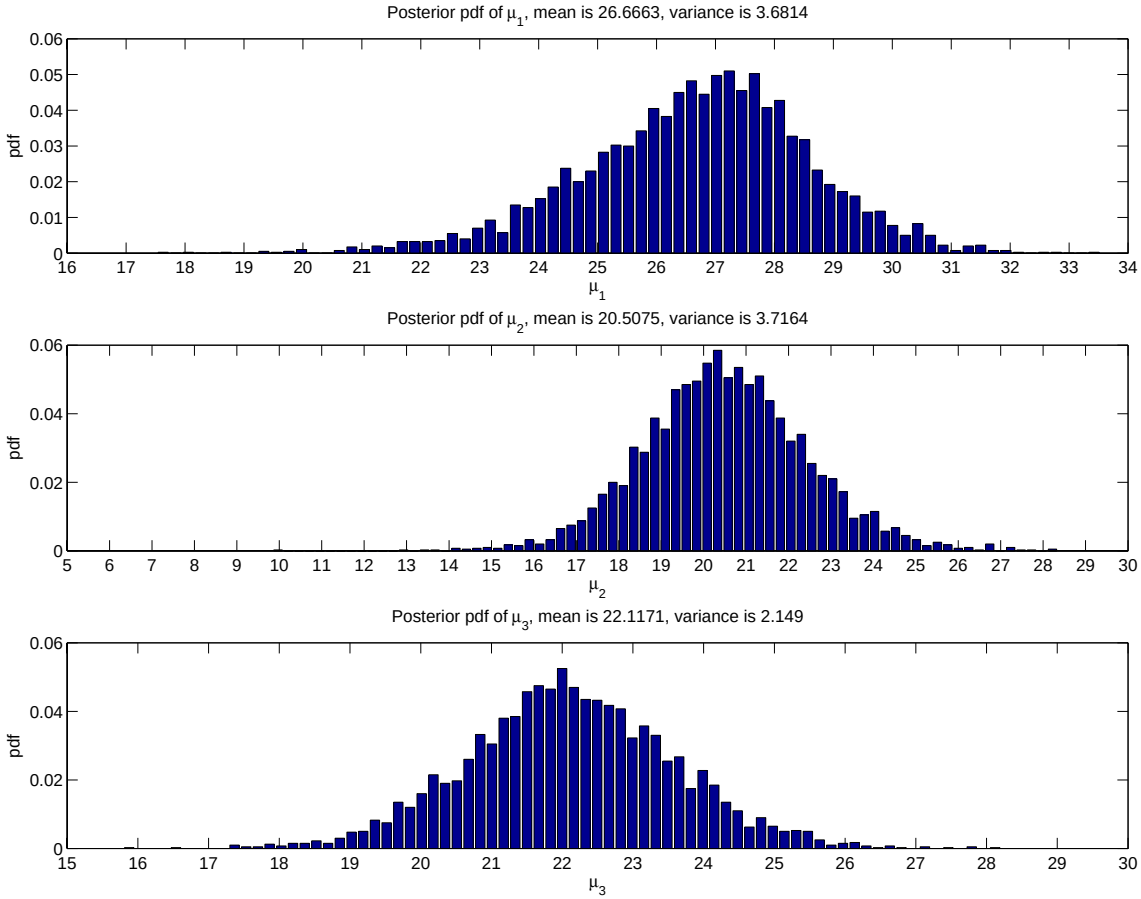
Bayesian Data Analysis, Midterm I

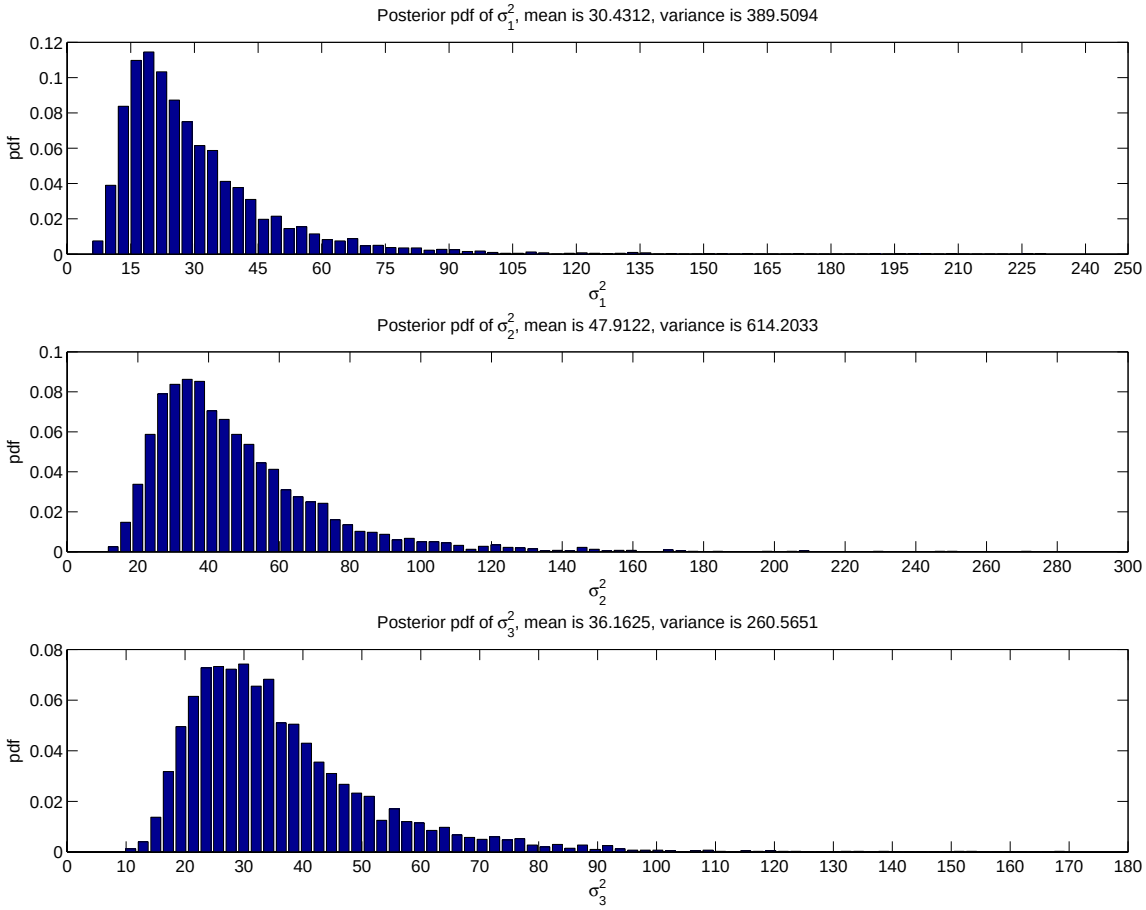
Bugra Gedik
bgedik@cc.gatech.edu

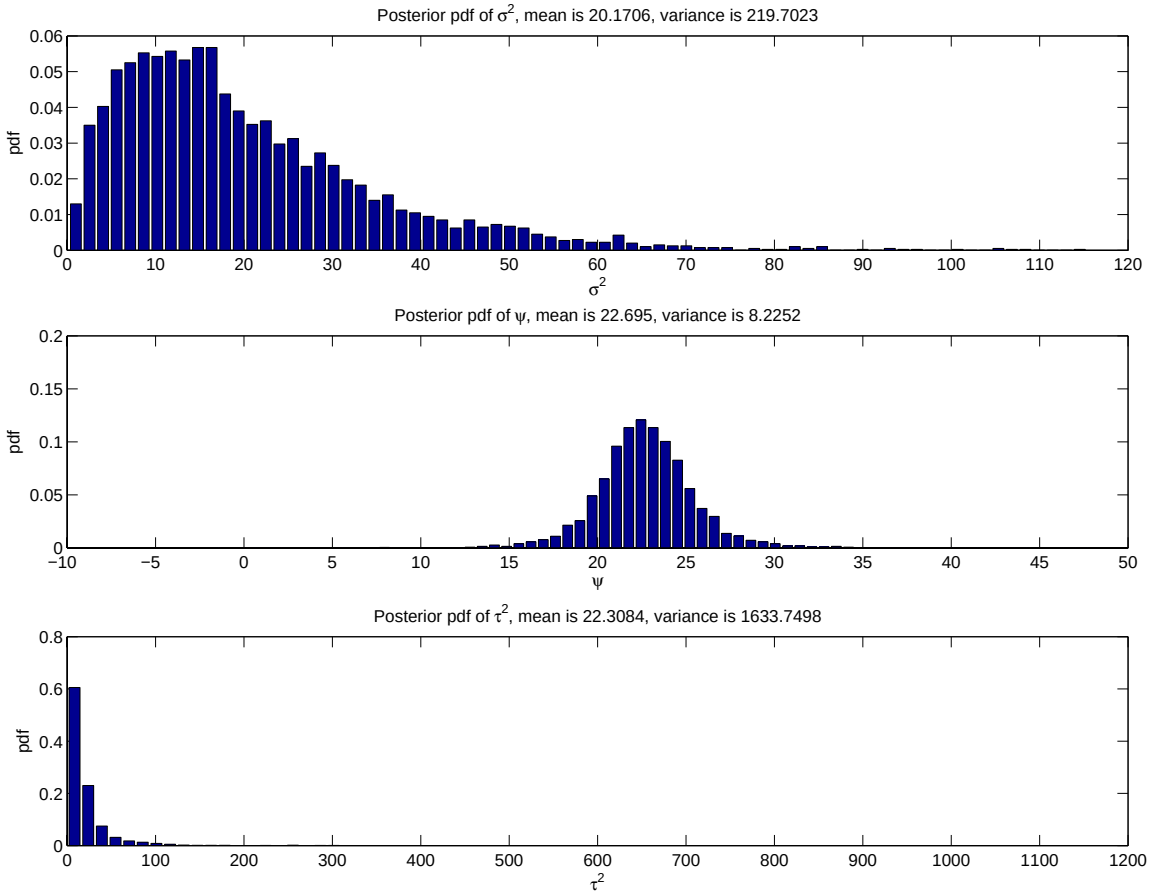
October 23, 2004

Q1)

I have used Gibs sampler to solve this problem. 5,000 iterations with burn-in value of 1,000 is used. The resulting histograms representing the posterior distributions of $\mu_i, \ldots, \mu_k$, $\sigma_1^2, \ldots, \sigma_k^2$, and σ^2, ψ, τ^2 are given below. Posterior sample means and variances of the parameters are also indicated on the histograms. The corresponding Matlab code is included at the end.




Posterior pdf of σ_1^2, mean is 30.4312, variance is 389.5094
pdf
σ_1^2
Posterior pdf of σ_2^2, mean is 47.9122, variance is 614.2033
pdf
σ_2^2
Posterior pdf of σ_3^2, mean is 36.1625, variance is 260.5651
pdf
σ_3^2



Q1 Matlab Code

```
% data related definitions
data = { [25.8 19.8 28.6 29.4 22.3 33.8 33.8 27.8 29.6]; ...
         [16.5 23.5 13.5 34.6 16.9 18.8 26.1 18.4 17.2 11.6 20.2]; ...
         [24.0 29.1 16.0 24.8 27.0 10.9 11.8 23.2 17.7 23.9 24.6 24.0 27.2 23.7] };
k = length(data);
n = zeros(1, k); % data row lengths
dms = zeros(1, k); % data row means
dvs = zeros(1, k); % data row variances
for i=1:k,
    n(i) = length(data{i});
    dms(i) = mean(data{i});
    dvs(i) = var(data{i});
end

% hyper parameters
a0 = 1; c0 = 1; d0 = 1; f0 = 1;
g0 = 0.1; psi0 = 10; xi0 = 0.1;

% initialize parameters
tausq  = 1/rand_gamma(c0/2,d0/2,1,1);
psi    = rand_nor(psi0,tausq/xi0,1,1);
mus    = rand_nor(psi,tausq,1,k);
sigsq  = rand_gamma(f0/2,g0/2,1,1);
sigsqs = 1./rand_gamma(a0/2,(a0*sigsq)/2,1,k);

% gibs parameters
itrcnt = 5000; burnin = 1000;
effcnt = itrcnt - burnin;

% collected parameter values
thetas = zeros(effcnt, 2*k+3);
```

```
for iter=1:itrcnt, % perform gibs
    %tausq step%
    ga = (c0 + k + 1) / 2;
    gb = ( d0 + sum((mus-psi).^2) + xi0*(psi-psi0)^2 ) / 2;
    tausq = 1 / rand_gamma(ga,gb,1,1);
    %psi step%
    nm = ( sum(mus) + xi0*psi0 ) / (k + xi0);
    nv  =  tausq / (k + xi0);
    psi = rand_nor(nm,sqrt(nv),1,1);
    %mus steps%
    for i=1:k,
        nv  = 1 / ( n(i)/sigsqs(i) + 1/tausq );
        nm = ( (n(i)*dms(i))/sigsqs(i) + psi/tausq ) * nv;
        mus(i) = rand_nor(nm,sqrt(nv),1,1);
    end
    %sigsq step%
    ga = (f0+k*a0) / 2;
    gb = ( g0 + a0*sum(1./sigsqs) ) / 2;
    sigsq = rand_gamma(ga,gb,1,1);
    %sigsqs steps%
    for i=1:k,
        ga = (a0+n(i))/2;
        gb = ( a0*sigsq + (n(i)-1)*dvs(i) + n(i)*(dms(i)-mus(i))^2 ) / 2;
        sigsqs(i) = 1 / rand_gamma(ga,gb,1,1);
    end
    %collect%
    eiter = iter - burnin;
    if(eiter > 0)
        thetas(eiter, :) = [mus, sigsqs, sigsq, psi, tausq];
    end
end

figure; B = 75;
for i=1:k,
    subplot(k, 1, i);
    muis = thetas(:,i);
    [cnt, pos] = hist(muis, B);
    bar(pos, cnt/sum(cnt));
    title(['Posterior pdf of \mu_{' num2str(i) '}, ' ...
           'mean is ' num2str(mean(muis)) ', ' ...
           'variance is ' num2str(var(muis))]);
    xlabel(['\mu_{' num2str(i) '}']); ylabel('pdf');
end

figure;
for i=1:k,
    subplot(k, 1, i);
    sigsqis = thetas(:,k+i);
    [cnt, pos] = hist(sigsqis, B);
    bar(pos, cnt/sum(cnt));
    title(['Posterior pdf of \sigma^{2}_{' num2str(i)  '}, ' ...
           'mean is ' num2str(mean(sigsqis)) ', ' ...
           'variance is ' num2str(var(sigsqis))]);
    xlabel(['\sigma^{2}_{' num2str(i) '}']); ylabel('pdf');
end

figure; subplot(3, 1, 1);
sigsqs = thetas(:,2*k+1);
[cnt, pos] = hist(sigsqs, B);
bar(pos, cnt/sum(cnt));
title(['Posterior pdf of \sigma^{2}, ' ...
       'mean is ' num2str(mean(sigsqs)) ', ' ...
       'variance is ' num2str(var(sigsqs))]);
xlabel('\sigma^{2}'); ylabel('pdf');

subplot(3, 1, 2);
psis = thetas(:,2*k+2);
[cnt, pos] = hist(psis, B);
bar(pos, cnt/sum(cnt));
title(['Posterior pdf of \psi, ' ...
       'mean is ' num2str(mean(psis)) ', ' ...
       'variance is ' num2str(var(psis))]);
xlabel('\psi'); ylabel('pdf');

subplot(3, 1, 3);
tausqs = thetas(:,2*k+3);
[cnt, pos] = hist(tausqs, B);
bar(pos, cnt/sum(cnt));
title(['Posterior pdf of \tau^{2}, ' ...
       'mean is ' num2str(mean(tausqs)) ', ' ...
       'variance is ' num2str(var(tausqs))]);
xlabel('\tau^{2}'); ylabel('pdf');
```

Q2)

a) I first find the joint posterior distribution of μ, σ up to a normalizing constant:

$$p(y_i|\mu,\sigma) = \frac{1}{\sigma} e^{-\frac{y_i-\mu}{\sigma}} e^{-e^{-\frac{y_i-\mu}{\sigma}}}$$

μ, σ are independent, thus $p(\mu,\sigma) = p(\mu)p(\sigma)$

$p(\mu) = \mathcal{N}(\mu|0,10^2), p(\sigma) = \mathcal{LN}(\sigma|0,10^2)$

$$p(y_1,\ldots,y_n|\mu,\sigma) = \prod_{i=1}^{n} \left(\frac{1}{\sigma} e^{-\frac{y_i-\mu}{\sigma}} e^{-e^{-\frac{y_i-\mu}{\sigma}}} \right)$$

$$p(y_1,\ldots,y_n|\mu,\sigma) = \frac{1}{\sigma^n} e^{-\frac{n*(\overline{y}-\mu)}{\sigma}} e^{-\sum_{i=1}^{n} e^{-\frac{y_i-\mu}{\sigma}}}$$

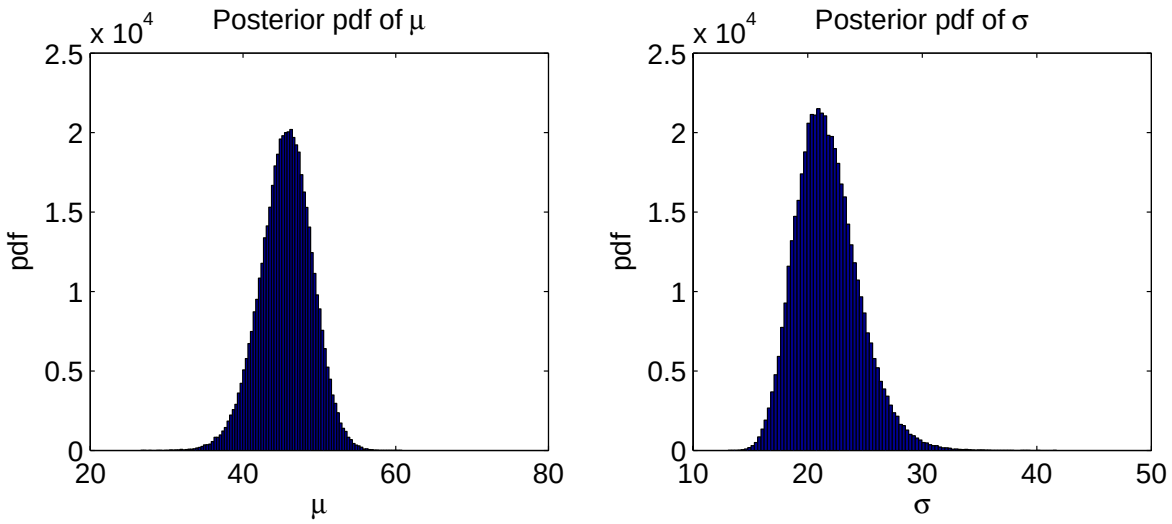
$$p(\mu,\sigma|y_1,\ldots,y_n) \propto p(y_1,\ldots,y_n|\mu,\sigma)p(\mu,\sigma)$$

$$p(\mu,\sigma|y_1,\ldots,y_n) \propto \frac{1}{\sigma^n} e^{-\frac{n*(\overline{y}-\mu)}{\sigma}} e^{-\sum_{i=1}^{n} e^{-\frac{y_i-\mu}{\sigma}}} e^{-\frac{\mu^2}{200}} \frac{1}{\sigma} e^{-\frac{(\ln\sigma)^2}{200}}$$

$$p(\mu,\sigma|y_1,\ldots,y_n) \propto \frac{1}{\sigma^{n+1}} e^{-\frac{n*(\overline{y}-\mu)}{\sigma} - \frac{\mu^2+(\ln\sigma)^2}{200}} e^{-\sum_{i=1}^{n} e^{-\frac{y_i-\mu}{\sigma}}}$$

b) I developed a metropolis algorithm using a bi-variate normal jumping function $\mathcal{N}(.,\Sigma)$. Σ is set to $c \times \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$. I also experimented with $\Sigma = c \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, which gave the same result. c is set to $m \times \frac{2.4}{\sqrt{2}}$, where m is taken as 10, in order to achieve an acceptance percentage of around 40% in metropolis jumps, which is suggested as a plausible value in the course book.

c) In order to calculate $p(y^* \geq 410)$, I used the $\mu^{(i)}$ and $\sigma^{(i)}$ values generated by the ith step of the metropolis algorithm to find $p(y \geq 410|\mu^{(i)},\sigma^{(i)})$ using the CDF function of the Gumbel distribution, i.e. $1 - F(410|\mu^{(i)},\sigma^{(i)})$. Then I took the average over different steps. Formally, $p(y^* \geq 410) = \frac{1}{b-a} \sum_{i=a+1}^{b} \left(1 - F(410|\mu^{(i)},\sigma^{(i)})\right)$, where a is the burn-in value and b is the total number of iterations. The resulting probability is $\approx 2.75 \times 10^{-7}$, when $a = 10,000$ and $b = 500,000$. The resulting posterior distributions of μ and σ are given below. The Matlab code is also included.



Q2 Matlab Code

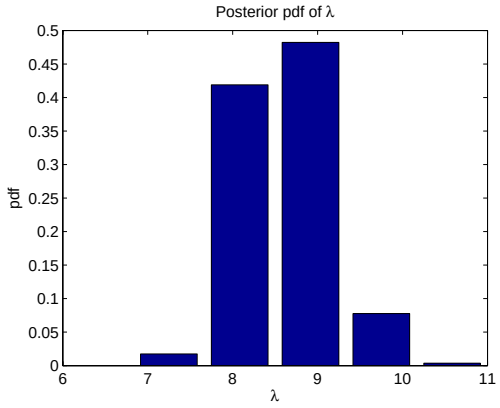
```
% data related parameters
y = [154 49.6 46.7 58.3 70.5 90 70.1 105.7 37.4 40.8 34.7 58.9 ...
     72.2 30 71.6 100 33.7 49.9 56.1 142.3 28.6 54.8 74.1 60 ...
     50.9 38.6 53.4 132.5 50.7 40.8 84.3 38.8 27.4 67 118.7 23.2 ...
     55 67.9 87.3 89 98.7 47.1 71.6 83.6 44.3 41.2 35.9 44.3];
n = length(y); ym = mean(y);

% metropolis parameters
itrcnt = 500000; burnin = 10000;
effcnt = itrcnt - burnin;
c = 10*2.4/sqrt(2);
s = c * [1 0.5; 0.5 1];

% collected parameter values
thetas = zeros(effcnt, 2);

p = 0; % find initial values with non-zero probability
while (p==0),
    theta = [rand_nor(0, 10,1,1) 100*rand];
    %theta = [rand_nor(0, 10,1,1) exp(10*rand_nor(0,1,1,1))];
    mu = theta(1); sig = theta(2);
    p = (1/sig^(n+1)) * exp(- (n*(ym-mu))/sig - (mu^2+log(sig)^2)/200) ...
                      * exp(- sum( exp( -(y-mu)./sig ) ) );
end
mcnt = 0;
for iter=1:itrcnt,
    % propose
    ntheta = rand_MVN(1, theta', s);
    % find ratio
    ntheta(2) = abs(ntheta(2));
    nmu = ntheta(1); nsig = ntheta(2);
    pn = (1/nsig^(n+1)) * exp(- (n*(ym-nmu))/nsig - (nmu^2+log(nsig)^2)/200) ...
                        * exp(- sum( exp( -(y-nmu)./nsig ) ) );
    r = pn / p;
    % decide move
    if (rand < r)
        mcnt = mcnt + 1;
        theta = ntheta; p = pn;
        mu = theta(1); sig = theta(2);
    end
    % collect
    eiter = iter - burnin;
    if(eiter > 0)
        thetas(eiter, :) = theta;
    end
end

fprintf(1, 'Acceptance rate: %f\n', mcnt/itrcnt);

B = 100;
subplot(1, 2, 1);
hist(thetas(:,1), B);
title(['Posterior pdf of \mu']);
xlabel(['\mu']); ylabel('pdf');

subplot(1, 2, 2);
hist(thetas(:,2), B);
title(['Posterior pdf of \sigma']);
xlabel(['\sigma']); ylabel('pdf');

% find probability that y* > 410
target = 410; %% >=410
probs = 1-exp(-exp(-(target-thetas(:,1))./thetas(:,2)));
eprob = mean(probs);
fprintf(1, ['probability of y>= %f is %d\n'], target, eprob);
```

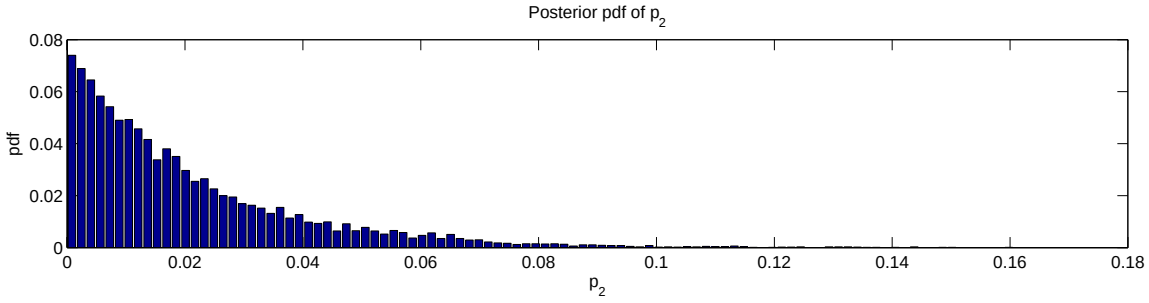
Q3)

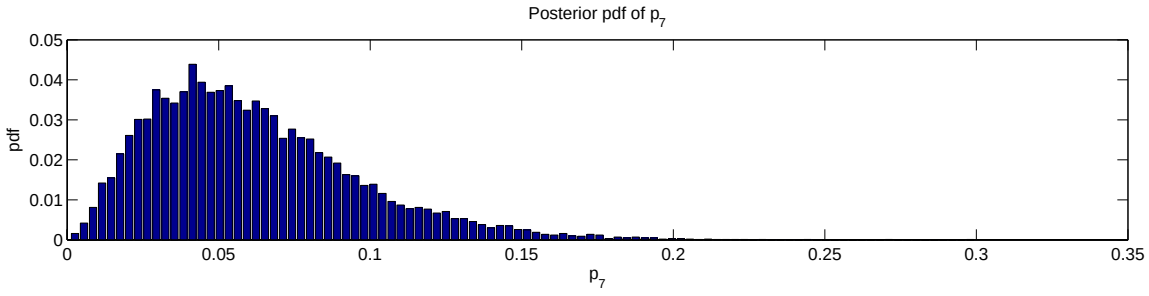
The posterior statistics are as follows:

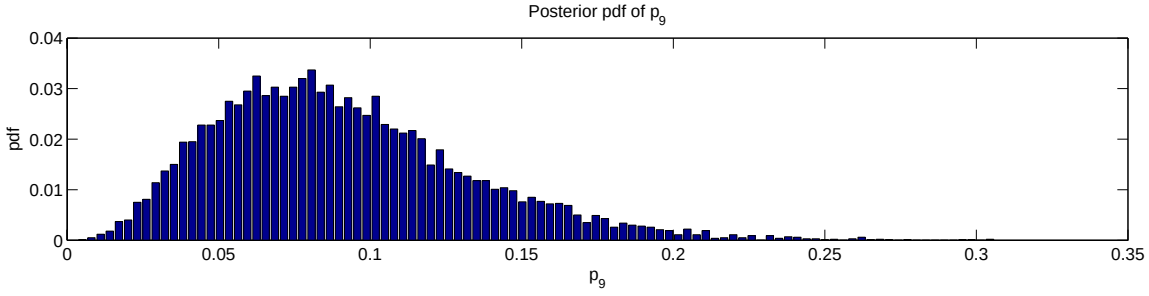
node	mean	sd	MC error	2.5%	median	97.5%
λ	8.631	0.6675	0.007305	8.0	9.0	10.0
p_2	0.02048	0.02012	1.809E-4	5.219E-4	0.01425	0.07257
p_7	0.06138	0.03417	3.476E-4	0.01256	0.05552	0.144
p_9	0.0906	0.04171	4.373E-4	0.02691	0.0847	0.1874
p_{10}	0.1036	0.04318	4.275E-4	0.03582	0.09828	0.2013
p_{13}	0.08195	0.03896	4.372E-4	0.02316	0.07633	0.1738
p_{17}	0.02043	0.02025	2.079E-4	5.54E-4	0.01433	0.07447

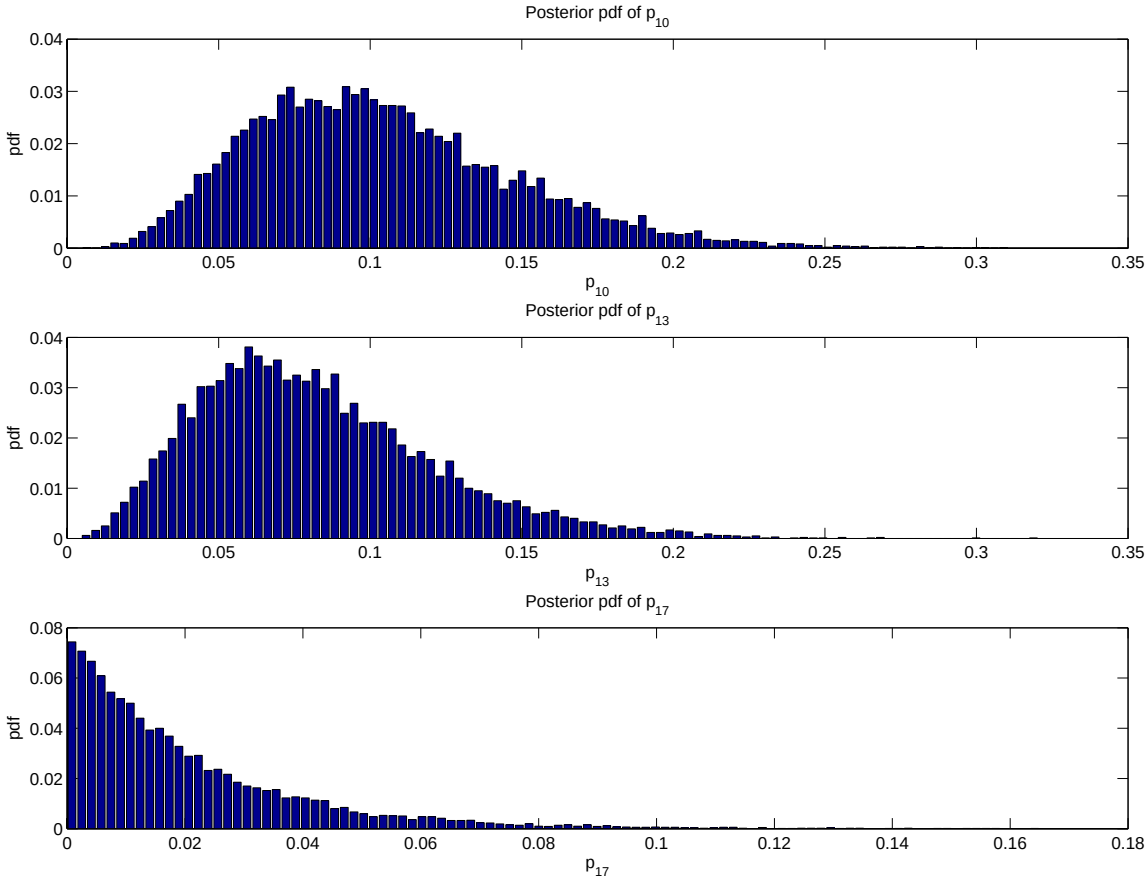
Matlab plots are included below:










Posterior pdf of p_{10}
pdf
p_{10}
Posterior pdf of p_{13}
pdf
p_{13}
Posterior pdf of p_{17}
pdf
p_{17}