Polar Slice Sampler

Justin Zhang
University of Toronto
justinj.zhang@mail.utoronto.ca

May 1, 2025

1 MCMC Basics

Markov Chain Monte Carlo methods are a set of common algorithms used for sampling from complex and high-dimensional target distributions that are intractable. There is a wide range of literature surrounding this topic, and numerous algorithms used, including Metropolis-Hastings (many variations), Gibbs Sampler, slice sampler, etc. The premise is that we sample a Markov Chain $X_1, X_2, ...$ that is eventually distributed approximately equal to the target distribution. In this section we will introduce some basic stochastic theory then formalize convergence properties of MCMC methods that justify its use. For a more detailed intro to Markov chains and stochastic theory, read *A First look at Stochastic Processes* by Jeffrey Rosenthal.

Definition 1.1 (Markov Chain)**.** *Let $X_1, X_2, ...$ be a time-indexed stochastic process in $\mathbb{R}^d$. We call this process a time-homogeneous Markov chain (Markov process) if $\forall t \in \mathbb{N}$ and $y \in \mathbb{R}^d$*

$$\mathbb{P}(X_t \in dy | X_i = x_i \; \forall i = 1, ..., t-1) = \mathbb{P}(X_t \in dy | X_{t-1} = x_{t-1}) = \mathbb{P}(X_1 \in dy | X_0 = x_0)$$

Denote this transition probability as

$$P(x, dy) = \mathbb{P}(X_t \in dy | X_{t-1} = x)$$

The first equality satisfies the Markov property, meaning each time-step of the chain is only dependent on the previous time-step. The second equality satisfies time-homogeneity, meaning each time-step X_t of the chain is not dependent on the time t.

For simplicity, in the rest of this report, we assume that the given transition probability admits a valid density. This is because the transition kernels we will introduce are probability densities. We can write

$$P(x, y) = \mathbb{P}(X_t = y | X_{t-1} = x)$$

Definition 1.2 (Stationary Distribution)**.** *Let $X_1, X_2, ...$ be Markov Chain with transition probability density P. We say that a probability distribution with density π is stationary for P if for some $t \in \mathbb{N}$ $X_t \sim \pi \implies X_{t+1} \sim \pi$. That is* $\forall y \in \mathbb{R}^d$

$$\int_{\mathbb{R}^d} \pi(x)P(x,y)dx = \pi(y)$$

The condition requires that the total probability density entering state y at time $t+1$ (the LHS) is the same as the original probability density of y at time t (RHS). This implies that if somewhere along the chain, the stationarity condition holds for π, every subsequent step will also have distribution π. In discrete cases, we get the simple representation $\pi P = \pi$.

Definition 1.3 (Irreducibility)**.** *Let $X_1, X_2, ...$ be Markov Chain with transition probability density P and stationary distribution with density π. This chain is π-irreducible if* $\forall x, y \in \mathbb{R}^d$

$$\pi(y) > 0 \implies \exists n \in \mathbb{N},\ P(x,y)^n > 0$$

This condition ensures that every state A that has positive density is reachable from every point x. This is essential for MCMC sampling as it ensures the entire support can be reached.

Definition 1.4 (Reversibility)**.** *Let $X_1, X_2, ...$ be Markov Chain. It is reversible if for any sequence of times $t_1 < ... < t_n < T \in \mathbb{N}$. The random vectors $(X_{t_1}, ..., X_{t_n})$ and $(X_{T-t_1}, ..., X_{T-t_n})$ have the same distribution.*

Definition 1.5 (Detailed Balance Condition)**.** *Let $X_1, X_2, ...$ be Markov Chain with transition probability density P. Let π be some density function. We say that it satisfies the detailed balance equation w.r.t. π if* $\forall x \neq y \in \mathbb{R}^d$

$$\pi(x)P(x,y) = \pi(y)P(y,x)$$

Proposition 1.6. *If a Markov chain $X_1, X_2, ...$ satisfies detailed balance w.r.t. to the distribution of X_0 then it is reversible.*

Proof. Assume that detailed balance is satisfied and say X_0 has density π. Consider the chain $(X_0, X_1, ..., X_T)$. Let $x_0, ..., x_T$ be sequence of in $\mathbb{R}^d$. For any $j = 0, ..., T-1$

$$\pi(x_{j+1})\mathbb{P}(X_1 = x_j | X_0 = x_{j+1}) = \pi(x_j)\mathbb{P}(X_1 = x_{j+1} | X_0 = x_j)$$
$$\implies \mathbb{P}(X_1 = x_j | X_0 = x_{j+1}) = \frac{\pi(x_j)\mathbb{P}(X_1 = x_{j+1} | X_0 = x_j)}{\pi(x_{j+1})}$$

Then by iterative application of Markov property, time homogeneity and conditional probability

$$\begin{aligned}
\mathbb{P}(X_T = x_T, ..., X_0 = x_0) &= \pi(x_0) \prod_{i=1}^{T} \mathbb{P}(X_i = x_i | X_{i-1} = x_{i-1}, ..., X_0 = x_0) \\
&= \pi(x_0) \prod_{i=1}^{T} \mathbb{P}(X_i = x_i | X_{i-1} = x_{i-1}) \\
&= \pi(x_0) \prod_{i=1}^{T} \mathbb{P}(X_1 = x_i | X_0 = x_{i-1}) \\
\mathbb{P}(X_T = x_0, ..., X_0 = x_T) &= \pi(x_T) \prod_{i=1}^{T} \mathbb{P}(X_i = x_{T-i} | X_{i-1} = x_{T-i+1}, ..., X_0 = x_T) \\
&= \pi(x_T) \prod_{i=1}^{T} \mathbb{P}(X_1 = x_{T-i} | X_0 = x_{T-i+1}) \\
&= \pi(x_T) \prod_{i=1}^{T} \frac{\mathbb{P}(X_1 = x_{T-i+1} | X_0 = x_{T-i}) \pi(x_{T-i})}{\pi(x_{T-i+1})} \\
&= \pi(x_T) \prod_{j=1}^{T} \frac{\mathbb{P}(X_1 = x_j | X_0 = x_{j-1}) \pi(x_{j-1})}{\pi(x_j)} \\
&= \pi(x_0) \prod_{i=1}^{T} \mathbb{P}(X_1 = x_j | X_0 = x_{i-1})
\end{aligned}$$

In the second last line, all the $\pi(x_j)$ terms cancel by a telescoping argument. Here, we apply detailed balance in the 7th line, which can also be seen as Bayes law. In the 8th line, we substitute $j = T - i + 1$. Since these distribution are the same for any $T \in \mathbb{N}$, we conclude the chain is reversible. □

Proposition 1.7. *If a Markov chain $X_1, X_2, ...$ is reversible then it has a stationary distribution with density π.*

Proof. Assume the Markov chain is reversible and that $X_t \sim \pi$ for some distribution with density π. Then for some $x \in \mathbb{R}^d$, using detailed balance condition and the fact conditional probability integrates to 1

$$\begin{aligned}
\mathbb{P}(X_{t+1} = dx) &= \int_{\mathbb{R}^d} \mathbb{P}(X_{t+1} = x | X_t = y) \pi(y) dy \\
&= \int_{\mathbb{R}^d} \mathbb{P}(X_{t+1} = y | X_t = x) \pi(x) dy \\
&= \pi(x) \int_{\mathbb{R}^d} \mathbb{P}(X_{t+1} = y | X_t = x) dy \\
&= \pi(x)
\end{aligned}$$

Hence $X_{t+1} \sim \pi$ as well, and so π is stationary distribution. $\square$

These propositions show that whenever a Markov chain satisfies detailed balance (equivalently, it is reversible), then a stationary distribution exists. This allows us to show convergence of MCMC algorithms (more later).

Definition 1.8 (Monte Carlo Estimation)**.** *Let $X_i \sim F$ be i.i.d. random variables with $E(X_1) = \mu, \mathrm{Var}(X_1) = \sigma^2$. Let h be an integrable function w.r.t. F, that is $E_F(|h(X)|) < \infty$. Then*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} h(X_i) = E_F(h(X)) = \int_{\mathbb{R}^d} h(x) dF(x)$$

This shows that we can estimate the expectation of h, and hence the Lebesgue integral of h with realizations of $h(X_i)$. This limiting behaviour holds as a consequence of the weak law of large numbers.
A Markov Chain Monte Carlo method combines the concepts of Monte Carlo estimation with Markov chains. In this case, our samples $\{X_i\}_{i=1}^{\infty}$ are not i.i.d. but rather a Markov chain, We can still show the convergence of MCMC estimates.

Theorem 1.9 (Convergence of MCMC)**.** *Let $X_1, X_2, ...$ be a Markov Chain with transition probability P and stationary distribution π. Suppose it is π-irreducible. Let h be a function with $E_\pi(|h|) < \infty$. Then $\forall b > 0$*

$$\lim_{n\to\infty} \frac{1}{n-b} \sum_{i=b}^{n} h(X_i) = E_\pi(h)$$

This holds for all initial values $X_0 = x$ π-a.e. That is, for all values x in the state space except for a measure 0 set w.r.t. π. The value b is known as the burn-in point.

The Markov chain achieves stationarity at a burn-in point b, and hence realizations X_i for $i > b$ are from the target distribution. This does not effect the limiting behaviour of the expectation as the points $X_1, ..., X_{b-1}$ have no effect on the mean as n goes to infinity. In practice, there's extensive literature on how to determine the burn-in point. This theorem shows that for large n, the Markov chain sample will allow us to determine properties of the target distribution π. There are many other conditions that can ensure the convergence of Markov chains. We will introduce one more (stronger) convergence property of MCMC algorithms, based on the property of geometric ergodicity.

Definition 1.10 (Geometric Ergodicity)**.** *Let $X_1, X_2, ...$ be a Markov Chain with transition probability P and stationary distribution π. We say it is geometrically ergodic if there exists some function $M : \mathbb{R}^d \to \mathbb{R}$ and constant $\rho < 1$ so that for $x \in \mathbb{R}^d$ π-a.e. and $n \in \mathbb{N}$*

$$\sup_{y\in\mathbb{R}^d} |P^n(x, dy) - \pi(dy)| = ||P^n(x, \cdot) - \pi|| \leq M(x)\rho^n$$

Here, the distance metric used is the *total variation distance*.

Theorem 1.11 (Convergence of MCMC 2)**.** *Let $X_1, X_2, ...$ be a geometrically ergodic Markov Chain with transition probability P and stationary distribution π.Let h be a function with $E_\pi(|h|^{2+\delta}) < \infty$ for some $\delta > 0$. Then the Central Limit Theorem holds. That is, $\frac{1}{n}\sum_{i=1}^{n} h(X_i)$ converges to a (multivariate) normal in distribution.*

Proofs of these convergence properties will be omitted, but can be found in *General State Space Markov Chains and MCMC Algorithms* by Rosenthal and Roberts.

2 Sampling Algorithms

Now that we have a basic understanding of what MCMC methods are and why they can be used for sampling, we will introduce some algorithms that allow us to generate Markov chains that converge to a desired target distribution. We start with a frequently used algorithm in Metropolis-Hastings, and then touch on rejection sampling and slice sampling.

Definition 2.1 (Metropolis Hastings)**.** *Let π be the target probability density known up to a normalizing constant. We sample a Markov chain $\{X_t\}$ from π. Let g be a proposal distribution that is easily sampled from. Choose an initial value X_0. For time $t = 1, 2, ...,$ given the value of X_{t-1} the algorithm iterates as follows:*

1. *Sample a point $Y \sim g(\cdot|X_{t-1})$*
2. *Set $P(X_{t-1}, Y) = \min\{\frac{\pi(Y)g(X_{t-1}|Y)}{\pi(X_{t-1})g(Y|X_{t-1})}, 1\}$. This is called the Metropolis Hastings acceptance probability.*
3. *Sample $U \sim$ Uniform$(0, 1)$*
4. *Set $X_t = Y$ if $U < P(X_{t-1}, Y)$, otherwise set $X_t = X_{t-1}$*

As we can see, this is algorithm will generate a Markov chain by construction (i.e. x_t only depend on x_{t-1}). We have irreducibility provided that for all x with $\pi(x) > 0$, there is some n with $g(x|x_{t-1})^n > 0$, meaning there is some sequence of iterations $x_{t-1} = x_1, x_2, ..., x_n = x$ where each $g(x_i|x_{i-1}) > 0$. We show that this chain $\{X_t\}$ satisfies the detailed balance condition (which implies reversibility and hence stationarity) with respect to the transition kernel K. Though this is a density, we should note the the probability of staying in the same state, $\mathbb{P}(X_t = x|X_{t-1} = x)$ is positive on support since the acceptance probabilities $P(x, x)$ are non-zero.

$$K(x'|x) = g(x'|x)P(x, x') + \delta_x(x') \int_{\mathbb{R}^d} q(y|x)(1 - P(x, y)dy$$

Proposition 2.2 (Metropolis Hastings satisfies detailed balance)**.** *The MH algorithm from definition 2.1 satisfies the detailed balance condition in 1.5*

Proof. We will show that both terms of the kernel K satisfy detailed balance. Let $r(x) = \int_{\mathbb{R}^d} q(y|x)(1 - P(x, y)dy$. We have that

$$(\delta_x(x') = \delta_{x'}(x) \neq 0 \implies x = x' \implies \pi(x)r(x)\delta_x(x') = \pi(x')r(x')\delta_{x'}(x)$$

For the first term of K, we will show LHS and RHS have the same value.

$$\begin{aligned}
\pi(x)g(x'|x)P(x|x') &= \pi(x)g(x'|x)\min\left\{\frac{\pi(x')g(x|x')}{g(x'|x)\pi(x)}, 1\right\} \\
&= \begin{cases} \pi(x')g(x|x') & \pi(x')g(x|x') > g(x'|x)\pi(x) \\ \pi(x)g(x'|x) & \pi(x')g(x|x') \leq g(x'|x)\pi(x) \end{cases} \\
\pi(x')g(x|x')P(x'|x) &= \pi(x')g(x|x')\min\left\{\frac{\pi(x)g(x'|x)}{g(x|x')\pi(x')}, 1\right\} \\
&= \begin{cases} \pi(x')g(x|x') & \pi(x')g(x|x') > g(x'|x)\pi(x) \\ \pi(x)g(x'|x) & \pi(x')g(x|x') \leq g(x'|x)\pi(x) \end{cases}
\end{aligned}$$

We conclude that $\pi(x)g(x'|x)P(x|x') = \pi(x')g(x|x')P(x'|x)$ and thus detailed balance is sataisfied. □

If detailed balance is satisfied then π is the stationary density of $\{X_t\}$ from propositions 1.6 and 1.7 and the nice convergence properties stated in section 1) will hold.

Definition 2.3 (Rejection Sampling)**.** *Let π be the target probability density known up to a normalizing constant. Choose another probability density f and a constant $K \in \mathbb{R}$ with $Kf(x) \geq \pi(x)$ π-a.e. on $\mathbb{R}^d$. Then we sample X_t as follows*

1. *Sample a point $Y \sim f$*
2. *Sample $U \sim Uniform(0, 1)$*
3. *Set $X_t = Y$ if $U \leq \frac{\pi(Y)}{Kf(Y)}$, otherwise reject Y and return to step 1*

Rejection sampling is part of a family of accept-reject algorithms. The resulting chain $\{X_t\}$ is composed of independent samples X_t. We now prove that the resulting chain in fact comes from the distirbution π.

Proposition 2.4 (Validity of Rejetion Sampler)**.** *Conditional on acceptance criteria $U \leq \frac{\pi(Y)}{Kf(Y)}$ being satisfied, the sampled point $X_t = Y \sim \pi$.*

Proof.

$$\begin{aligned}
\mathbb{P}\left(Y \leq x | U \leq \frac{\pi(Y)}{Kf(Y)}\right) &= \frac{\int_{\mathbb{R}^d} \mathbb{P}(Y \leq x, U \leq \frac{\pi(Y)}{Kf(Y)} | Y = y)\mathbb{P}(Y = y)dy}{\mathbb{P}(U \leq \frac{\pi(Y)}{Kf(Y)})} \\
&= \frac{\int_{\mathbb{R}^d} \frac{\pi(y)}{Kf(y)} 1_{y \leq x} f(y)dy}{\frac{1}{K}} \\
&= \frac{\frac{1}{K}\int_{\{y \in \mathbb{R}^d : y \leq x\}} \pi(y)dy}{\frac{1}{K}} \\
&= \int_{\{y \in \mathbb{R}^d : y \leq x\}} \pi(y)dy
\end{aligned}$$

So indeed we have $X_t \sim \pi$ and our sampling algorithm will recover π □

Assuming π to be a normalized density, the acceptance probability of each draw from $Y \sim f$ is

$$\begin{aligned}\mathbb{P}\left(U \leq \frac{\pi(Y)}{Kf(Y)}\right) &= \int_{\mathbb{R}^d} \mathbb{P}\left(U \leq \frac{\pi(Y)}{Kf(Y)} | Y = y\right) \mathbb{P}(Y = y) dy \\ &= \int_{\mathbb{R}^d} \frac{\pi(y)}{Kf(y)} f(y) dy \\ &= \frac{1}{K}\end{aligned}$$

We can see from this, that as K decreases, the acceptance probability increases, so it is computationally efficient to find a density f that is somewhat close to π so that the constant K would be smaller. However, we must have $K \geq 1$ for the condition to hold.

Definition 2.5 (Slice Sampling)**.** *Let π be the target probability density known up to a normalizing constant. Decompose π as*

$$\pi(x) = f_0(x) f_1(x)$$

Choose an initial value X_0. For time $t = 1, 2, ...,$ given the value of X_{t-1} the algorithm iterates as follows:

1. $Y_t \sim Uniform(0, f_1(X_{t-1}))$
2. $X_t \sim f_0(x) 1_{\{0 \leq Y_t \leq f_1(x)\}}(x)$

In the case that f_0 is a constant, we call the algorithm the *uniform slice sampler*. Let g be the density of $X_t|Y_t$ and h be the density of $Y_t|X_{t-1}$. The transition kernel for this chain is defined as

$$\begin{aligned}P(x'|x) &= \int_{\mathbb{R}^d} g(x'|y) h(y|x) dy \\ &= \int_{\mathbb{R}^d} \frac{f_0(x')}{Q(y)} 1_{\{0 \leq y \leq f_1(x')\}} \frac{1}{f_1(x)} 1_{\{0 \leq y \leq f_1(x)\}} dy \\ &= \frac{1}{f_1(x)} \int_0^{f_1(x)} \frac{f_0(x') 1_{\{f_1(z) \geq y\}}}{Q(y)} dy\end{aligned}$$

where Q is the normalizing constant for $f_0(x') 1_{\{0 \leq Y_t \leq f_1(x)\}} | Y + t = y$. Another view is that Q is the stationary (unnormazlized) marginal density of Y_t for the distribution $f(x, y) = f_0(x) 1_{\{0 \leq Y_t \leq f_1(x)\}}$

$$Q_{f_0, f_1}(y) = \int_{\mathbb{R}^d} f_0(z) 1_{\{f_1(z) \geq y\}} dz \tag{2.1}$$

Proposition 2.6 (Slice Sampling satisfies detailed balance)**.** *The slice sampling algorithm, as defined in 2.5 satisfies the detailed balance condition.*

Proof. We show the LHS and RHS in the detailed balance equation 1.5 are equal.

$$\begin{aligned}
\pi(x)P(x'|x) &= f_0(x)f_1(x)\frac{1}{f_1(x)}\int_0^{f_1(x)} \frac{f_0(x')1_{\{f_1(x')\geq y\}}}{Q(y)}dy \\
&= f_0(x)f_0(x')\int_0^{f_1(x)} \frac{1_{\{f_1(x')\geq y\}}}{Q(y)}dy \\
&= f_0(x)f_0(x')\int_0^{\min\{f_1(x),f_1(x')\}} \frac{1}{Q(y)}dy \\
\pi(x')P(x|x') &= f_0(x')f_1(x')\frac{1}{f_1(x')}\int_0^{f_1(x')} \frac{f_0(x)1_{\{f_1(x)\geq y\}}}{Q(y)}dy \\
&= f_0(x)f_0(x')\int_0^{f_1(x')} \frac{1_{\{f_1(x)\geq y\}}}{Q(y)}dy \\
&= f_0(x)f_0(x')\int_0^{\min\{f_1(x),f_1(x')\}} \frac{1}{Q(y)}dy
\end{aligned}$$

From propositions 1.6 and 1.7 we know that the slice sampler admits a stationary distribution, and hence the resulting chain converges. □

In particular, if we have the uniform slice sampler, i.e. f_0 is constant, the stationary distribution of (X_t, Y_t) in $\mathbb{R}^d \times \mathbb{R}$ is the uniform distribution of the region under the graph of π. Step 2 in the algorithm can still be somewhat difficult to perform based on the difficulty of sampling from f_0 (not a problem for uniform), and computing the region that is above the graph of f_1. There are many ways to do this including with rejection sampling with f_0 as the proposal distribution and $K = 1$. The acceptance probability will not be 1 as we discussed earlier since $f_0(x)1_{\{0\leq Y_t\leq f_1(x)\}}(x)$ is unnormalized. More details can be found in the paper *Slice Sampler* by Radford Neal. We will construct a specific sub-method of the slice sampler (with rejection sampling) in the next section. For a direct use case of the slice sampler, we give a specific condition for when it is guaranteed to converge.

Proposition 2.7. *Consider the slice sampling algorithm on any target density $\pi(x) = f_0(x)f_1(x)$ in $\mathbb{R}^d$. If $yQ'(y)$ is non-increasing on $y \leq Y$ for some $0 \leq Y \leq 1$ and Q in equation 2.1, then the slice sampler converges. That is*

$$\forall \epsilon > 0,\ \exists N_Y \ s.t.\ \forall n \geq N_Y \quad ||\mathbb{P}(X_n \in \cdot | X_0 = x) - \nu_\pi(\cdot)|| < \epsilon$$

Here, we use total variation distance defined in the first section. The proof is omitted but can be found in the original *Polar Slice Sampling paper* by Rosenthal and Roberts. Moreover, they determine the exact value N_y needed for various $Y \in [0, 1]$ at level $\epsilon = 0.01$. As Y decreases to 0, the number of iterations needed to converge at level $\epsilon = 0.01$ increases rapidly, since our 'nice' property of $yQ'(y)$ non-increasing fails to hold for increasing proportion of points.

3 Polar Slice Sampler

The previous section explored some basic sampling algorithms, which we can now use to define the polar slice sampler. The (uniform) slice sampler is seen to perform poorly in high dimensions, due to high autocorrelation, which makes convergence incredibly slow (or impossible). The polar slice sampler is a version of slice sampling with a specific choice of functions f_0, f_1 that allow for better performance in higher dimensions, especially with spherically symmetrical distributions. Where the standard slice sampler takes a horizontal slice of the support at Y_t and samples from those points, the polar slice sampler takes a spherical slice of radius R_t and samples on the surface. We first introduce the algorithm. We will use the euclidean norm

$$|x| = \sqrt{x_1^2 + ... + x_d^2}$$

Definition 3.1 (Polar Slice Sampler)**.** *Let π be the target probability density known up to a normalizing constant. Decompose $\pi(x) = f_0(x)f_1(x)$ where*

$$f_0(x) = |x|^{-(d-1)}, \qquad f_1(x) = |x|^{d-1}\pi(x)$$

Choose an initial value X_0. For time $t = 1, 2, ...,$ given the value of X_{t-1} the algorithm will iterate just as in the standard slice sampler (def 2.5)

1. $Y_t \sim$ *Uniform*$(0, |X_{t-1}|^{d-1}\pi(X_{t-1}))$
2. $X_t \sim |x|^{-(d-1)}1_{\{Y_t \leq |x|^{d-1}\pi(x)\}}(x)$

We can define the transition probability kernel (w.r.t function Q) as in 2.1 for a general slice sampler. This choice of the function f_0 allows us to show fast convergence of the Markov chain for (more later) through nice properties of Q. It also allows gives us a trick to sample from the density in step 2) of the algorithm, based on rejection sampling 2.3

Proposition 3.2 (Sampling from $f_0(x)1_{f_1(x)\geq y}(x)$)**.** *Assume we have sample Y_t from step 1 of the polar slice sampler algorithm, and wish to sample from the density in part 2 using polar coordinates. To that end, we wish to define $X_t = R_t\theta_t$ for magnitude $R_t \geq 0$ and $|\theta_t| = 1$ on the unit sphere. We then perform the following steps*

1. *Sample $R_t \sim$ Uniform$(0, R_t^*)$ where $R_t^* \geq \sup\{|x||x|^{d-1}\pi(x) \geq Y_t\}$*
2. *Sample $Z_1, ..., Z_d \sim N(0, 1)$ independently. Denote $Z = (Z_1, ..., Z_d)$*
3. *Set $\theta_t = \frac{Z}{|Z|}$ and let $X_t = R_t\theta_t$*
4. *If $|X_t|^{d-1}\pi(X_t) \geq Y_t$, accept this choice of X_t, otherwise restart from step 1*

Here we are defining R_t^* such that a ball of radius R_t^* is fully outside the truncated distribution of f_1 (equivalent to the function $Kf(x)$ in rejection sampling 2.3). We then uniformly sample a radius R_t less than R_t^*. Next, we randomly select an angle θ_t, and define X_t to be the point on the ball of radius R_t with angle θ_t. The closer the ball of radius R_t^* comes to touching the distribution f_1, the smaller the radius R_t and less likely we are to reject. Since the choice of θ_t is completely random, we can see that polar slice sampling would be less efficient for asymmetric distributions, since entire 'quadrants' may have magnitude less than R_t under the distribution.

We now aim to present a convergence property for the polar slice sampler.

Definition 3.3 (Asymmetry Parameter)**.** *Given a density $\pi(x) = f_0(x)f_1(x)$ in $\mathbb{R}^d$, we define the asymmetry parameter as*

$$A(f_1) = \frac{\inf_\theta M(f_1, \theta)}{\sup_\theta M(f_1, \theta)}$$
$$\text{where} \qquad M(f_1, \theta) = \sup\{f_1(t\theta); t \geq 0\}$$

Lemma 3.4 (Condition for $yQ'(y)$)**.** *Consider the polar factorization $\pi(x) = f_0(x)f_1(x)$ in $\mathbb{R}^d$ with $f_0(x) = |x|^{-(d-1)}$, $f_1(x) = |x|^{(d-1)}\pi(x)$. Suppose $\pi(x)$ is log-concave, then $yQ'(y)$ is non-increasing for $y \leq A(f_1)$*

Proof. Recall, the definition of log-concavity is that

$$f(\lambda x + (1-\lambda)y) \geq f(x)^\lambda f(y)^{1-\lambda} \qquad \forall x, y, \in \mathbb{R}^d, 0 < \lambda < 1$$

Since $|x|^{d-1}$ is also log-concave (polynomial), we know that f_1 is log-concave. Moreover, this implies f_1 is also continuous. We start by transforming Q to polar coordinates $(x_1, ..., x_d) \to (r, \theta_1, ..., \theta_d)$. From calculus we have

$$r = |x| = \sqrt{x_1^2 + ... + x_d^2} \implies dx = r^{d-1} dr d\theta \implies |x|^{-(d-1)} dx = dr d\theta$$

where dr is Lebesgue measure on $\mathbb{R}$ and $d\theta$ is Lebesgue measure on surface of unit sphere in $\mathbb{R}^d$. Then we get that

$$\begin{aligned} Q(y) &= \int_{\mathbb{R}^d} |x|^{-(d-1)} \mathbf{1}_{f_1(x)} dx \\ &= \int_{\mathbb{R}^d} \mathbf{1}_{f_1(r,\theta)} dr d\theta \\ &= \int_{\mathbb{R}^d} (R^+(y,\theta) - R^-(y,\theta)) d\theta \end{aligned}$$

where $R^-(y,\theta) = \inf\{r : f_1(r,\theta) \geq y\}, R^+(y,\theta) = \sup\{r : f_1(r,\theta) \geq y\}$. Note of course that $f_1(r,\theta) \in \mathbb{R}$. For a fixed θ, R^-, R^+ are functions of y. Since f_1 is continuous by log-concavity, the inf, sup respectively are attained, and R^-, R^+ are continuous. Moreover, we have $R^+(y,\theta) = f_1^{-1}(r,\theta) = R^-(y,\theta)$, which is possible

since f_1 is (generally) not a bijective function, this is possible even though $R^- \neq R^+$ (if it was bijective, we would have $R^- = R^+$). By the inverse function theorem we know

$$\begin{aligned}
Q'(y) &= \int_{\mathbb{R}^d} \frac{\partial}{\partial r}(R^+(y,\theta) - R^-(y,\theta))d\theta \\
&= \int_{\mathbb{R}^d} \left(\frac{1}{f_1'(f_1^{-1}(r,\theta),\theta)} - \frac{1}{f_1'(f_1^{-1}(r,\theta),\theta)} \right) d\theta \\
&= \int_{\mathbb{R}^d} \left(\frac{1}{f_1'(R^+(y,\theta),\theta)} - \frac{1}{f_1'(R^-(y,\theta),\theta)} \right) d\theta \\
yQ'(y) &= \int_{\mathbb{R}^d} \left(\frac{y}{f_1'(R^+(y,\theta),\theta)} - \frac{y}{f_1'(R^-(y,\theta),\theta)} \right) d\theta \\
&= \int_{\mathbb{R}^d} \left(\frac{f_1(R^+(y,\theta),\theta)}{f_1'(R^+(y,\theta),\theta)} - \frac{f_1(R^-(y,\theta),\theta)}{f_1'(R^-(y,\theta),\theta)} \right) d\theta \\
&= \int_{\mathbb{R}^d} \left(\frac{1}{\frac{\partial}{\partial r}\log f_1(R^+(y,\theta),\theta)} - \frac{1}{\frac{\partial}{\partial r}\log f_1(R^-(y,\theta),\theta)} \right) d\theta
\end{aligned}$$

Consider a fixed θ. Since the set $\{r : f_1(r,\theta) \geq y\}$ is non-increasing as y increases, we have by definition that R^+ is a non-increasing function and R^- a non-decreasing function. Since $f_1(x)$ is log-concave, we know that $\log f_1(x)$ is a concave function, and hence its derivative is always non-increasing. These facts imply that $\frac{\partial}{\partial r}\log f_1(R^+(y,\theta),\theta)$ and $\frac{\partial}{\partial r}\log f_1(R^-(y,\theta),\theta)$ are non-decreasing and non-increasing functions respectively. Taking reciprocals, we get that $\frac{1}{\frac{\partial}{\partial r}\log f_1(R^+(y,\theta),\theta)}$ is a non-increasing function in y and $\frac{1}{\frac{\partial}{\partial r}\log f_1(R^-(y,\theta),\theta)}$ is a non-decreasing function in y, which implies $-\frac{1}{\frac{\partial}{\partial r}\log f_1(R^-(y,\theta),\theta)}$ is a non-increasing function in y. Hence the integrand of $yQ'(y)$ is a non-increasing function, meaning that $yQ'(y)$ is non-increasing as desired. □

Theorem 3.5 (Convergence of Polar Slice Sampler)**.** *Consider the polar factorization $\pi(x) = f_0(x)f_1(x)$ in $\mathbb{R}^d$ with $f_0(x) = |x|^{-(d-1)}, f_1(x) = |x|^{(d-1)}\pi(x)$. Suppose $\pi(x)$ is log-concave. Then for any initial value x with $\frac{f_1(x)}{\sup_{w\in\mathbb{R}^d} f_1(w)}$ and $Y = A(f_1)$ the polar slice sampler converges. That is*

$$\forall \epsilon > 0,\ \exists N_Y\ s.t. \forall n \geq N_Y \quad ||\mathbb{P}(X_n \in \cdot | X_0 = x) - \nu_\pi(\cdot)|| < \epsilon$$

Proof. From Lemma 3.4 we know that $yQ'(y)$ is non-increasing for $y \leq A(f_1)$. It follows immediately by Proposition 2.7 that the polar slice sampler converges. □

Corollary 3.6 (Polar Slice Sample for Spherically Symmetrical Distributions)**.** *Suppose target density $\pi(x)$ in $\mathbb{R}^d$ is spherically symmetrical (and log concave). Then 3.5 holds with $Y = A(f_1) = 1$*

This is immediate because norm is spherically symmetric as well, hence f_1 is spherically symmetric. Then $M(f_1,\theta)$ is constant for every θ, i.e. every ray from origin

has the same maximum value so that $A(f_1) = 1$. From previous observations, this means that the polar slice sampler will converge very quickly for spherically symmetrical distributions.

4 Examples

In this section we will now look at a couple simulated examples using the polar slice sampler. These will be used to compare against the uniform slice sampler to see if it does indeed perform better in higher dimensions with respect to computational efficiency, and in what cases they perform best.

4.1 A simple symmetrical example

We start by running the polar slice sampler on $e^{-|x|}$ for different dimensions. $n = 1000$ time-steps are used. Figure 4.1 shows the norm of the MCMC chain as a stochastic process and the autocorrelation between time-steps for various dimensions.

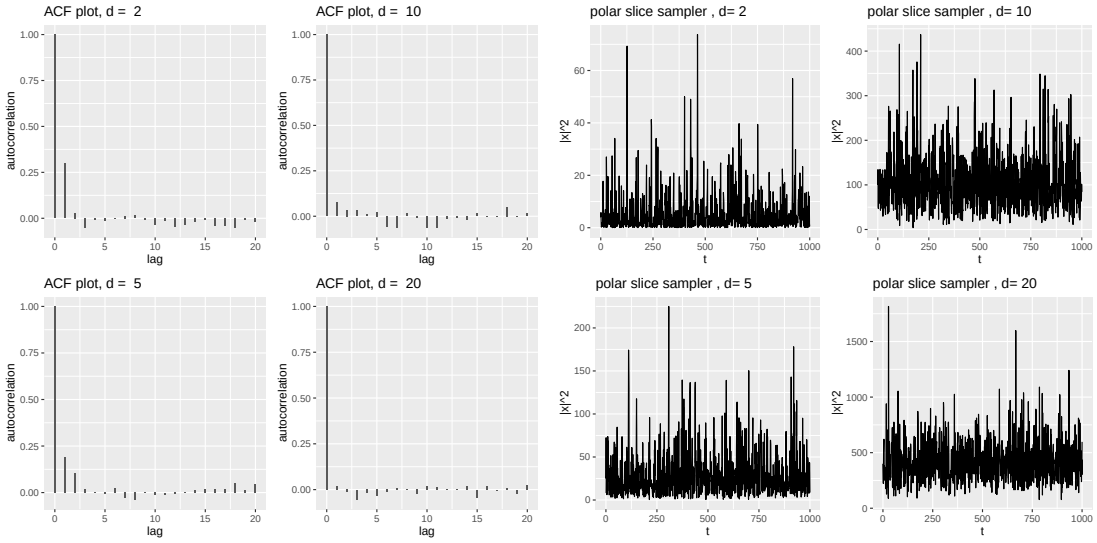

Figure 4.1: ACF (left) and Norm process (right) of polar slice sampler for $\pi(x) = e^{-|x|}$

Here, we can see that the autocorrelation is low for all dimensions. Though the plots use $n = 1000$ step chain, we get similar results for larger values of n. We can also see in Figure 4.2 that the number of rejected samples increases linearly with dimension, and is not very high. This preserves efficiency of the algorithm in higher dimensions. Of course, since the number of accepted samples is fixed at 1000, the rejection ratio increases rather quickly. In fact, here we used a rather naive (loose) choice of r^*, chosen through a numerical computation (using a 'trick' with δ being a free variable we choose).

$$y \leq \delta^{-(d-1)}(\delta|x|)^{d-1}e^{-|x|} \leq \delta^{-(d-1)}e^{(d-1)\delta|x|}e^{-|x|} = \delta^{-(d-1)}e^{((d-1)\delta-1)|x|}$$
$$\implies |x| \geq \frac{\log y\delta^{d-1}}{(d-1)\delta - 1} = r^*$$

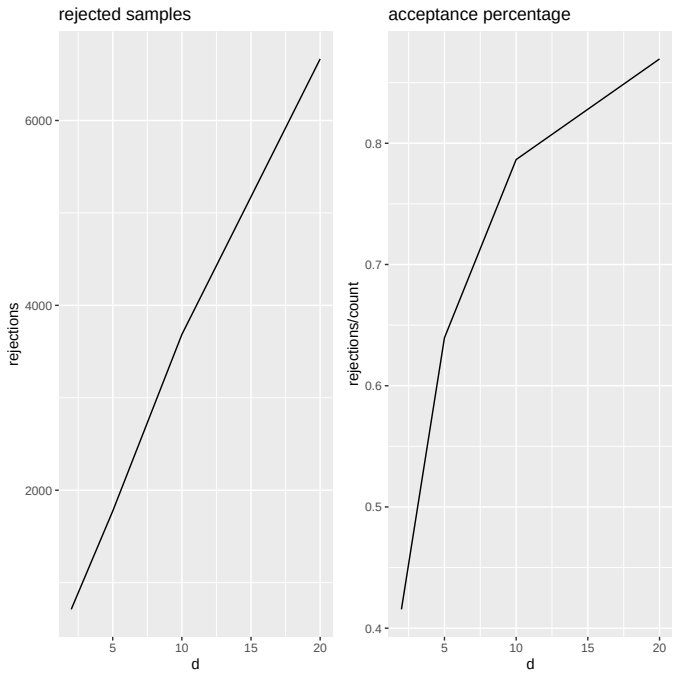

Figure 4.2: Number of Rejections (left) and acceptance ratio (right) during running of polar slice sampler for $\pi(x) = e^{-|x|}$

We need $(d-1)\delta - 1 < 0$ for the RHS to converge to 0. It turns out that $\delta = \frac{1}{d^2}$ gives the tightest bound of $|x|^{d-1}e^{-|x|}$. In this simple example, we could of actually found the roots of $f_1(x) - y$ and used that as a tighter r^*, which would decrease number of rejections. From a rejection sampling lens, the ball of radius r^* is our scaled 'proposal density' $g(x)$ scaled by factor K. As d increases, r^* rapidly increases, and so the volume in the ball increases. This means that our theoretical K value increases rapidly (since the proposal density always integrates to 1). This is evidenced by our rejection ratio increasing to 1. We proceed to compare polar slice sampling with the uniform slice sampler in Figure 4.3. We immediately see in , even at lower dimensions, that autocorrelation spikes, meaning that the uniform slice sampler is not going through the entire support very well.

In our toy example, $e^{-|x|}$ is spherically symmetrical since it is a function of the norm. Each slice at y will partition the points $\pi(x) \geq y$ as being inside the ball of radius y. Intuitively, uniform slice sampler is poor because in high dimensions, a large proportion of points will be close to the boundary of the ball of radius r (which is the value $|x|$ so that $f(x) = y$). For example the ratio of points within a ball of ratio $\frac{r}{2}$ and ball of ratio r is $\frac{1}{2^d}$. So in high dimensions, uniformly sampling in this ball drastically oversamples points away from the mode, i.e. the origin (from below image there are no the entire range $(-r, r)$, leading to poor efficiency and convergence results. On the other hand, the polar slice sampler will sample a radius uniformly within the ball, and so each distance from origin has equal chance of

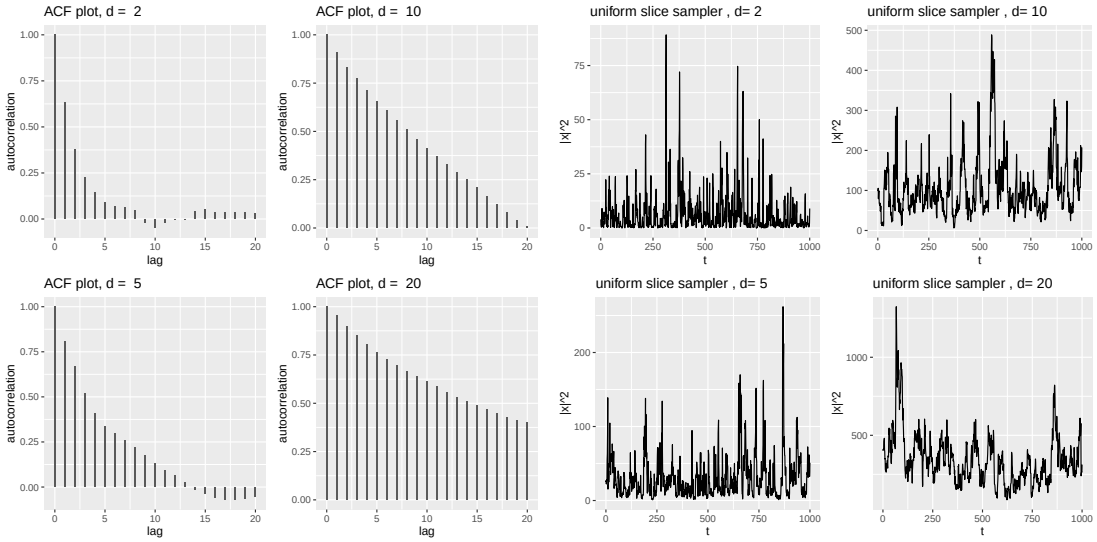

Figure 4.3: ACF (left) and Norm process (right) of uniform slice sampler for $\pi(x) = e^{-|x|}$

being chosen (in polar sampling, we actually consider $f_1(x) = |x|^{d-1}\pi(x)$, which is different than $\pi(x)$ but the same idea holds). This means the chain will mix faster leading to better convergence properties. However, to get a complete comparison, we must also account for the CPU time, measured as the time to run the process between polar and uniform slice samplers. In Table 4.4 we take the product of autocorrelation time and CPU time as a metric to evaluate algorithmic efficiency. Using this metric, polar slice sampler is far more efficient because the benefit in reduced autocorrelation exceeds the longer runtime (which is due to rejections).

act.p	act.u	cpu.p	cpu.u	ac.u	ac.p
1.44	4.19	0.37	0.28	1.17	0.53
1.19	4.36	0.44	0.24	1.06	0.52
0.99	9.17	0.68	0.38	3.51	0.67
0.94	19.07	1.38	0.71	13.58	1.29

Figure 4.4: Comparison of CPU and autocorrelation time for $\pi(x) = e^{-|x|}$

4.2 An asymmetrical Example

Next, we will consider another example, with the function $\pi_2(x) = e^{-\sum_{i=1}^{d} ix_i^2}$. In this example (similar to previous), we have the r^* value

$$y \leq |x|^{d-1} e^{-\sum_{i=1}^{d} ix_i^2} \leq |x|^{d-1} e^{-|x|^2} \leq \delta^{-\frac{d-1}{2}} e^{(\frac{d-1}{2}\delta - 1)|x|^2} \implies |x| \geq \sqrt{\frac{\log y \delta^{\frac{d-1}{2}}}{\frac{d-1}{2}\delta - 1}} = r^*$$

Again, we want $\frac{d-1}{2}\delta - 1 < 0$ for convergence, and we choose $\delta = 2d^{-\frac{3}{2}}$. In Figure 4.5 and 4.6 we again display the ACF, norm process and rejection ratios.

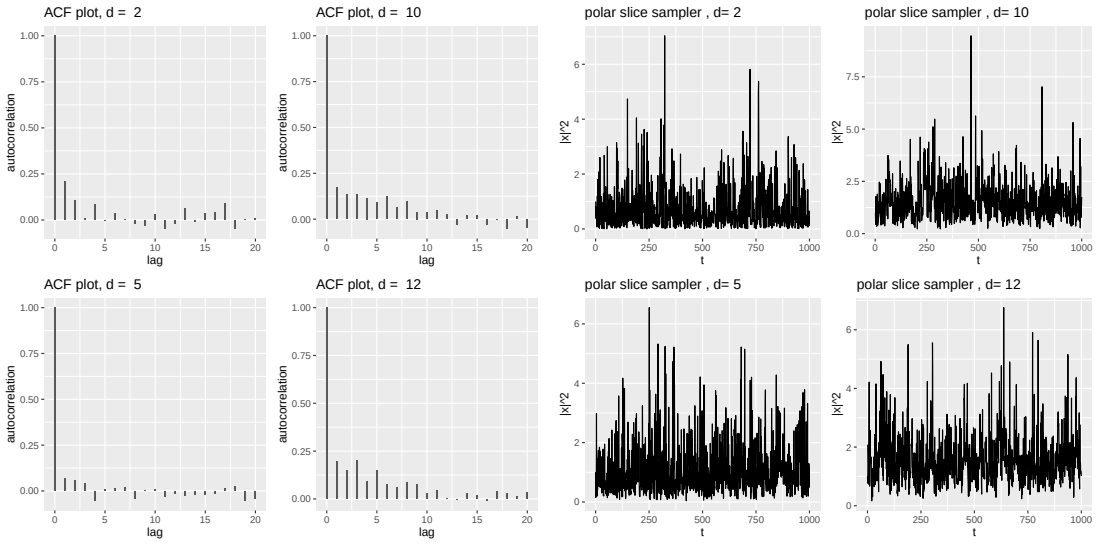

Figure 4.5: ACF (left) and Norm process (right) of polar slice sampler for $\pi_2(x) = e^{-\sum_{i=1}^{d} ix_i^2}$

Compared to the previous example, the number of rejections has drastically increased for all d, and increases exponentially as a function of d. The rejection percentage immediately shoots up to around 1. In fact, for $d = 15$ the algorithm failed to run and got stuck at around 61 million rejections and 150 iterations (for a single run). Intuitively, this is because π_2 is not spherically symmetrical and our chosen r^* value is dependent on the norm of x (which of course is spherically symmetrical). This means that for large choices of y, r^* will increase (as a function of y). However, for large $d, |x|$, we have $\sum_{i=1}^{d} ix_i^2 > |x|^2 \implies e^{-\sum_{i=1}^{d} ix_i^2} << e^{-|x|^2}$. Thus most values of $r \sim Uniform(0, r^*)$ will be too large and not satisfy constraint $|x|^{d-1}\pi_2(x) \geq y$. For a visualization, consider π_2 as having 'elliptical' cross-sections of the form $\sum_{i=1}^{m} ix_i^2 = -\log \frac{y}{(r^*)^{d-1}}$. Taking 2 dimensional cross-sections for $d = 20$, since each x_i is independent, we can model the (1,20) marginal as proportional to the below ellipse ($r^* = |x_{20}|$)

$$x_1^2 + 20x_{20}^2 = -\log k \implies x_1 = \pm\sqrt{-\log y - 20x_{20}^2}$$

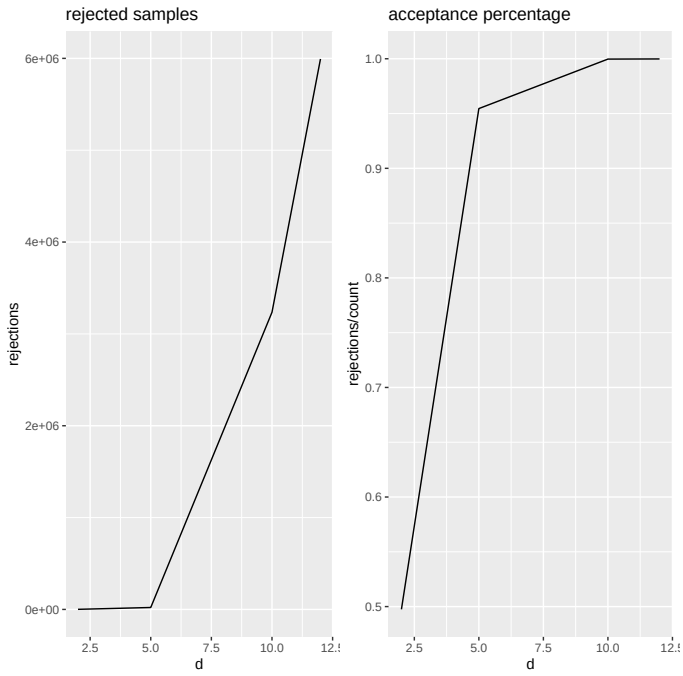

Figure 4.6: Number of Rejections (left) and acceptance ratio (right) during running of polar slice sampler for $\pi_2(x) = e^{-\sum_{i=1}^{d} ix_i^2}$

Here, we use $\frac{y}{(r^*)^{d-1}} = 10^{-10}$, which is reasonable for large y and corresponding r^*. The plot is in Figure 4.7a. We are examining the function π_2, but $\pi_2|x|^{d-1}$ will also be elliptical since $|x|^{d-1}$ is itself spherical. From these cross-sections, we can see that the norm is maximized on the axes (i.e. when $x_{20} = 0$) for both the ellipse and circle. Extrapolating to the function π_2, the norm will be maximized when $x_i = 0$ for $i = 2, 3, ..., d$ for π_2 but the norm value is actually the same for both 'shapes'. In our first example, any radius within the ball radius r^* satisfied $|x|^{d-1}\pi(x) \geq y$, but for this example many choices for r will not satisfy $|x|^{d-1}\pi_2(x) \geq y$, as the area of the ellipse around 3 times smaller (and in d dimensions, the ratio of volumes is taken to dth power). Note that $e^{-|x|^2}$ is a poor approximation of $e^{\sum_{i=1}^{d} ix_i^2}$, which we then approximate again (w.r.t. the norm) to determine r^*, multiplying the issue. In the language of rejection sampling, the constant K is very large in order to have $g(x) \geq |x|^{d-1}\pi_2(x)$ where g is the surface of a d dimensional ball (scaled to have density 1), leading to a small acceptance probability. For this reason, we also see higher autocorrelation in the Markov chains.

We can try using a tighter r^*, using an exact root finding method for $|x|^{d-1}\pi_2(x) - y = 0$, but the algorithm still fails to converge for high dimensional d, showing the issue is in the lack of spherical symmetry. Of course, computing $\sup\{|x| | |x|^{d-1}\pi_2(x) \geq y\}$ exactly will help, but there will still be far higher rejections than for $\pi(x)$. For visualization, and to see the polar sampler does work, we will show the $1, 3, 5, 10$

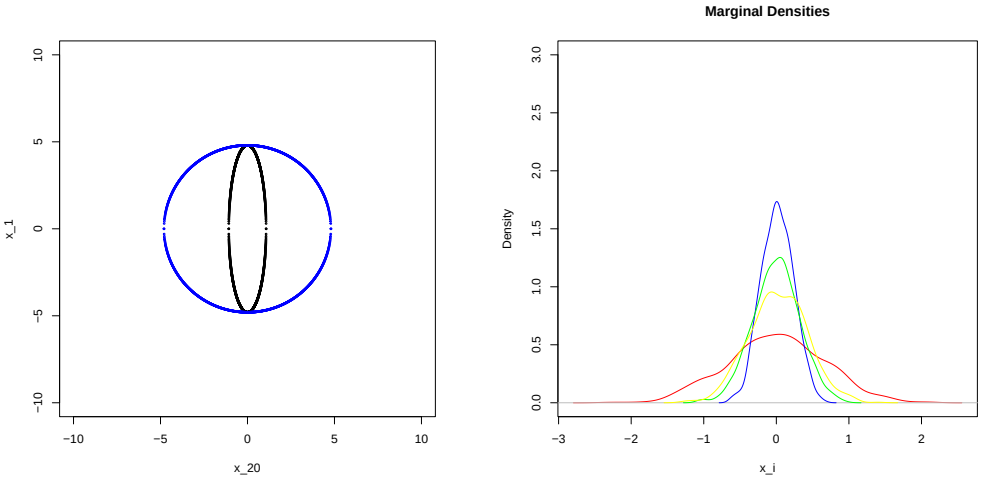

Figure 4.7: (left) Visualization of elliptical cross-sections for π_2. (right) Empirical marginal distributions of the resulting Markov chain

marginals of π_2 for $d = 10$ in 4.7b . We also again compare with the uniform slice sampler in Figure 4.8 and see that it also exhibits high autocorrelation and does a poor job of convergence.

4.3 Comparing CPU time for Varying levels of Symmetry

From our first 2 examples, it seems reasonable to look at what degree of symmetry leads to better results for the polar slice sampler as oppopsed to the uniform slice sampler w.r.t. out CPU time and ACT metrics. We consider the family of functions 'in-between' $\pi(x)$ and $\pi_2(x)$, denoted $\pi_j(x) = e^{-\sum_{i=1}^{d} i^j x_i^2}$. For $j = 0$ this is simply the function $\pi(x)$ from our first example and for $j = 1$ this is the function $\pi_2(x)$ from our second example. We choose $j = 0, 0.25, 0.5, 0.75, 1$ and compare the ACT and CPU times of the polar and uniform samplers for $d = 5, 10$. Note that the slice samplers will produce different results depending on the specific chain (randomly) sampled and so we run each sampler 6 times and average results for stability. Results are in Table 4.9 and 4.10.

For $d = 5$ the polar slice sampler still outperforms the uniform slice sampler in efficiency for all values of j, however when we move up to $d = 10$ the results start shifting. For $j < 0.5$ we still see polar slice sampler performing well but for $j \geq 0.5$ the CPU time increases rapidly, while the change in ACT is minimal, meaning that the the product of the two is greater for the polar slice sampler. This is directly caused by the larger number of rejections we see (as in example 2 Figure 4.6). Moreover, the ACT actually decreases for the uniform slice sampler as j increases to 1 since the relative non-symmetry will allow for better mixing in this case. This

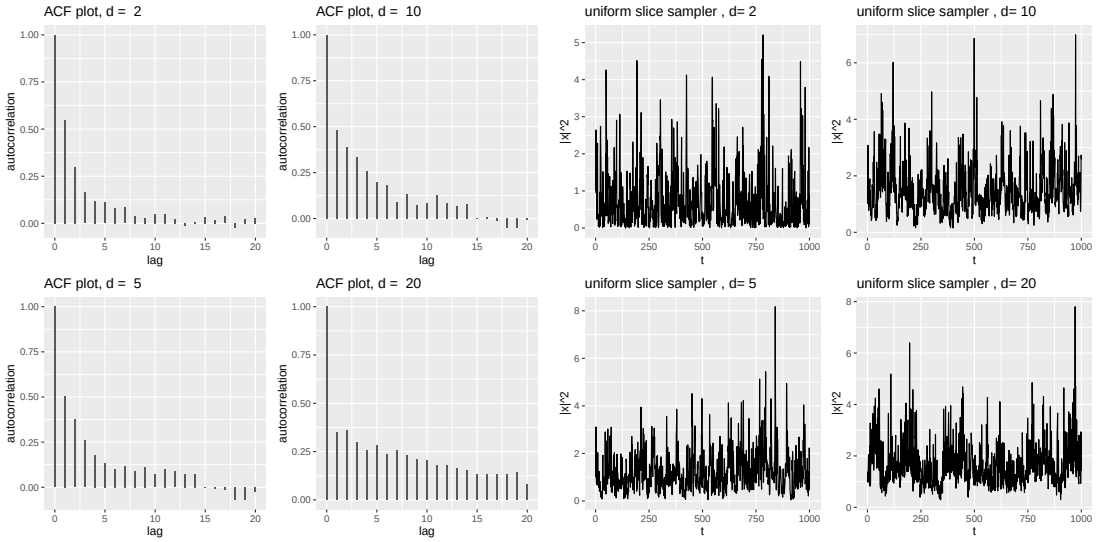

Figure 4.8: ACF (left) and Norm process (right) of uniform slice sampler for $\pi(x) = e^{-\sum_{i=1}^{d} i x_i^2}$

is because the norm, which we are measuring is more restricted on the ellipses. From this we can draw the conclusion that in very high dimensions, polar slice sampling in the given form is a poor algorithm for non-symmetrical distributions as the CPU time is too high. On the other hand, it performs quite well on symmetrical distributions and for medium dimensions.

	act.p	act.u	cpu.p	cpu.u	ac.u	ac.p
0	1.12	3.85	0.62	0.45	1.73	0.7
0.25	1.06	3.39	0.67	0.42	1.43	0.71
0.5	1.08	3.08	0.69	0.44	1.35	0.75
0.75	0.92	3.02	0.76	0.42	1.25	0.7
1	1	2.59	0.85	0.42	1.09	0.85

Figure 4.9: Comparison of CPU and autocorrelation time for $\pi_j(x) = e^{-\sum_{i=1}^{d} i^j x_i^2}$ for dimension $d = 5$

4.4 Example on subset of $\mathbb{R}^d$

For another example, we will use $\pi_3(x) = e^{-|x|}]\mathbf{1}_{\mathbf{x_1},\ldots,\mathbf{x_{\frac{d}{2}}} \geq \mathbf{0}}$. That is, the function $\pi(x)$ from our first example, restricted to the first $\frac{d}{2}$ components being positive. For simplicity, we will assume d to be even. Clearly, this is a non-symmetric function.

	act.p	act.u	cpu.p	cpu.u	ac.u	ac.p
0	1.1	5.85	1.01	0.68	3.96	1.11
0.25	1.07	5.15	1.11	0.66	3.41	1.18
0.5	1.09	4.82	7.76	0.66	3.2	9.2
0.75	1.28	4.2	6.88	0.68	2.85	8.7
1	2.03	3.72	17.51	0.66	2.47	37.15

Figure 4.10: Comparison of CPU and autocorrelation time for $\pi_j(x) = e^{-\sum_{i=1}^{d} i^j x_i^2}$ for dimension $d = 10$

We will use the same r^* function. Same plots as before will be in Figures 4.11 and 4.12.

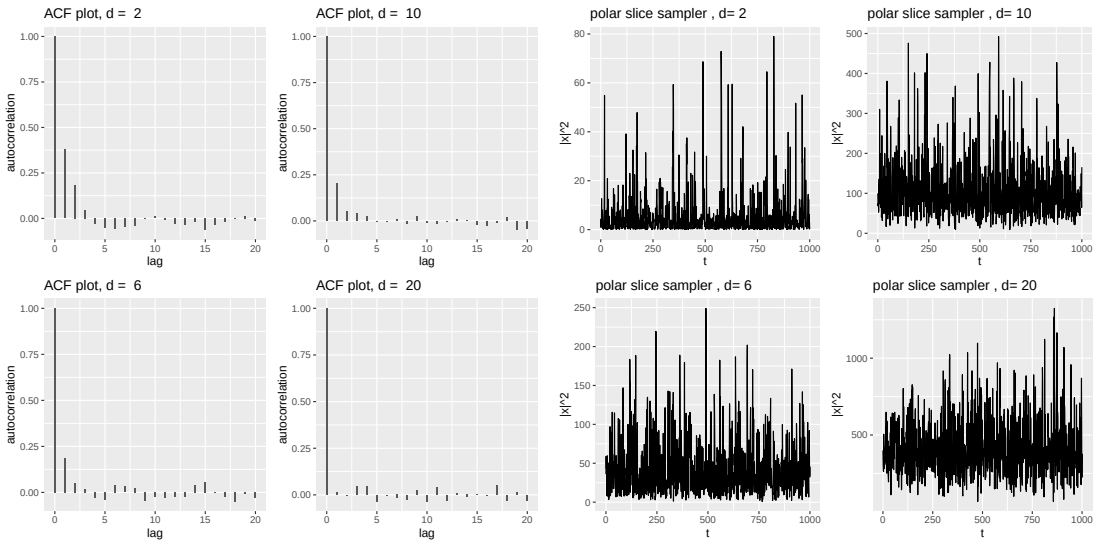

Figure 4.11: ACF (left) and Norm process (right) of polar slice sampler for $\pi_3(x) = e^{-|x|}]\mathbf{1}_{\mathbf{x_1},\ldots,\mathbf{x_{\frac{d}{2}}} \geq \mathbf{0}}$

The polar sampler still mixes very well, but we notice the number of rejected samples is significantly higher than the first example (which would be worse if increased chain length or dimension). In fact, the rate of increase is increasing so that for higher values of d, the computational power needed drastically increases. This would in turn increase the CPU time making the uniform slice sampler more efficient using our previous metrics. Of course, the chain still converges nicely. This shows it could be beneficial to sample the 'angle' in our polar-rejection sample not completely at random. In this case, it would be easy to simply ensure a positive θ_i value for each $i = 1, ..., \frac{1}{2}$. But oftentimes, it is not so easy to sample r, θ_i in a more efficient

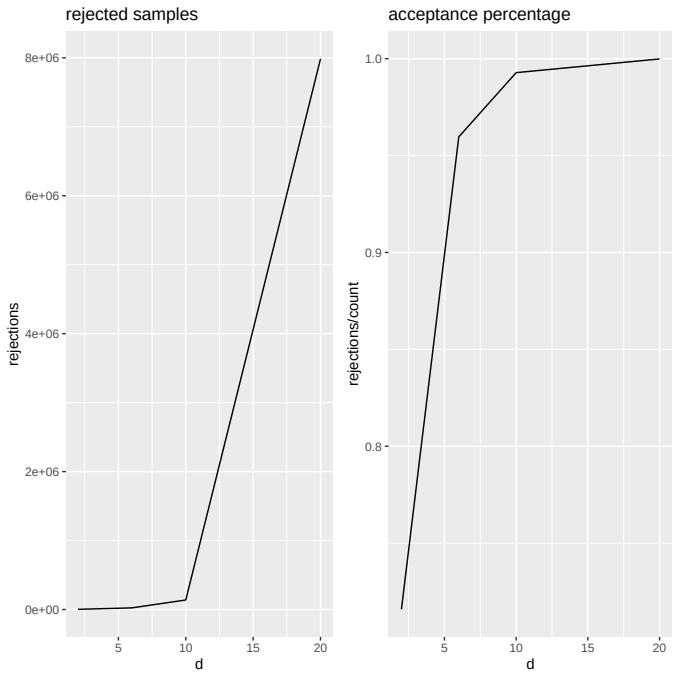

Figure 4.12: Number of Rejections (left) and acceptance ratio (right) during running of polar slice sampler for $\pi_3(x) = e^{-|x|}]\mathbf{1}_{\mathbf{x_1},\ldots,\mathbf{x_{\frac{d}{2}}} \geq \mathbf{0}}$

manner.