscript. We must pre-specify a minimum value K of the probability density $f(\cdot)$ as our stopping threshold. The first threshold t_1 must be pre-specified, and we arbitrarily use $t_1 = 0.95 * f(\boldsymbol{\theta}_{\max})$ where $\boldsymbol{\theta}_{\max}$ is the mode of $f(\boldsymbol{\theta})$. We set the number of hit-and-run iterations per level set to $m = 1000$.

When considering the proposal for level set $k+1$, the initial t_{prop} is set to $t_k - (t_{k-1} - t_k)$, so that our proposal moves the same distance in t as the previous successful move. If this initial t_{prop} is rejected for not being warm enough, we choose a new proposal t_{prop} as the midpoint of the old proposal and t_k, and so on.

Algorithm 2 LSHR2: Exponentially-tilted Level-set Hit-and-run Sampler

initialize $\boldsymbol{\theta} = \boldsymbol{\theta}_{\max}$ and $\Sigma_1 = \mathrm{I}$
initialize $p = \log g(\boldsymbol{\theta}_{\max})$ and $\boldsymbol{\theta}^\star = (\boldsymbol{\theta}, p)$
initialize t_1 and set $k = 1$
define current level set D_1 as $\{\boldsymbol{\theta}^\star : f(\boldsymbol{\theta}) \geq t_1 \, , \, p < \log g(\boldsymbol{y}|\boldsymbol{\theta})\}$
for m iterations **do**
    sample random direction $\boldsymbol{d}$ in $(d+1)$ dimensions
    calculate boundaries $\boldsymbol{a}$ and $\boldsymbol{b}$ of D_1 along direction $\boldsymbol{d}$
    sample new $\boldsymbol{\theta}^\star$ proportional to $\exp(p)$ from line segment between $\boldsymbol{a}$ and $\boldsymbol{b}$
    store all samples $\boldsymbol{\theta}^\star$ in $\{\boldsymbol{\theta}^\star\}_1$
while $t_k > K$ **do**
    propose new $t_{\text{prop}} < t_k$
    define proposed level set D_{prop} as $\{\boldsymbol{\theta}^\star : f(\boldsymbol{\theta}) \geq t_{\text{prop}} \, , \, p < \log g(\boldsymbol{y}|\boldsymbol{\theta})\}$
    for m iterations **do**
        sample random direction $\boldsymbol{d}^\star$ in $(d+1)$ dimensions and set $\boldsymbol{d} = \Sigma_k^{1/2}\boldsymbol{d}^\star$
        calculate boundaries $\boldsymbol{a}$ and $\boldsymbol{b}$ of D_{prop} along direction $\boldsymbol{d}$
        sample new $\boldsymbol{\theta}^\star$ proportional to $\exp(p)$ from line segment between $\boldsymbol{a}$ and $\boldsymbol{b}$
        if new $\boldsymbol{\theta}^\star \in D_k$ **then** $\hat{R}_{k:\text{prop}} = \hat{R}_{k:\text{prop}} + 1$
    $\hat{R}_{k:\text{prop}} = \hat{R}_{k:\text{prop}}/m$
    if $0.55 \leq \hat{R}_{1:\text{prop}} \leq 0.8$ **then**
        $k = k+1$
        $t_k = t_{\text{prop}}$; $D_{\text{k}} = D_{\text{prop}}$; $\hat{R}_{k:k+1} = \hat{R}_{k:\text{prop}}$; $\Sigma_{k+1} = \mathrm{Cov}(\boldsymbol{x})$
        store all samples $\boldsymbol{\theta}^\star$ in $\{\boldsymbol{\theta}^\star\}_k$
define n = number of level sets
for $i = 1, \ldots, n$ **do**
    $q_i = (t_{i-1} - t_i) \times \prod_{k=i}^{n} \hat{R}_{k:k+1}$
    where t_0 is the maximum of $f(\cdot)$ and $\hat{R}_{n:n+1} = 1$
for $l = 1, \ldots, L$ **do**
    Sample level set s w.p. $p_s = q_s / \sum q_s$
    Sample point $\boldsymbol{\theta}_l^\star$ randomly from level set collection $\{\boldsymbol{\theta}^\star\}_s$

3 Gibbs Switching Calculation

For a variable in the zero component, with norm $||\mathbf{x}||_2^2 \approx d\sigma_0^2$, the probability of a switch is

$$
\begin{aligned}
P\left(I = 1 \,|\, ||\mathbf{x}||_2^2 \approx d\sigma_0^2\right) &= \frac{\sigma_1^{-d}\exp(-||\mathbf{x}||_2^2/2\sigma_1^2)}{\sigma_1^{-d}\exp(-||\mathbf{x}||_2^2/2\sigma_1^2) + \sigma_0^{-d}\exp(-||\mathbf{x}||_2^2/2\sigma_0^2)} \\
&= \frac{\sigma_1^{-d}\exp(-d\sigma_0^2/2\sigma_1^2)}{\sigma_1^{-d}\exp(-d\sigma_0^2/2\sigma_1^2) + \sigma_0^{-d}\exp(-d/2)} \\
&= \frac{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\left(1 - \frac{\sigma_0^2}{\sigma_1^2}\right)\right)}{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\left(1 - \frac{\sigma_0^2}{\sigma_1^2}\right)\right) + 1} \\
&\approx \frac{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\right)}{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\right) + 1}
\end{aligned}
$$

since $\sigma_0 \ll \sigma_1$. Now, as d increases, $(\sigma_0/\sigma_1)^d$ goes to zero, and so

$$\frac{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\right)}{\left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\right) + 1} \approx \left(\frac{\sigma_0}{\sigma_1}\right)^d \exp\left(\frac{d}{2}\right) = \left(\frac{\sigma_0\sqrt{e}}{\sigma_1}\right)^d$$

4 Exploration of Ratio of Level Set Volumes

The estimation of the ratio of level set volumes $R_{i:i+1}$ for each level set i is a necessary step in either of our LSHR1 or LSHR2 sampling procedures. For this section, we use the more compact notation $R_i = V_i/V_{i+1}$ for the volume ratio of level sets i and $i+1$. These ratios R_i are crucial in two senses.

First, in order to ensure that each move to a new level set is "warm", we would like a ratio of volumes R_i that is greater than 0.5 (as suggested by Vempala (2005)). Second, we need the cumulative ratio of all level set volumes in order to calculate the probability of each level set, so that we can subsample from that level set with the appropriate weight.

In this section, we explore two aspects of the volume ratios R_i: (1) the accuracy of the cumulative ratio of all level set volumes, and (2) the number of hit-and-run iterations to run within each proposed level set in order to estimate each R_i.

The cumulative ratio of all level set volumes, $\mathcal{R}$, is the product of each ratio of volumes,

$$\mathcal{R} = \frac{V_0}{V_n} = \prod_{i=0}^{n-1} \frac{V_i}{V_{i+1}} = \prod_{i=0}^{n-1} R_i$$

We estimate $\mathcal{R}$ with individual ratios of level set volumes

$$\widehat{\mathcal{R}} = \prod_{i=0}^{n-1} \hat{R}_i$$

where $\hat{R}_i$ is estimated as the proportion of samples taken from level set $i+1$ which are also contained in level set i. Consider the log ratio of the estimated $\widehat{\mathcal{R}}$ to the true $\mathcal{R}$,

$$\log \frac{\widehat{\mathcal{R}}}{\mathcal{R}} = \sum_{i=0}^{n-1} \log \frac{\hat{R}_i}{R_i} = \sum_{i=0}^{n-1} \log\left(1 + \frac{\hat{R}_i - R_i}{R_i}\right)$$

By Taylor series expansion,

$$\log \frac{\widehat{\mathcal{R}}}{\mathcal{R}} \approx \sum_{i=0}^{n-1} \frac{\hat{R}_i - R_i}{R_i} - \frac{1}{2}\left(\frac{\hat{R}_i - R_i}{R_i}\right)^2$$

Now if we assume that $\hat{R}_i - R_i = o(1/n)$ then we have no significant bias in the above expression. If we further assume that each ratio of level set volumes calculation is independent, then the variance can be calculated as

$$\text{Var}\left(\log \frac{\widehat{\mathcal{R}}}{\mathcal{R}}\right) \approx \sum_{i=0}^{n-1} \text{Var}\left(\frac{\hat{R}_i - R_i}{R_i}\right) \approx n \frac{\text{Var}(\hat{R}_\star)}{R_\star^2} \tag{1}$$

where the last approximation in (1) comes from setting the R_i's to be approximately equal, $R_i = R_\star = \mathcal{R}^{1/n}$ which is a characteristic of our level-set hit-and-run sampler. Now, let T be the total number of samples taken by our sampler, and T/n be the number of samples dedicated to level set i, then

$$\text{Var}(\hat{R}_\star) = k(d)\, \frac{n\, R_\star\, (1 - R_\star)}{T}$$

where $k(d)$ is a function that specifies how the dependance between samples scales with dimension d. This means that

$$\text{Var}\left(\log \frac{\widehat{\mathcal{R}}}{\mathcal{R}}\right) \approx k(d) \frac{n^2}{T} \cdot \frac{1 - R_\star}{R_\star} \approx k(d) \frac{(\log \mathcal{R})^2}{T} \cdot \frac{1 - R_\star}{R_\star\, (\log R_\star)^2} \tag{2}$$

since $\log R_\star = n^{-1} \log \mathcal{R}$. We can try to minimize (2) as a function of $R_\star$. In Figure 1, we plot $g(R_\star) = (1 - R_\star)/R_\star (\log R_\star)^2$ as a function of $R_\star$. The function $g(R_\star)$ is minimized at $R_\star \approx 0.204$.

However, as outlined in our manuscript, we choose the more conservative restriction of $R_\star \geq 0.5$ to ensure that each level set is a warm start for the next level set. Fortunately, the function $g(R_\star)$ appears quite flat for a wide range $0.1 \leq R_\star \leq 0.6$, so we can be reassured that our conservative restriction of $R_\star \geq 0.5$ does not sacrifice much efficiency relative to the optimal value in terms of minimizing the variance of our estimated cumulative ratio of level set volumes $\widehat{\mathcal{R}}$.

Another important factor in the estimation of each level set volume ratio is the number of iterations m that the hit-and-run algorithm is run within each level set. We explore this issue by picking an arbitrary pair of level sets from our spike-and-slab mixture example (Section 3 of paper), and running a small simulation study of our estimation of level set volumes for different numbers of iterations m.

Specifically, the hit-and-run algorithm was run for m iterations and the ratio of volumes R was estimated using the hit-and-run samples for the chosen pair of level sets. This process was repeated 100 times, giving us a distribution of estimated level set volumes $\hat{R}$ for that value of m.

In Figure 2, we give the distributions of the estimated ratio of level set volumes for a different values of m, in the case where $d = 20$. We see that for small

Figure 1: The function $g(R_\star) = (1 - R_\star)/R_\star(\log R_\star)^2$ plotted as a function of $R_\star$. The red line indicates the minimizing point $R_\star \approx 0.204$.

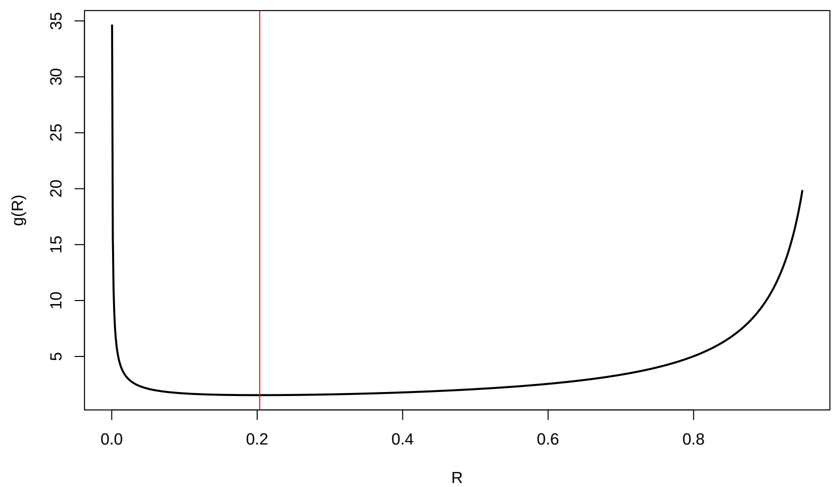
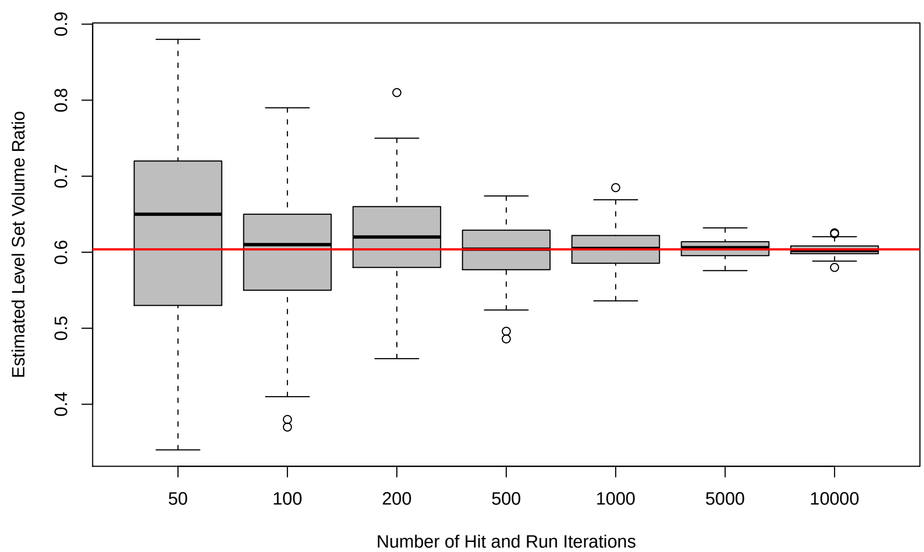
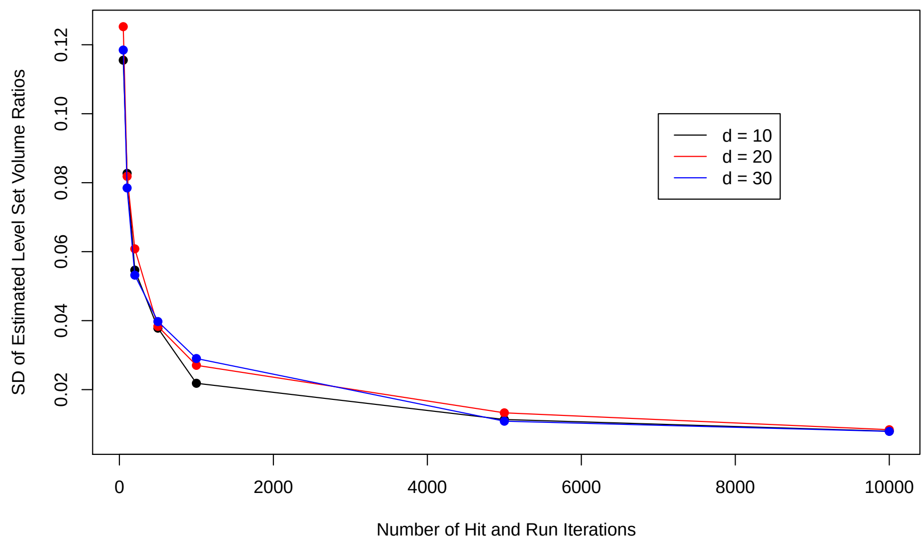


Figure 2: Estimated ratio of volumes for different numbers of iterations of the level-set hit-and-run sampler. Red line indicates the true ratio of volumes based on the spike-and-slab density. This plot is for dimension $d = 20$ though the same trend is seen for other dimensions.



values of m, the estimated ratio of level set volumes are highly variable around the true value (given by the red line). We see in Figure 2 that the estimated ratio of volumes are quite variable for a small number of iterations, and it seems that at least $m = 1000$ iterations is needed to get a reasonably low-variance estimate. In Figure 3, we see that this same trend occurs for a variety of dimensions d: the main reduction in variance (across simulations) occurs within the first 1000 iterations of the hit-and-run algorithm. These simulation results motivated our choice of $m = 1000$ in our empirical studies.

Figure 3: Standard deviation of the estimates (across simulations) of the ratio of volumes between two chosen level sets for different numbers of iterations of the level-set hit-and-run sampler.



5 Rotation Procedure for True Cauchy-Normal Posterior Density

The true posterior density $h(\boldsymbol{\theta}|\boldsymbol{y})$ for our Cauchy-normal model is difficult to directly evaluate in dimensions higher than $d = 2$ (for $d = 1$ or $d = 2$, we can just evaluate the posterior density on a fine grid of values).

The Gibbs sampler does a very poor job of estimating this posterior density when we set $\boldsymbol{y} = (10, 10, \ldots, 10)$. However, the Gibbs sampler is able to accurately estimate the posterior density in a single dimension (when $y = 10$). If we project our higher-dimensional $\boldsymbol{y} = (10, 10, \ldots, 10)$ onto a single axis $\boldsymbol{w} = (10\sqrt{d}, 0, \ldots, 0)$, then the Gibbs sampler is able to accurately estimate the posterior density for projected data $\boldsymbol{w}$.

Our true data location $\boldsymbol{y}$ is a rotation $\boldsymbol{y} = \mathbf{R}\boldsymbol{w}$ of the single axis $\boldsymbol{w}$. Letting $\boldsymbol{\theta} = \mathbf{R}\boldsymbol{\mu}$, where $\mathbf{R}$ is a rotation matrix, we observe that

$$h(\boldsymbol{\theta}|\boldsymbol{y}) = h(\boldsymbol{\mu}|\boldsymbol{w})$$

We use Gibbs sampling to accurately estimate $h(\boldsymbol{\mu}|\boldsymbol{w})$, and then undo the rotation $\boldsymbol{y} = \mathbf{R}\boldsymbol{w}$ to get an accurate estimate of $h(\boldsymbol{\theta}|\boldsymbol{y})$.

Note that this procedure is not a general "fix" for the poor performance of Gibbs sampling in high-dimensional models, since we take advantage of the rotational symmetry in this simple Cauchy-normal model that may not be present in more complicated situations.

References

Vempala, S. (2005). Geometric random walks: A survey. *Combinatorial and Computational Geometry* **52**, 573–612.