VE414 assignment6

Hu Chong 515370910114

July 9, 2019

Q1

(a)

```julia
using Distributions
using Plots
using StatsBase
x = 0.5
function f_post(mu, sigma_2)
    1/ sqrt(sigma_2) *exp(-(x-mu)^2/(2*sigma_2)-
    (mu-10)^2/2)* 1/(400*sigma_2)
end

function g_cond(mu_new, sigma_2_new, mu, sigma_2)
    if sigma_2_new > exp(200) || sigma_2_new < exp(-200)
        return 0.0
    else
        Mean = [mu, sigma_2]
        Var = Array{Float64,2}(undef, 2, 2)
        Var[1] = Var[4] = sigma_2
        Var[2] = Var[3] = 0.1*sigma_2
        return pdf(MvNormal(Mean,Var), [mu_new, sigma_2_new])
    end
end

n = 100000
sample_sigma2_arr = Array{Float64, 1}(undef, n)
sample_mu_arr = Array{Float64, 1}(undef, n)
sample_mu_mean_arr = Array{Float64, 1}(undef, n)
mu_prev = 10.0
sigma_2_prev = exp(1.0)

for i = 1:n
```

```
    # draw from the proposal
    global sigma_2_prev
    global mu_prev
    Mean = [mu_prev, sigma_2_prev]
    Var = Array{Float64,2}(undef, 2, 2)
    Var[1] = Var[4] = sigma_2_prev
    Var[2] = Var[3] = 0.1*sigma_2_prev

    mu, sigma_2 = rand(MvNormal(Mean, Var))
    while sigma_2 >= exp(200) || sigma_2 <= exp(-200)
        mu, sigma_2 = rand(MvNormal(Mean,Var))
    end

    alpha = min(1, f_post(mu, sigma_2)*
    g_cond(mu_prev, sigma_2_prev, mu, sigma_2)
    / (f_post(mu_prev, sigma_2_prev)*
    g_cond(mu, sigma_2, mu_prev, sigma_2_prev)))
    # draw uniform
    v = rand(Uniform(0,1))
    if v <= alpha
        sample_sigma2_arr[i] = sigma_2
        sample_mu_arr[i] = mu
        sigma_2_prev = sigma_2
        mu_prev = mu
    else
        sample_sigma2_arr[i] = sigma_2_prev
        sample_mu_arr[i] = mu_prev
    end
    sample_mu_mean_arr[i] = mean(sample_mu_arr[1:i])
end

plot(sample_mu_mean_arr, label="mean of f_mu|x")
savefig("q1_a_mean.png")
# burn-in 10000
res = fit(Histogram, sample_mu_arr[10000:end], nbins=50)
sample_mu_mode = res.edges[1][findmax(res.weights)[2]]+
res.edges[1].step/2

histogram(sample_mu_arr[10000:end], bins=50)
savefig("q1_a_hist.png")
println("sample mode of mu: $sample_mu_mode")
```

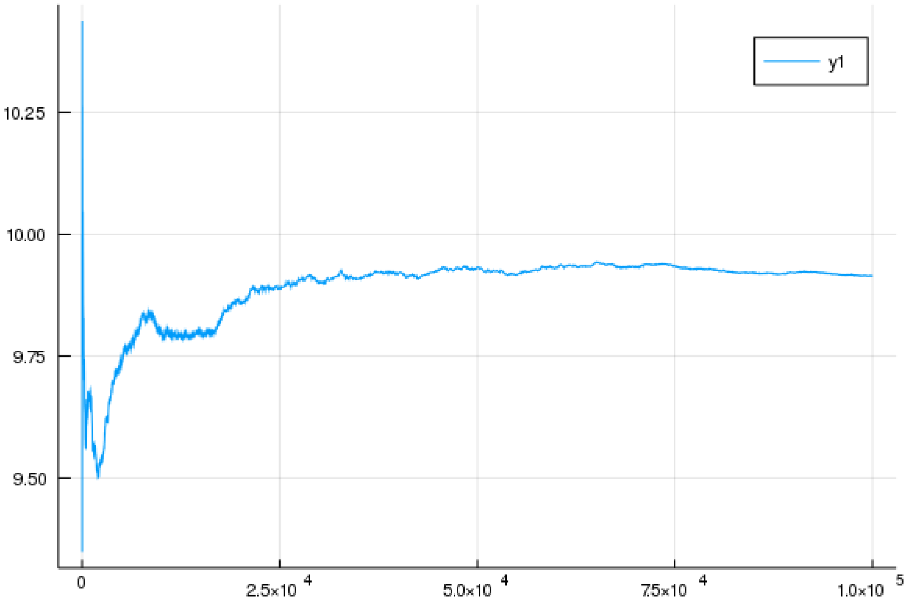

Figure 1: part a) mean of $\mathbf{f}_{\mu|\mathbf{x}}$

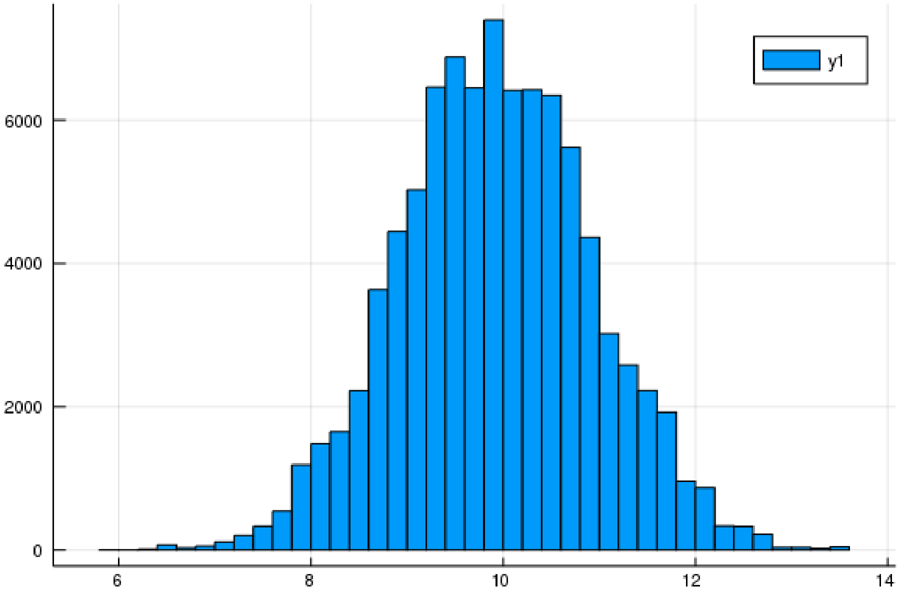

Figure 2: part a) histogram of $\mathbf{f}_{\mu|\mathbf{x}}$

(b)

```
using Distributions
using Plots
using StatsBase

```

```
x = 0.5
n = 10000
sample_sigma2_arr = Array{Float64, 1}(undef, n)
sample_mu_arr = Array{Float64, 1}(undef, n)
sample_mu_mean_arr = Array{Float64, 1}(undef, n)
mu_prev = 10.0
for i = 1:n
    global mu_prev
    sigma_2 = rand(InverseGamma(1/2, (x-mu_prev)^2/2))
    while sigma_2 >= exp(200) || sigma_2 <= exp(-200)
        sigma_2 = rand(InverseGamma(1/2, (x-mu_prev)^2/2))
    end
    mu = rand(Normal((x/sigma_2+10)/(1/sigma_2 + 1),
     sigma_2/(sigma_2+1)))
    sample_sigma2_arr[i] = sigma_2
    sample_mu_arr[i] = mu
    sample_mu_mean_arr[i] = mean(sample_mu_arr[1:i])
    mu_prev = mu

end
plot(sample_mu_mean_arr, label="mean of f_mu|x")
savefig("q1_b_mean.png")
# burn-in 2500
res = fit(Histogram, sample_mu_arr[2500:end], nbins=50)
sample_mu_mode = res.edges[1][findmax(res.weights)[2]]+
res.edges[1].step/2
histogram(sample_mu_arr[2500:end], bins=50)
savefig("q1_b_hist.png")
println("sample mode of mu: $sample_mu_mode")
```

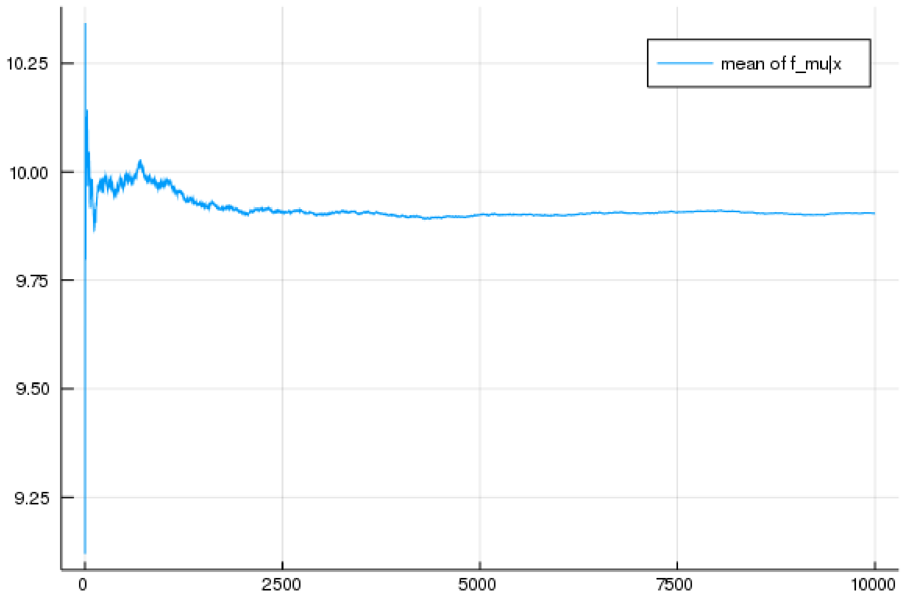

Figure 3: part b) mean of $\mathbf{f}_{\mu|\mathbf{x}}$

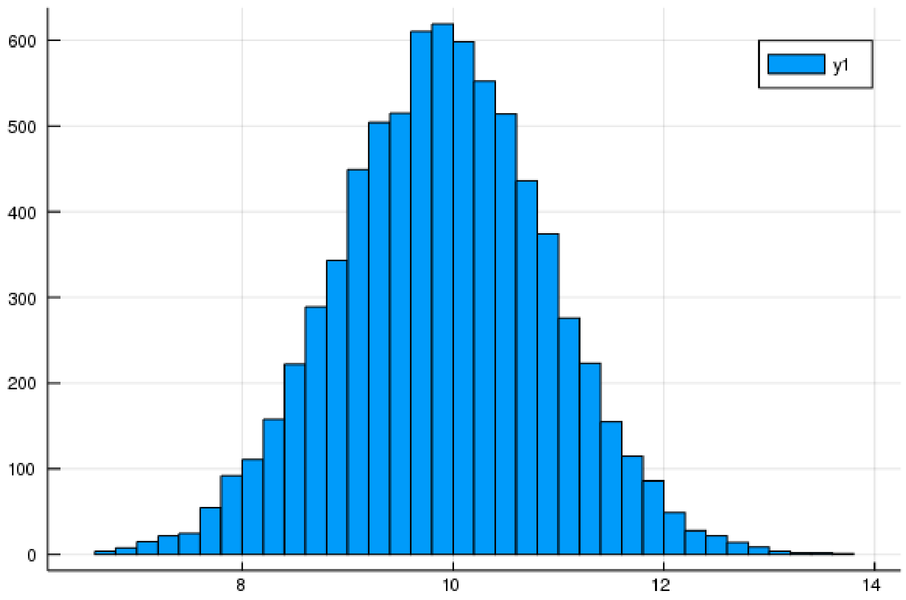

Figure 4: part b) histogram of $\mathbf{f}_{\mu|\mathbf{x}}$

(c)

```
using Distributions
using Plots
using StatsBase
x = 0.5
```

```
epsilon = 1e-9
mu_prev = 10.0

for i = 1:1e6
    global mu_prev
    mu = (x+10*(x-mu_prev)^2)/(1+(x-mu_prev)^2)
    if abs(mu_prev-mu) < epsilon
        mu_prev = mu
        break
    end
    mu_prev = mu
end
println("sample mode of mu: $mu_prev")
```