1 Simple Metropolis: Normal Precision – Gamma.

We know that the prior $\pi(\theta) \propto \theta^{\alpha_0 - 1} e^{-\beta_0 \theta}$, where $\alpha_0 = \frac{1}{2}$ and $\beta_0 = 1$. We also know that the likelihood $f(x|\theta) \propto \theta^{\frac{1}{2}} e^{-\frac{1}{2}\theta x^2}$. Then the posterior

$$\pi(\theta|x) \propto \pi(\theta) f(x|\theta) = \theta^{\alpha_0 - \frac{1}{2}} e^{-\theta\left(\beta_0 + \frac{1}{2}x^2\right)} = e^{-\theta\left(1 + \frac{1}{2}x^2\right)}.$$

We first derive the acceptance ratio for the Metropolis algorithm.

$$\begin{aligned}\rho(\theta_n, \theta') &= \min\left\{1, \frac{\pi(\theta')}{\pi(\theta_n)} \frac{q(\theta_n|\theta')}{q(\theta'|\theta_n)}\right\} \\ &= \min\left\{1, \frac{e^{-\theta'\left(1+\frac{1}{2}x^2\right)}}{e^{-\theta_n\left(1+\frac{1}{2}x^2\right)}} \frac{q(\theta_n|\theta')}{q(\theta'|\theta_n)}\right\} = \min\left\{1, \frac{e^{-3\theta'}}{e^{-3\theta_n}} \frac{q(\theta_n|\theta')}{q(\theta'|\theta_n)}\right\},\end{aligned}$$

where $q(\theta_n|\theta')$ and $q(\theta'|\theta_n)$ are determined based on the proposal distribution ($\mathcal{G}a(\alpha, \beta)$). We then apply the Metropolis algorithm to approximate the posterior distribution with the following steps.

- **Step 1**. Start with an arbitrary θ_0.
- **Step 2**. At stage n, generate proposal θ' from $\mathcal{G}a(\alpha, \beta)$ for the chosen α and β.
- **Step 3**. Set

  $\theta_{n+1} = \theta'$ with probability $\rho(\theta_n, \theta')$; and

  $\theta_{n+1} = \theta_n$ with probability $1 - \rho(\theta_n, \theta')$.
- **Step 4**. Increase n by 1 and go to **Step 2**.

We consider the following two examples of choosing α and β (any reasonable choices of α and β are accepted).

(a) For independent Metropolis, we can choose $\alpha = 1$ and $\beta = 3$. Then the acceptance ratio can be found as

$$\rho(\theta_n, \theta') = \min\left\{1, \frac{e^{-3\theta'}}{e^{-3\theta_n}} \frac{e^{-3\theta_n}}{e^{-3\theta'}}\right\}.$$

Notice that we get the acceptance ratio equals 1 for all generated θ'. This is due to the fact that our proposal distribution is identical to the posterior distribution.

(b) Or if we want to use the information of θ_n, we may consider setting $\alpha = 1$ and $\beta = \frac{1}{\theta_n}$. This means that

$$[\theta_n|\theta'] \sim \mathcal{G}a(1, \frac{1}{\theta'}), \text{ and } [\theta'|\theta_n] \sim \mathcal{G}a(1, \frac{1}{\theta_n}).$$

The acceptance ratio can be found as

$$\rho(\theta_n, \theta') = \min\left\{1, \frac{e^{-3\theta'}}{e^{-3\theta_n}} \frac{e^{-\frac{\theta_n}{\theta'}}}{e^{-\frac{\theta'}{\theta_n}}}\right\}.$$

We implement the Metropolis algorithm with the two proposal distributions. The posterior distribution with the proposal distribution discussed in (a) is shown in Figure 1, and the posterior distribution with the proposal distribution disucssed in (b) is shown in Figure 2.

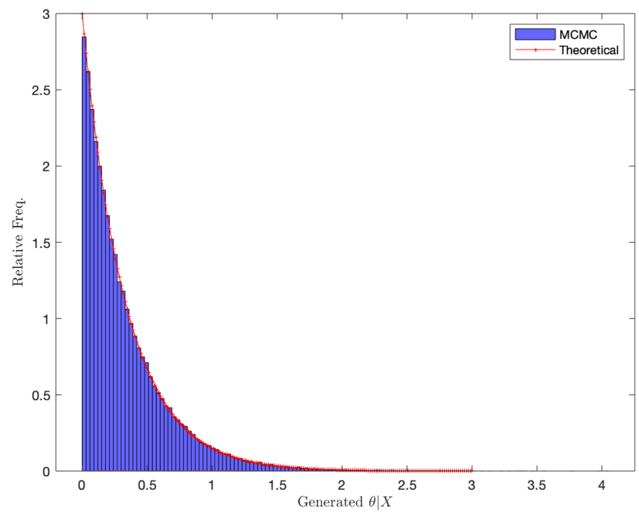

Figure 1: Histogram of generated θ

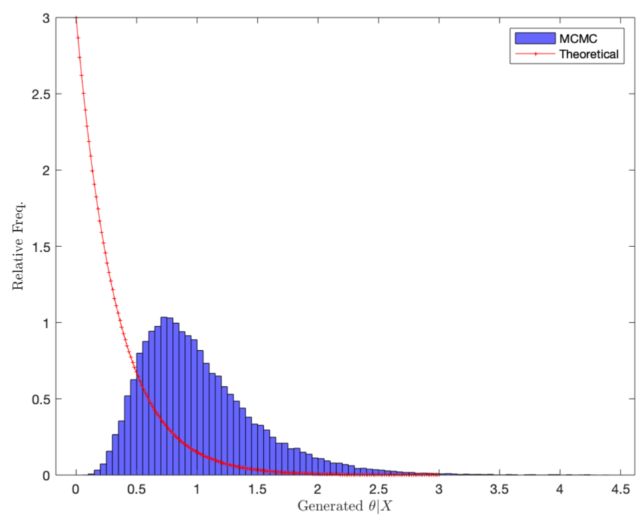

Figure 2: Histogram of generated θ

We see that the Bayes estimator based on the proposal distribution from (a) is 0.3357, and the Bayes estimator based on the proposal distribution from (b) is 1.0146.

We could also easily find that the percentage of the generated θ being accepted is 1 in (a) and only 0.3015 in (b). This also shows that the proposal distribution in (a) is more preferred.

2 Normal-Cauchy by Gibbs.

(a) As $\pi(\bar{y}|\theta,\sigma^2) \propto \exp\left(-\frac{(\bar{y}-\theta)^2}{2\frac{\sigma^2}{n}}\right)$, $\pi(\theta|\lambda) \propto \sqrt{\frac{\lambda}{\tau^2}}\exp\left(-\frac{(\theta-\mu)^2}{2\frac{\tau^2}{\lambda}}\right)$ and $\pi(\lambda) \propto \lambda^{\alpha-1}e^{-\beta\lambda}$ where $\alpha = \frac{1}{2}$ and $\beta = \frac{1}{2}$. The product of the likelihood and the prior is proportional to

$$\exp\left(-\frac{(\bar{y}-\theta)^2}{2\frac{\sigma^2}{n}}\right)\sqrt{\frac{\lambda}{\tau^2}}\exp\left(-\frac{(\theta-\mu)^2}{2\frac{\tau^2}{\lambda}}\right)\lambda^{\alpha-1}\exp(-\beta\lambda).$$

We find the full conditional for $[\theta|\bar{y},\lambda]$ as

$$\begin{aligned}
\pi(\theta|\bar{y},\lambda) &\propto \exp\left(-\frac{(\bar{y}-\theta)^2}{2\frac{\sigma^2}{n}}\right)\exp\left(-\frac{(\theta-\mu)^2}{2\frac{\tau^2}{\lambda}}\right)\\
&= \exp\left(-\frac{(n\tau^2+\lambda\sigma^2)\theta^2 - 2(\bar{y}n\tau^2+\lambda\sigma^2\mu)\theta + n\tau^2\bar{y}^2+\lambda\sigma^2\mu^2}{2\sigma^2\tau^2}\right)\\
&= \exp\left(-\frac{\theta^2 - 2\frac{\bar{y}n\tau^2+\lambda\sigma^2\mu}{n\tau^2+\lambda\sigma^2}\theta + \frac{n\tau^2\bar{y}^2+\lambda\sigma^2\mu^2}{n\tau^2+\lambda\sigma^2}}{2\frac{\sigma^2\tau^2}{n\tau^2+\lambda\sigma^2}}\right)\\
&\propto \exp\left(-\frac{\theta^2 - 2\frac{\bar{y}n\tau^2+\lambda\sigma^2\mu}{n\tau^2+\lambda\sigma^2}\theta + \left(\frac{\bar{y}n\tau^2+\lambda\sigma^2\mu}{n\tau^2+\lambda\sigma^2}\right)^2}{2\frac{\tau^2\sigma^2/n}{\tau^2+\lambda\sigma^2/n}}\right)\\
&= \exp\left(-\frac{\left(\theta - \left(\frac{\tau^2}{\tau^2+\lambda\sigma^2/n}\bar{y} + \frac{\lambda\sigma^2/n}{\tau^2+\lambda\sigma^2/n}\mu\right)\right)^2}{2\frac{\tau^2\sigma^2/n}{\tau^2+\lambda\sigma^2/n}}\right).
\end{aligned}$$

This means that $[\theta|\bar{y},\lambda] \sim \mathcal{N}\left(\frac{\tau^2}{\tau^2+\lambda\sigma^2/n}\bar{y} + \frac{\lambda\sigma^2/n}{\tau^2+\lambda\sigma^2/n}\mu, \frac{\tau^2\sigma^2/n}{\tau^2+\lambda\sigma^2/n}\right)$.

We find the full conditional for $[\lambda|\bar{y},\theta]$ as

$$\begin{aligned}
\pi(\lambda|\bar{y},\theta) &\propto \sqrt{\lambda}\exp\left(-\frac{(\theta-\mu)^2}{2\frac{\tau^2}{\lambda}}\right)\lambda^{\alpha-1}\exp(-\beta\lambda)\\
&= \lambda^{\alpha-1/2}\exp\left(-\frac{\tau^2+(\theta-\mu)^2}{2\tau^2}\lambda\right)\\
&= \exp\left(-\frac{\tau^2+(\theta-\mu)^2}{2\tau^2}\lambda\right).
\end{aligned}$$

This means that $[\lambda|\bar{y},\theta] \sim \mathcal{E}\left(-\frac{\tau^2+(\theta-\mu)^2}{2\tau^2}\lambda\right)$.

(b) We implement the Gibbs sampler and obtain the histogram of θ (shown in Figure 3). The Matlab for this problem is attached in Appendix B.

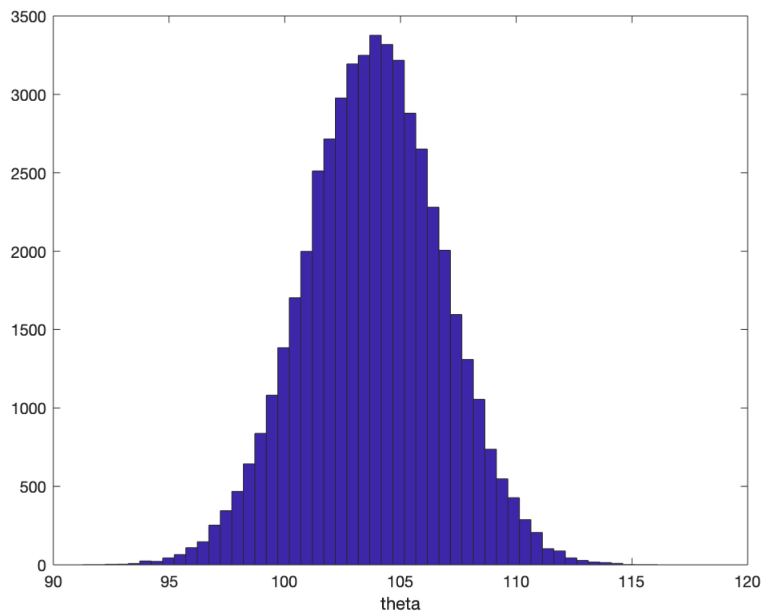

Figure 3: Histogram of θ

The posterior mean and variance are estimated as 103.8367 and 8.7567, respectively. The 95% credible set is estimated as $(97.9998, 109.6256)$.

A Matlab Code for Problem 1

```
clear all
close all

B = 100000;
burn = 0.1*B;

x = -2;
theta = 0.5;     % random starting theta

% Parameters for the proposal distribution
alpha = 1;
%beta = 3;   % first option
beta = 1/theta;      % second option

thetas = [];     % generated samples
accepted = [];  % accepted generated samples
thetas_prop = [];    % collected proposed theta
r_gen = [];      % collect min(1,rho)

for i = 1:B
    theta_prop = gamrnd(alpha, 1/beta);
    thetas_prop = [thetas_prop theta_prop];

    % first option
    %r =
    %exp(-beta*theta)*exp(-3*theta_prop)/(exp(-beta*theta_prop)*exp(-3*theta));
    % second option
    r = exp(-3*theta_prop)*exp(-beta*theta/theta_prop)/...
        (exp(-3*theta)*exp(-beta*theta_prop/theta));
    r_gen = [r_gen r];
    rho = min(1,r);
    U = rand(1);

    if (U<rho)
        theta = theta_prop;
        accepted = [accepted 1];
        thetas = [thetas theta];
    else
        accepted = [accepted 0];
    end
end
estimated_mean = mean(thetas(ceil(burn):end))
efficiency=mean(accepted)

t=linspace(0,3,200);
histogram(thetas,'FaceColor','blue','normalization','pdf')
```

```
hold on
plot(t,gampdf(t,1,1/3),'r-+','MarkerSize',2)
hold off
xlabel('Generated $\theta|X$','Interpreter','Latex')
ylabel('Relative Freq.','Interpreter','Latex')
legend('MCMC','Theoretical')
```

B Matlab Code for Problem 2

```
close all
clear all
% Random number seed
rand('state', 10);

y_bar = 103.5;
NN = 50000;
thetas = []; lambdas = [];
n = 10;
mu = 110;
tau = sqrt(120);
sigma2 = 90;

theta = mu;  % Set parameters as the prior mean
lambda = 1;

for i = 1:NN
    theta_mean = tau^2*y_bar/(tau^2+lambda*sigma2/n) ...
        +lambda*sigma2*mu/(n*tau^2+lambda*sigma2);
    theta_var = tau^2*sigma2/(n*tau^2+lambda*sigma2);
    lambda_mean = (tau^2+(theta-mu)^2)/(2*tau^2);

    newtheta = normrnd(theta_mean, sqrt(theta_var));
    newlambda = exprnd(lambda_mean);

    thetas = [thetas newtheta];
    lambdas = [lambdas newlambda];
    theta = newtheta;
    lambda = newlambda;
end
burn = 1000;
mean(thetas(burn:end))
var(thetas(burn:end))

lower_bound = prctile(thetas, 2.5)
upper_bound = prctile(thetas, 97.5)

```

```
hist(thetas, 50)
xlabel('theta')
```