STA 601 - Lab 8

Kedar Prabhudesai

November 8, 2013

Data Likelihood:

$$L(x;\mu,\sigma) = \prod_{i=1}^{n} \frac{1}{x_i\sigma\sqrt{2\pi}} exp\left[\frac{-(lnx_i-\mu)^2}{2\sigma^2}\right]$$

Let $\tau = 1/\sigma^2$

$$\begin{aligned} L(x;\mu,\tau) &= \prod_{i=1}^{n} \frac{\sqrt{\tau}}{x_i\sqrt{2\pi}} exp\left[\frac{-\tau(lnx_i-\mu)^2}{2}\right] \\ \therefore L(x;\mu,\tau) &= \frac{\tau^{n/2}}{x_i^n 2\pi^{n/2}} exp\left[-\frac{\tau}{2}\sum_{i=1}^{n}(lnx_i-\mu)^2\right] \end{aligned}$$

Priors:

$$\begin{aligned} \mu &\sim \mathcal{N}(\mu_0,\tau_0) \therefore p(\mu) = \frac{\sqrt{\tau_0}}{\sqrt{2\pi}} exp\left[\frac{-\tau_0(\mu-\mu_0)^2}{2}\right] \\ \tau &\sim Gamma(\alpha,\beta) \therefore p(\tau) = \frac{\beta^\alpha}{\Gamma(\alpha)}\tau^{\alpha-1}exp(-\beta\tau) \end{aligned}$$

Posterior:

$$\begin{aligned} p(\mu,\tau \mid x) &\propto L(x;\mu,\tau)\times p(\mu)\times p(\tau) \\ \therefore p(\mu,\tau \mid x) &\propto \frac{\tau^{n/2}}{x_i^n} exp\left[-\frac{\tau}{2}\sum_{i=1}^{n}(lnx_i-\mu)^2\right] \times \sqrt{\tau_0}exp\left[\frac{-\tau_0(\mu-\mu_0)^2}{2}\right] \times \tau^{\alpha-1}exp(-\beta\tau) \end{aligned}$$

Full Conditionals:

$$
\begin{aligned}
p(\mu \mid \tau, x) &\propto exp\left[-\frac{\tau}{2}\sum_{i=1}^{n}(lnx_i - \mu)^2 - \frac{\tau_0}{2}(\mu - \mu_0)^2\right] \\
p(\tau \mid \mu, x) &\sim Gamma\left(\alpha + \frac{n}{2}, \frac{\sum_{i=1}^{n}(lnx_i - \mu)^2}{2} + \beta\right)
\end{aligned}
$$

In our Gibbs Sampler, we update μ using Metropolis-Hastings and τ using the Gamma Distribution.

Gibbs Sampler:

Start with $\{\mu^{(0)}\}$

- Sample, $\mu' \sim \mathcal{N}(\mu^{(s)}, \sigma_{cand})$
- Compute acceptance ratio, $r = p(\mu' \mid \tau^{(s)}, x)/p(\mu \mid \tau^{(s)}, x)$
- Draw, $u \sim$ Uniform$[0, 1]$. If $u < r$, $\mu^{(s+1)} = \mu'$, else $\mu^{(s+1)} = \mu^{(s)}$.
- Sample, $\tau^{(s+1)} \sim p(\tau \mid \mu^{(s+1)}, x)$

Posterior Predictive:

$$
p(x_{n+1} \mid x_1, x_2, \ldots, x_n) = \int\int p(x_{n+1} \mid \mu, \tau)p(\mu, \tau \mid x_1, x_2, \ldots, x_n)d\mu d\tau
$$

We can sample from the posterior predictive using Monte-Carlo procedure. Use $\{\mu^{(s)}, \tau^{(s)}\}$ samples from the Gibbs Sampler and sample from $x^{(s)} \sim p(x \mid \mu^{(s)}, \tau^{(s)})$. Then posterior predictive can be approximated as, $p(x_{n+1} \mid x_1, x_2, \ldots, x_n) = \frac{\sum_{s=1}^{S} x^{(s)}}{S}$.

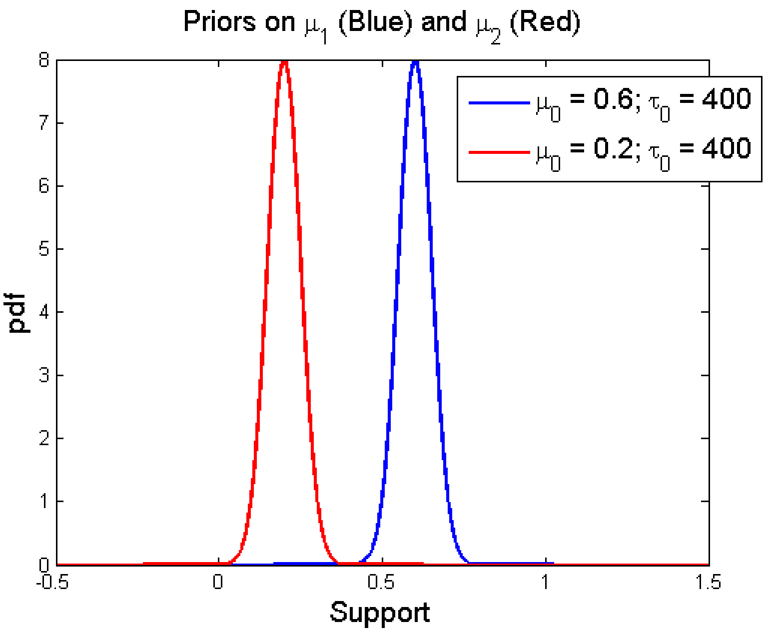
Weekday Prior Values:

Prior parameters for μ_1: $\mu_0 = 0.6, \tau_0 = 400$.
Prior parameters for τ_1: $\alpha = 1, \beta = 20$.

Weekend Prior Values:

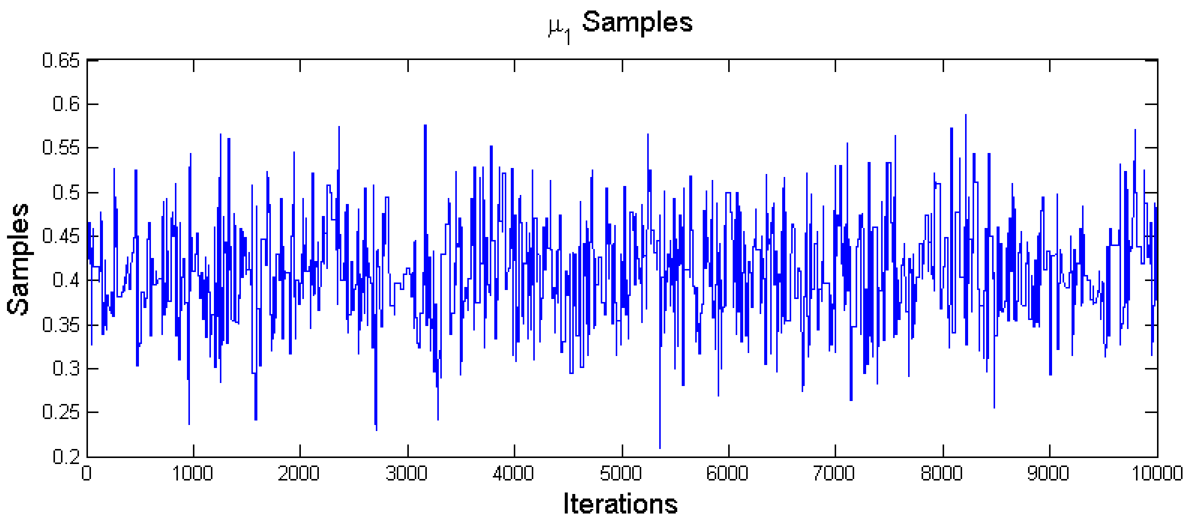
Prior parameters for μ_2: $\mu_0 = 0.2, \tau_0 = 400$.
Prior parameters for τ_2: $\alpha = 1, \beta = 20$.

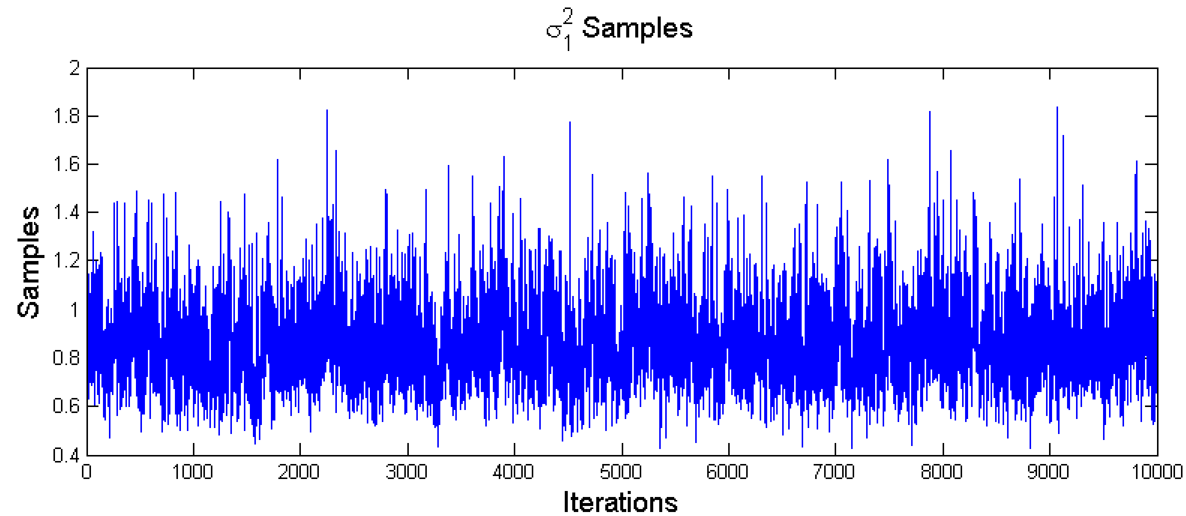
We verify that, $P(\mu_1 > \mu_2) = 1$. This is comparison of the two priors.



Sampling Results:

Following are trace plots for μ_1 and σ_1^2.





Estimates and 95% Credible Intervals:
$\mu_1 \to 0.4102$ $[0.3037, 0.5219]$.
$\sigma_1^2 \to 0.8585$ $[0.5750, 1.2491]$.
$\mu_2 \to 0.1533$ $[0.0462, 0.2561]$.
$\sigma_2^2 \to 2.4503$ $[1.4273, 4.1320]$.

Probabilities of Interest:
Posterior $P(\mu_1 > \mu_2) = 0.9998$. Posterior $P(\sigma_1 > \sigma_2) = 0.0012$.
Posterior Predictive, $P(x_{n+1}^{Weekday} > x_{n+1}^{Weekend} \mid x_1, x_2, \ldots, x_n) = 0.5470$.

Appendix:

```matlab
%% STA 601: Lab 8
% Author: Kedar S Prabhudesai
% Created on: 11/8/2013
function sta601_ksp6_lab8
close all;
clear all;

% Get Data
X = importdata('data.txt');
XWkDays = [];
XWkEnds = [];

% Separate data into Weekdays and Weekends
for iData = 2:numel(X)
    quotes = find(X{iData} == '"');
    day = X{iData}(quotes(1)+1:quotes(2)-1);
    value = str2double(X{iData}(1:quotes(1)-2));

    if strcmp(day,'Saturday') || strcmp(day,'Sunday')
        XWkEnds = cat(1,XWkEnds,value);
    else
        XWkDays = cat(1,XWkDays,value);
    end
end

[mu1Samples,s21Samples,xWkDSamples,EstMu1,Mu1ConfInts,EstS21,S21ConfInts] = GibbsSampler(...
    XWkDays,0.6,400,1,0.05);

% Manage Plotting
figure('Position',[67   304   922   345]);
plot(mu1Samples,'b-');
xlabel('Iterations','FontSize',14);
ylabel('Samples','FontSize',14);
title('\mu_1 Samples','FontSize',14);

figure('Position',[67   304   922   345]);
plot(s21Samples,'b-');
xlabel('Iterations','FontSize',14);
ylabel('Samples','FontSize',14);
title('\sigma_1^2 Samples','FontSize',14);

[mu2Samples,s22Samples,xWkESamples,EstMu2,Mu2ConfInts,EstS22,S22ConfInts] = GibbsSampler(...
    XWkEnds,0.2,400,1,0.05);

% Compute Probabilities of Interest
M1GtM2 = mean(mu1Samples > mu2Samples);
S1GtS2 = mean(sqrt(s21Samples) > sqrt(s22Samples));
X1GtX2 = mean(xWkDSamples > xWkESamples);

keyboard
    function [muSamples,s2Samples,xSamples,EstMu,MuConfInts,EstS2,S2ConfInts] = ...
        GibbsSampler(data,mu0,t0,a,b)
%           % Prior Parameters: mu0, t0, a, b
%           % Remember that 'b' parameter in matlab's gamma function is in fact '1/b'
        n = numel(data);

        % Target Distribution for mu
        muFullCond =  @(t,mu) exp(-0.5*t*sum((log(data)-mu).^2) - 0.5*t0*(mu-mu0)^2);

        % Number of Trials
```

```
        nTrials = 12000;
        % Burn-In
        nBurnIn = 2000;
        % Proposal Distribution Std Dev
        SCand = 0.5;

        % Initialize
        muSamples = zeros(1,nTrials);
        tSamples = zeros(1,nTrials);
        xSamples = zeros(1,nTrials);

        % Full Conditional distribution for tau
        tFullCond = makedist('Gamma','a',a+n/2,'b',1/(0.5*sum((log(data)-muSamples(1)).^2) ...
            + b));
        tSamples(1) = tFullCond.random();

        % Gibbs Sampling
        for iTrial = 2:nTrials
            home;disp(iTrial);
            % Update mu | tau,x using M-H

            % Step 1: Sample from mu'|mu(s)
            muPrime = normrnd(muSamples(iTrial-1),SCand);

            % Step 2: Compute Acceptance Ratio
            r = muFullCond(tSamples(iTrial-1),muPrime)/muFullCond(tSamples(iTrial-1),...
                muSamples(iTrial-1));

            % Step 3: Accept/Reject
            u = rand;
            if u < r
                muSamples(iTrial) = muPrime;
            else
                muSamples(iTrial) = muSamples(iTrial-1);
            end

            % Update tau | mu,x
            tFullCond.b = 1/(0.5*sum((log(data)-muSamples(iTrial)).^2) + b);
            tSamples(iTrial) = tFullCond.random();

            % Get Samples from likelihood for Posterior Predictive
            xSamples(iTrial) = lognrnd(muSamples(iTrial),sqrt(1/tSamples(iTrial)));
        end

        % Burn-In
        muSamples(1:nBurnIn) = [];
        tSamples(1:nBurnIn) = [];
        xSamples(1:nBurnIn) = [];

        % Convert to Sigma^2
        s2Samples = 1./tSamples;

        % Get Estimates and Credible Intervals
        EstMu = mean(muSamples);
        MuConfInts = quantile(muSamples,[0.025 0.975]);

        EstS2 = mean(s2Samples);
        S2ConfInts = quantile(s2Samples,[0.025 0.975]);
    end

end
```