Statistical Computing for Scientists and Engineers

Homework 5

Jiale Shi

Nov/12/2018

1 Importance Sampling

Use importance sampling to approximate expectation $\mathbb{E}_p[f(x)]$ where

$$f(x) = 2\sin\left(\frac{\pi}{1.5}x\right), x \geq 0 \tag{1}$$

and target distribution is defined as

$$p(x) = x^{1.65-1}\exp\left(-x^2/2\right), x \geq 0 \tag{2}$$

write down your choice of sampling distribution. Plot sampling distribution and target distribution on a plot. Write down the algorithm of your implementation.

Solution: The target distribution

$$p(x) = x^{1.65-1}\exp\left(-x^2/2\right) = x^{0.65}\exp\left(-x^2/2\right), x \geq 0 \tag{3}$$

The shape of the target distribution is similar to the Gamma distribution Gamma(3,2.5) to some extent. Therefore, we choose Gamma distribution Gamma(3,2.5) as our proposed distribution.

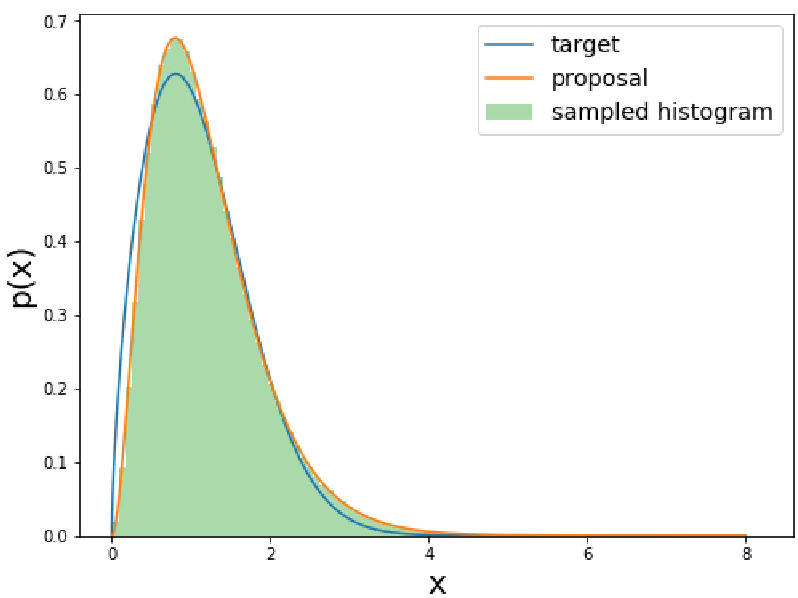

Figure 1: sampling distribution Gamma(3,2.5), target distribution

We samples for $N = 10^6$ times, collect samples(x), calculate weight$W$.

$$W(samples) = \frac{p(samples)}{Gamma(3, 2.5)} \tag{4}$$

$$E(f(x)) \approx \frac{1}{N}\sum_{i=1}^{N} W(samples)f(samples) \approx 0.7758554717760647 \tag{5}$$

2 Sequential Monte Carlo Sampler

Implement a sequential Monte Carlo sampler to draw samples from target distribution π, which is an equal mixture of 5 normal distributions in $\mathbb{R}^2$ with unit covariance and centers that are equally distributed on the circumference of a circle with diameter 40.

Initialize 3000 particles with points located at the center of the circle.

Using 100 bridging densities $\pi_0, \ldots, \pi_{100}$ where

$$\pi_k(x) \propto \pi(x)^{\alpha_k} \tag{6}$$

and $0 \leq \alpha_1 \leq ... \leq \alpha_n = 1$.

Use a normal random walk proposal with step size $\sqrt{6}$ to evolve particles

Measure degeneracy using effective sample size with a threshold of $N/2$ and resample accordingly.

To show the evolution of the points, including distribution of points for π_0, π_{50}, π_{100}. Additionally, generate an animation showing the points evolving from π_0 to π_n. Write down algorithm of your implementation.

Solution: Target distribution π, is an equal mixture of 5 normal distributions in $\mathbb{R}^2$ with unit covariance and centers that are equally distributed on the circumference of a circle with diameter 40.

The five center points should be
$(x_1, y_1) = (20\cos(0), 20\sin(0))$;
$(x_2, y_2) = (20\cos(\frac{2\pi}{5}), 20\sin(\frac{2\pi}{5}))$;
$(x_3, y_3) = (20\cos(\frac{4\pi}{5}), 20\sin(\frac{4\pi}{5}))$;
$(x_4, y_4) = (20\cos(\frac{6\pi}{5}), 20\sin(\frac{6\pi}{5}))$;
$(x_5, y_5) = (20\cos(\frac{8\pi}{5}), 20\sin(\frac{8\pi}{5}))$

The final target distribution $\pi(x, y)$ should be

$$\begin{aligned}\pi(x,y) =& \exp\{-[(\frac{(x-x_1)^2}{2}) + (\frac{(y-y_1)^2}{2})]\} \\ &+ \exp\{-[(\frac{(x-x_2)^2}{2}) + (\frac{(y-y_2)^2}{2})]\} \\ &+ \exp\{-[(\frac{(x-x_3)^2}{2}) + (\frac{(y-y_3)^2}{2})]\} \\ &+ \exp\{-[(\frac{(x-x_4)^2}{2}) + (\frac{(y-y_4)^2}{2})]\} \\ &+ \exp\{-[(\frac{(x-x_5)^2}{2}) + (\frac{(y-y_5)^2}{2})]\}\end{aligned} \tag{7}$$

All the 3000 particles start from the original point $(0, 0)$. For $K = [1, 2, 3, ..., 100]$, $a_K = [0.01, 0.02, 0.03, .., 1]$ the target distribution $\pi(x, y)_K = \pi(x, y)^{a_K}$. The proposal distribution is a 2D normal distribution with step size is $\sqrt{6}$. The update follows the MH algorithm. The detail is in the following figure.

The distribution of points for π_0, π_{50},π_{100}.

Algorithm 1: The SMC sampler for γ

Sample x_0^i from γ_0 and set weights $w_0^i = 1/N$, for $i = 1, \dots, N$
for $k = 1$ **to** K **do**
    Compute $\widetilde{w}_k^i = w_{k-1}^i \frac{\gamma_k(x_{k-1}^i)}{\gamma_{k-1}(x_{k-1}^i)}$
    Compute the normalized weights $w_k^i = \frac{\widetilde{w}_k^i}{\sum_{i=1}^N \widetilde{w}_k^i}$
    if *ESS too low* **then**
        Resample, e.g. by sampling a_k^i from the categorical distribution $\mathcal{C}(\{w_k^j\}_{j=1}^N)$, for $i = 1, \dots, N$
        Set $w_k^i = 1/N$ for $i = 1, \dots, N$
    else
        Set $a_k^i = i$ for $i = 1, \dots, N$
    end
    Set $x_k^i \leftarrow x_{k-1}^{a_k^i}$ for $i = 1, \dots, N$
    for $i = 1, \dots, N$ **do**
        Sample x' from a proposal $r(x' \mid x_k^i)$, e.g., $\mathcal{N}(x' \mid x_k^i, \Sigma)$
        Compute the acceptance rate $\alpha = \min(1, \frac{\gamma_k(x')}{\gamma_k(x_k^i)} \frac{r(x_k \mid x')}{r(x' \mid x_k)})$
        Sample $d \sim \mathcal{U}(0, 1)$
        if $d < \alpha$ **then**
            Accept and set $x_k^i \leftarrow x'$
        else
            Reject and discard x'
        end
    end
end

Figure 2: SMC Algorithm

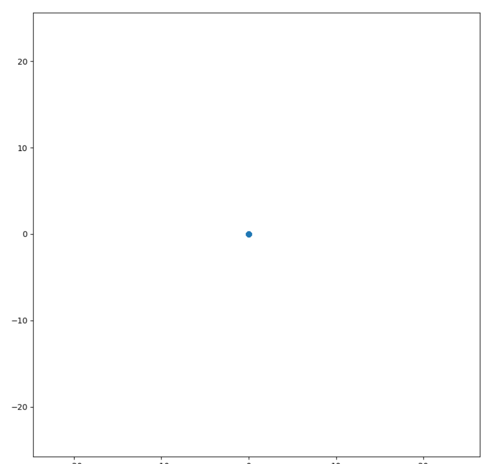

Figure 3: π_0 start from the center of the circle

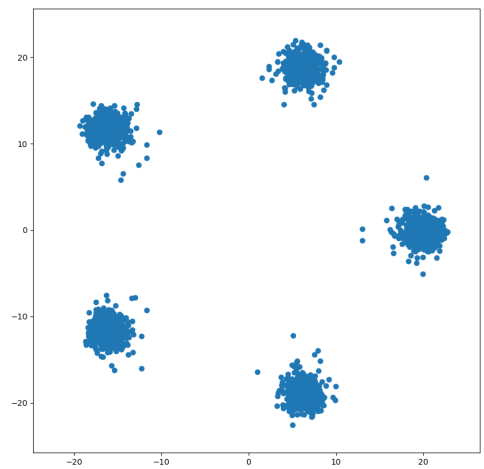

Figure 4: π_{50}

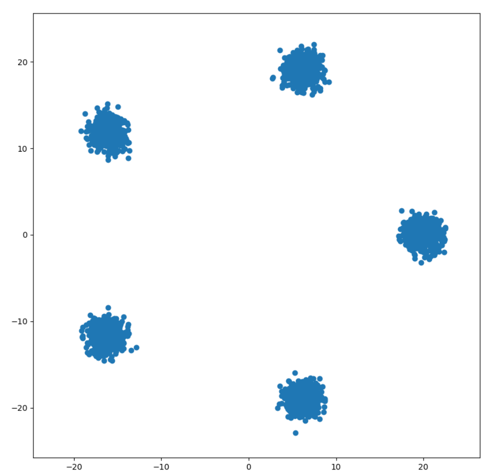

Figure 5: π_{100}

3 Hamiltonian Monte Carlo (HMC)

HMC introduces an momentum variable q, and uses Hamiltonian dynamics to generate samples.

The potential energy $U(x) = -\log p(x)$, the kinetic energy $K(q) = -\log p(q)$. $p(x)$ is the target density and $p(q)$ is the proposal density for q. The summation $H(x,q) = U(x)+K(q)$. If we can generate samples $\propto \exp(-H(x,q)) = p(x)p(q)$, the resulting x samples will be distributed according to the target one.

To generate new candidate samples based on the Hamilton's equation of motion:

$$\begin{aligned}\frac{\partial x_i}{\partial t} &= \quad \frac{\partial H}{\partial q_i} = \frac{\partial K}{\partial q_i} \\ \frac{\partial q_i}{\partial t} &= \quad -\frac{\partial H}{\partial x_i} = -\frac{\partial U}{\partial x_i}\end{aligned} \tag{8}$$

For numerical implementation, Hamilton's equations must be approximated by non-continual time, using small step ϵ. It starts with half step update for the momentum variable, and then do a full step for x using the update momentum, and then do the other half step for momentum.

$$\begin{aligned}&q_i(t+\epsilon/2) = q_i(t) - (\epsilon/2)\frac{\partial U}{\partial x_i(t)} \\ &x_i(t+\epsilon) = x_i(t) + \epsilon\frac{\partial H}{\partial q_i} \qquad \text{where} \frac{\partial H}{\partial q_i} = q_i(t+\epsilon/2) \\ &q_i(t+\epsilon) = q_i(t+\epsilon/2) - (\epsilon/2)\frac{\partial U}{\partial x_i(t+\epsilon)}\end{aligned} \tag{9}$$

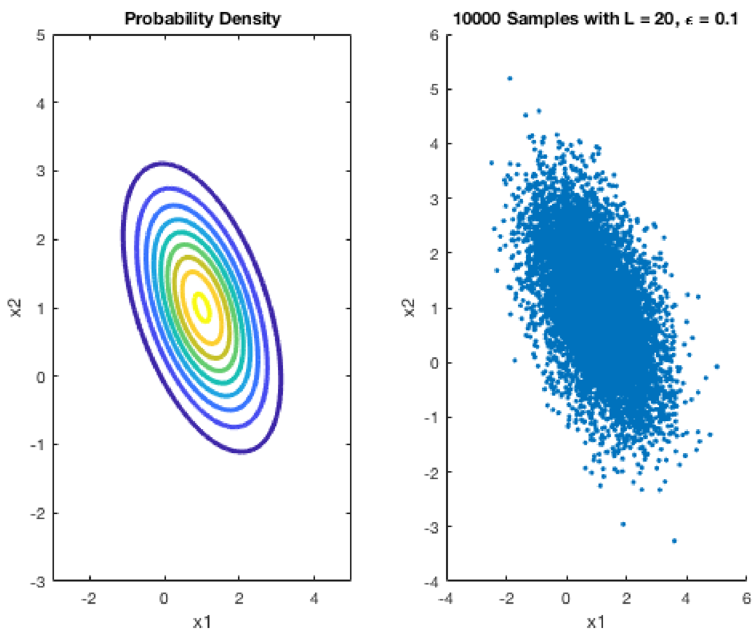

Figure 6: target distribution, sampled distribution

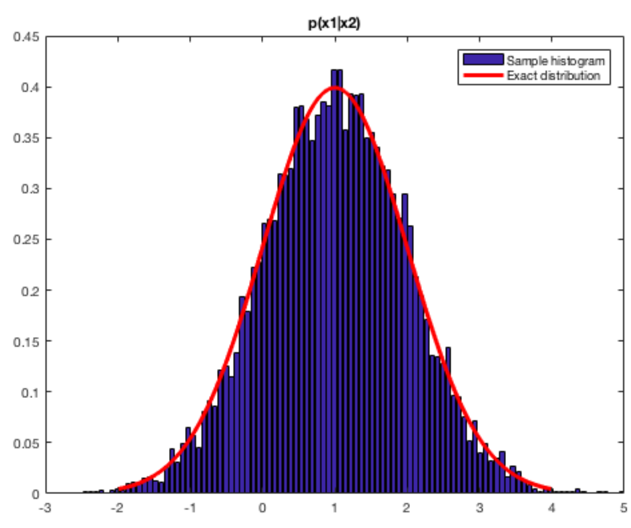

Figure 7: $p(x_1|x_2)$

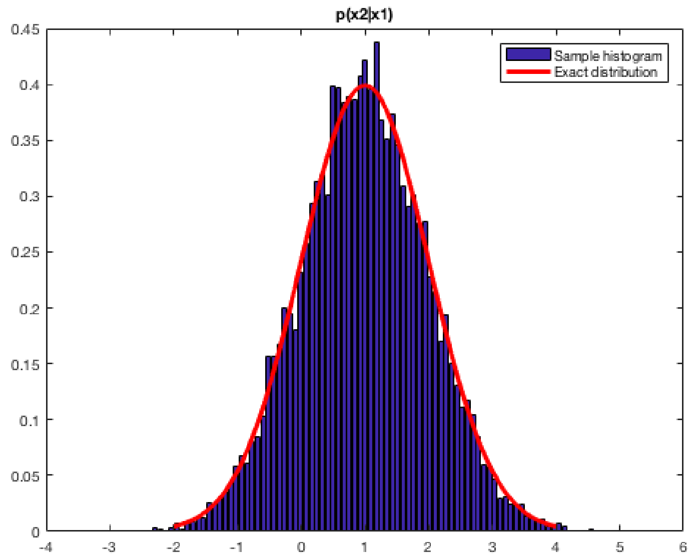

Figure 8: $p(x_2|x_1)$

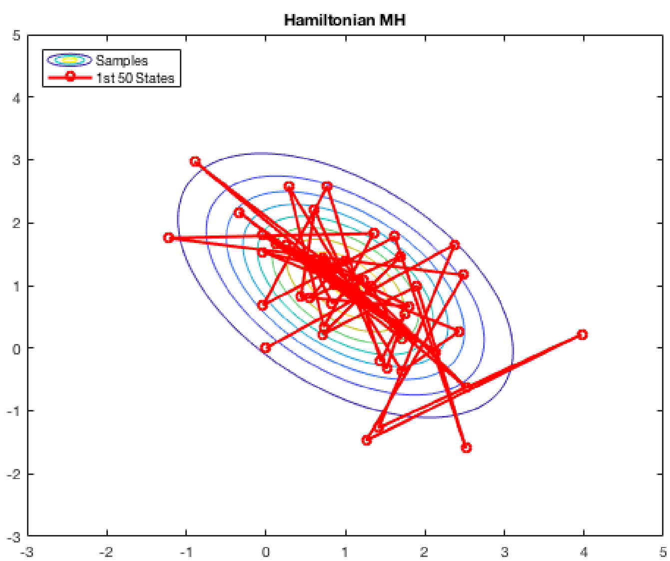

Figure 9: The first 50 steps

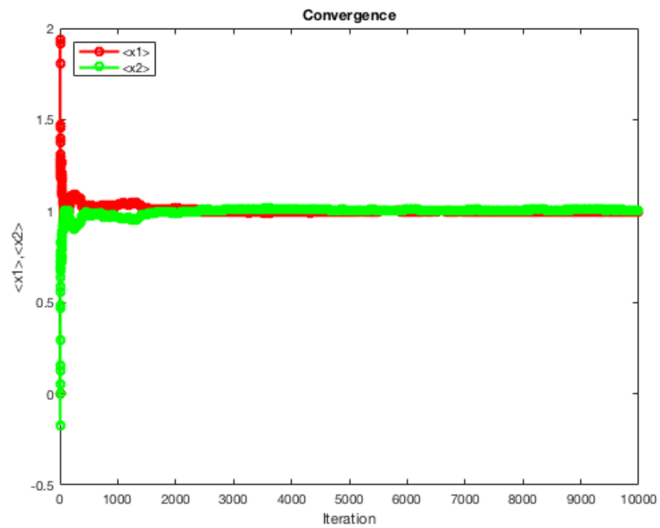

Figure 10: Convergence of $< x_1 >, < x_2 >$

4 Sequential Importance Sampling for Solving Integral Equations

Devise and implement a (sequential) importance sampling Monte Carlo scheme to solve the following integral equation for $f(x)$ (defined in [-1,1]) at $x = 0$:

$$f(x) = x + \frac{1}{2}\int_{-1}^{1}(t-x)f(t)dt \tag{10}$$

and compare results with the actual solution:

$$f(x) = \frac{3}{4}x + \frac{1}{4} \tag{11}$$

Consider uniform distribution $U[-1-\alpha, 1+\alpha]$ as the transition kernel and with the appropriate stopping rule (similarly to the example considered in class in solving $Ax = b$). Comment on the performance for $[\alpha = 0.001, 0.1, 1, 1.5, 2]$

Solution:

According to the actual solution, $f(x) = \frac{3}{4}x + \frac{1}{4}$. The true value $f(0) = 0.25$. The integral equation can be transformed into solving

$$f(x) = x + \sum_{n=1}^{\infty}\int_{-1}^{1}(\prod_{k=1}^{n}(t_k - t_{k-1}))t_k dt_{1:n} \tag{12}$$

It involves an infinite sum of integrals of increasing dimension, which can be solved by SIS. Consider uniform distribution $U[-1-\alpha, 1+\alpha]$ as the transition kernel.

$P_d = \alpha$

$M = \dfrac{1}{(2 + 2 \cdot \alpha)}$

simulate a path using Markov chain. Start from $x^{(i)} = 0$, then generate sample $t_1^{(i)} \sim M(t_1^{(i)}, t)$, $t_2^{(i)} \sim M(t_2^{(i)}, t)$, until $t_{k+1}^{(i)}$ reaches s, the cemetery state.

Calculate the associated weight

$$W^{(i)}(x, t_1, .., t_k) = \begin{cases} \frac{0.5((t_1 - x))}{M} (\prod_{k=1}^{n} \frac{0.5(t_k - t_{k-1})}{M}) \frac{t_k}{P_d} & \text{for} \quad k > 0 \\ \frac{0.5((t_1 - x))}{M} \frac{t_k}{P_d} & \text{for} \quad k = 0 \end{cases}$$

$f(x) = y^{(N)} = \dfrac{y_0 + \sum_i^N W^{(i)}}{N}$ where $\quad y_0 = 0$

For $[\alpha = 0.001, 0.1, 1, 1.5, 2]$, the performance is:

$$\begin{aligned} &\alpha = 0.001, && f(0) = 0.2764 \text{slow} \\ &\alpha = 0.01, && f(0) = 0.2527 \\ &\alpha = 0.015, && f(0) = 0.2501 \\ &\alpha = 0.02, && f(0) = 0.2410 \\ &\alpha = 0.04, && f(0) = 0.2368 \\ &\alpha = 0.1, && f(0) = 0.2305 \\ &\alpha = 1, && f(0) = 0.1205 \\ &\alpha = 1.5, && f(0) = 0.0982 \\ &\alpha = 2, && f(0) = 0.0820 \text{fast} \end{aligned} \tag{13}$$

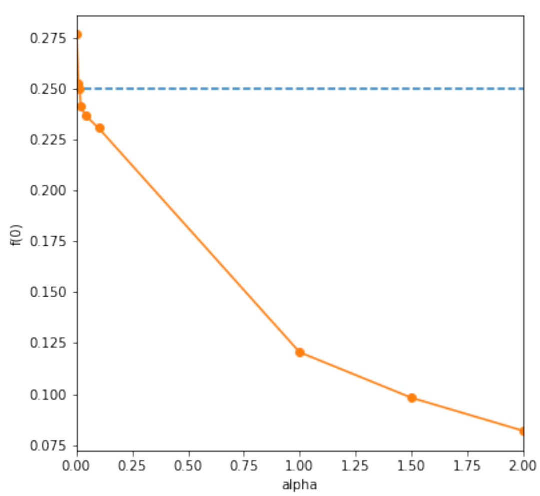

Figure 11: Convergence of $< x_1 >, < x_2 >$

When α becomes larger and larger, the speed becomes faster and faster. From $\alpha = [0.001, 0.01, 0.015, 0.02, 0.04, 0.1, 1, 1.5, 2]$, when α becomes larger and larger, $f(0)$ initially a little bigger than 0.25, and then $f(0)$ decreases. At $\alpha = 0.015$, $f(0)$ is closest to the true value 0.25.