Pseudocode for Bayesian Posterior MCMC Sampling Methods

1. Metropolis-Hastings (MH)

Algorithm 1 Metropolis-Hastings Algorithm

```
Initialize parameter θ_0
for t = 1 to T do
    Propose a new state θ' ~ q(θ'|θ_{t-1}) (proposal distribution)
    Compute acceptance ratio:
```

$$r = \frac{p(\theta'|\text{data})q(\theta_{t-1}|\theta')}{p(\theta_{t-1}|\text{data})q(\theta'|\theta_{t-1})}$$

```
    Accept θ' with probability min(1, r), otherwise set θ_t = θ_{t-1}
end for
```

2. Gibbs Sampling

Algorithm 2 Gibbs Sampling Algorithm

```
Initialize parameters θ = (θ_1, θ_2, ..., θ_d)
for t = 1 to T do
    for i = 1 to d do
        Sample θ_i^(t) ~ p(θ_i|θ_{-i}, data), where θ_{-i} are all parameters except θ_i
    end for
end for
```

3. Hamiltonian Monte Carlo (HMC)

Algorithm 3 Hamiltonian Monte Carlo Algorithm

```
Initialize parameter θ_0 and momentum p_0
for t = 1 to T do
    Sample momentum p ~ N(0, M)
    Simulate Hamiltonian dynamics:
      • Compute gradient of log-posterior with respect to θ
      • Update θ and p using leapfrog integration
    Accept or reject the new state (θ', p') based on the Hamiltonian
end for
```

4. No-U-Turn Sampler (NUTS)

Algorithm 4 No-U-Turn Sampler Algorithm

Initialize θ_0
for $t = 1$ to T **do**
    Perform Hamiltonian steps as in HMC
    Detect and stop when a U-turn is encountered (momentum reverses)
    Adjust step size and path length adaptively
end for

5. Slice Sampling

Algorithm 5 Slice Sampling Algorithm

Initialize θ_0
for $t = 1$ to T **do**
    Sample threshold $u \sim \text{Uniform}(0, p(\theta_{t-1}))$
    Define slice $S = \{\theta : p(\theta) > u\}$
    Sample θ_t uniformly from S using interval sampling
end for

6. Reversible Jump MCMC (RJ-MCMC)

Algorithm 6 Reversible Jump MCMC Algorithm

Initialize parameter θ_0 and model dimension k_0
for $t = 1$ to T **do**
    Propose a move:
    - Add/remove parameters (birth/death) or change model dimension k'
    - Propose new parameters θ'

    Compute acceptance ratio:

$$r = \frac{p(\text{data}|\theta')p(\theta')q(\theta_{t-1}|\theta')}{p(\text{data}|\theta_{t-1})p(\theta_{t-1})q(\theta'|\theta_{t-1})}$$

    Accept or reject (θ', k')
end for

Algorithm 7 Particle MCMC Algorithm

Initialize particles $\{x_1^{(i)}\}_{i=1}^N$
for $t = 1$ to T **do**
  Update particles using Sequential Monte Carlo (SMC):
  - Resample particles based on weights
  - Propagate particles forward

  Sample posterior parameters θ given particle weights
end for

Algorithm 8 Approximate Bayesian Computation MCMC Algorithm

Initialize parameter θ_0
for $t = 1$ to T **do**
  Propose $\theta' \sim q(\theta'|\theta_{t-1})$
  Simulate data $\tilde{y}$ under θ'
  Compute distance $\Delta(\tilde{y}, \text{data})$
  if $\Delta(\tilde{y}, \text{data}) \leq \epsilon$ **then**
    Accept θ'
  else
    Set $\theta_t = \theta_{t-1}$
  end if
end for

7. Particle MCMC (PMCMC)

8. Approximate Bayesian Computation (ABC-MCMC)

9. Polya-Gamma Sampling

Algorithm 9 Polya-Gamma Sampling Algorithm

Initialize β_0 for logistic regression
for $t = 1$ to T **do**
  Sample Polya-Gamma auxiliary variables ω_i for each observation
  Update β from the Gaussian posterior conditional on ω
end for

10. Sequential Monte Carlo (SMC)

Algorithm 10 Sequential Monte Carlo Sampler Algorithm

```
Initialize particles $\{\theta^{(i)}\}_{i=1}^{N}$ and weights $w^{(i)} = 1/N$
for $t = 1$ to $T$ do
    Update weights $w^{(i)} \propto p(\theta^{(i)}|\text{data})$
    Resample particles based on weights
    Propagate particles $\theta^{(i)} \sim p(\theta|\theta^{(i-1)})$
end for
```