UM–SJTU Joint Institute

Bayesian Analysis
(VE414)

# Assignment 5

Name: Wu Guangzheng
Student ID: 515370910014

# 1 Question 1

## a)

With the results in the last assignment, the constant $c$ is 0.645, that is

$$f_{Y|X} = 0.645 \times \exp(-\frac{(x-y)^2}{2}) \times \frac{1}{1+y^2}$$

And we have

$$0 < \frac{1}{1+y^2} \leq 1, \qquad y \in R$$

So

$$f_{Y|X} \leq 0.645 \times \exp(-\frac{(x-y)^2}{2}) = 0.645 \cdot \sqrt{2\pi\sigma^2} \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp(-\frac{(y-\mu)^2}{2\sigma^2})$$

Where

$$\sigma = 1, \qquad \mu = x = 0.5$$

So we have

$$f_{Y|X} \leq AMq(y) < Mq(y)$$
$$A = 0.645, \qquad M = \sqrt{2\pi}, \qquad q(y) \sim \text{Norm}(0.5, 1)$$

## b)

We can get the following result

| Grid Size $n$ | $\mathbb{E}[Y \mid X = 0.5]$ |
|---|---|
| 50 | 0.2059457941459266 |
| 250 | 0.26953845572240126 |
| 750 | 0.2728096525929137 |
| 1500 | 0.2800835653746651 |
| 3000 | 0.2585988667944461 |

With code

```
using Distributions

sample_sizes = [50, 250, 750, 1500, 3000]

for n in sample_sizes
    samples = Array{Float64}(undef, n)
    i = 1
```

```
    while i <= n
        v = rand(1)[1]
        y = rand(Normal(0.5, 1), 1)[1]
        if v <= 1 / (1+y^2)
            samples[i] = y
            i+=1
        end
    end
    exp = sum(samples) / n
    println(exp)
end
```

## c)

We can use the same $g(y)$ here, which is

$$q(y) \sim \text{Normal}(0.5, 1)$$

So we have

$$\begin{aligned}
E\left[Y \mid X\right] &= \int Y \cdot f_{Y|X} dy = \int \frac{Y \cdot f_{Y|X}}{q(y)} \cdot q(y) dy \\
&\approx \frac{1}{n} \sum_{i=1}^{n} \frac{p(y^{(i)})}{q(y^{(i)})} \cdot y^{(i)} \\
&= \frac{1}{n} \sum_{i=1}^{n} \frac{f_{Y|X}(y^{(i)})}{q(y^{(i)})} \cdot y^{(i)} \\
&= \frac{1}{n} \sum_{i=1}^{n} \sqrt{2\pi} \cdot \frac{y}{1+y^2}
\end{aligned}$$

## d)

We can get the following result

| Grid Size $n$ | $\mathbb{E}\left[Y \mid X = 0.5\right]$ |
|---|---|
| 50 | 0.31778840599136227 |
| 250 | 0.2931960705649614 |
| 750 | 0.2853944180824812 |
| 1500 | 0.2609568019578496 |
| 3000 | 0.2758034794541073 |

With code

```
using Distributions

sample_sizes = [50, 250, 750, 1500, 3000]

for n in sample_sizes
    samples = Array{Float64}(undef, n)
    wi = Array{Float64}(undef, n)
    i = 1
    while i <= n
        v = rand(1)[1]
        y = rand(Normal(0.5, 1), 1)[1]
        w = sqrt(2 * pi) * y / (1 + y^2)
        wi[i] = sqrt(2 * pi) / (1 + y^2)
        samples[i] = w
        i+=1
    end
    exp = sum(samples) / sum(wi)
    println(exp)
end
```

## 2 Question 2

According to Box Muller Transform, if $x_1$ and $x_2$ are uniform random samples on the interval $(0,1)$, and

$$z_1 = \sqrt{-2\ln x_1}\cos(2\pi x_2), \qquad z_2 = \sqrt{-2lnx_1}sin(2\pi x_2)$$

then $z_1$ and $z_2$ are random samples follows Normal$(0,1)$.

Then we can sample any Normal distribution by the following equation, then we can sample any Normal Distribution.

$$\text{Normal}(\mu, \sigma^2) = \mu + \sigma \cdot \text{Normal}(0,1)$$

As we have

$$f_{Y_1Y_2}(y_1, y_2) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left(-\frac{y_1^2 - 2\rho y_1 y_2 + y_2^2}{2(1-\rho^2)}\right)$$

And

$$f_{Y_1}(y_1) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{y_1^2}{2}), \qquad f_{Y_2}(y_2) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{y_2^2}{2})$$

So

$$f_{Y_1|Y_2=y_2}(y_1) = \frac{f_{Y_1Y_2}(y_1, y_2)}{f_{Y_2}(y_2)} = \frac{1}{\sqrt{2\pi(1-\rho^2)}} \exp(-\frac{(y_1 - \rho y_2)^2}{2(1-\rho^2)}) \sim \text{Normal}(\rho y_2, 1-\rho^2)$$

And similarly

$$f_{Y_2|Y_1=y_1}(y_2) \sim \text{Normal}(\rho y_1, 1-\rho^2)$$

Based on all the things above, we can use a Gibbs Sampling that

$$P(Y_1^t \mid Y_2^{t-1} = y_2) \sim \text{Normal}(\rho y_2, 1-\rho^2)$$
$$P(Y_2^t \mid Y_1^t = y_1) \sim \text{Normal}(\rho y_1, 1-\rho^2)$$

Where $Y_1^0$ and $Y_2^0$ are both sampled from a unifrom distribution.

# 3 Question 3

When the proposal distribution is the corresponding conditional distribution,

## a)

Let

$$u = g(y) = \arctan y \qquad v = h(y) = \arctan y$$

So

$$y = g^{-1}(u) = \tan u \qquad y = h^{-1}(v) = \tan v$$

So we have

$$\begin{aligned}
\int_{-\infty}^{\infty} f_{Y|X} dy &= \int_{-\infty}^{0} f_{Y|X} dy + \int_{0}^{\infty} f_{Y|X} dy \\
&= \int_{-\infty}^{0} \exp(-\frac{(x-y)^2}{2}) \cdot \frac{1}{1+y^2} dy + \int_{0}^{\infty} \exp(-\frac{(x-y)^2}{2}) \cdot \frac{1}{1+y^2} dy \\
&= \int_{-\frac{\pi}{2}}^{0} \exp(-\frac{(x-\tan u)^2}{2}) du + \int_{0}^{\frac{\pi}{2}} \exp(-\frac{(x-\tan v)^2}{2}) dv \\
&\approx \frac{\pi}{2n} \sum_{i=1}^{n} \exp(-\frac{(x-\tan u_i)^2}{2}) + \frac{\pi}{2n} \sum_{i=1}^{n} \exp(-\frac{(x-\tan v_i)^2}{2}) \\
&= \frac{\pi}{2n} \sum_{i=1}^{n} \exp(-\frac{(x-\tan(-\frac{\pi}{2}+\frac{\pi(i-1)}{2n}))^2}{2}) + \frac{\pi}{2n} \sum_{i=1}^{n} \exp(-\frac{(x-\tan(\frac{\pi(i-1)}{2n}))^2}{2})
\end{aligned}$$

Here we define the following, in order to ensure the continuity

$$\tan -\frac{\pi}{2} = -\infty \qquad \text{and} \qquad \tan \frac{\pi}{2} = \infty$$

## b)

So the expectation should be

$$\begin{aligned}
\mathbb{E}[Y \mid X] &= \int_{-\infty}^{\infty} y \cdot f_{Y|X} dy \\
&\approx \frac{\pi}{2n} \sum_{i=1}^{n} \tan u_i \cdot \exp(-\frac{(x-\tan u_i)^2}{2}) + \frac{\pi}{2n} \sum_{i=1}^{n} \tan v_i \cdot \exp(-\frac{(x-\tan v_i)^2}{2})
\end{aligned}$$

Where $u_i$ and $v_i$ are the linspace values dividing $(-\frac{\pi}{2}, 0)$ and $(0, \frac{\pi}{2})$ into $n-1$ parts.

So we can calculate the value by Julia, with the following code,

```

n = [50, 250, 750, 1500, 3000]
for ni in n
    ni = round(Int, ni/2)
    u = range(-1 * pi / 2, length = ni+1, stop = 0)
    v = range(0, length = ni+1, stop = +1 * pi / 2)

    u = collect(u[2:ni+1])
    v = collect(v[1: ni])

    u_tan = map((x)->tan(x), u)
    v_tan = map((x)->tan(x), v)

    u_exp = map((x) -> exp(-((0.5 - x)^2)/2), u_tan)
    v_exp = map((x) -> exp(-((0.5 - x)^2)/2), v_tan)

    inte = pi / (2*ni) * sum(u_exp) + pi/(2*ni) * sum(v_exp)
    c = 1 / inte
    E = c * pi / (2*ni) * sum(u_tan .* u_exp) +
        c * pi/(2*ni) * sum(v_tan .* v_exp)
    println(E)
end
```

And we get the result

| Grid Size $n$ | $\mathbb{E}[Y \mid X = 0.5]$ |
|---|---|
| 50 | 0.2569800409221861 |
| 250 | 0.264284665679843 |
| 750 | 0.2655426364320458 |
| 1500 | 0.2658590025448422 |
| 3000 | 0.2660174684574401 |

## c)

We use the former grid approximation, and apply a direct sampling scheme on it.

```
using Distributions
n = [50, 250, 750, 1500, 3000]

for ni in n
    ni = round(Int, ni/2)
    u = range(-1 * pi / 2, length = ni+1, stop = 0)
    v = range(0, length = ni+1, stop = +1 * pi / 2)
```

```
    weights = Array{Float64, 1}(undef, ni)
    for x in 1:ni
        weights[x] = 1/ni
    end

    u = collect(u[2:ni+1])
    v = collect(v[1: ni])

    m = [100, 1000]

    for mi in m
        um = wsample(u, weights, mi)
        vm = wsample(v, weights, mi)

        u_tan = map((x)->tan(x), um)
        v_tan = map((x)->tan(x), vm)

        u_exp = map((x) -> exp(-((0.5 - x)^2)/2), u_tan)
        v_exp = map((x) -> exp(-((0.5 - x)^2)/2), v_tan)

        inte = pi / (2*ni) * sum(u_exp) + pi/(2*ni) * sum(v_exp)
        c = 1 / inte
        E = c * pi / (2*ni) * sum(u_tan .* u_exp)
                + c * pi/(2*ni) * sum(v_tan .* v_exp)
        println(ni*2, " ", mi, " ", E)
    end
end
```

## d)

From the code above, we get the following results

| Grid Size $n$ | m | $\mathbb{E}[Y \mid X = 0.5]$ |
|---|---|---|
| 50 | 100 | 0.26858755239311605 |
| | 1000 | 0.2430165559582972 |
| 250 | 100 | 0.26221830883887703 |
| | 1000 | 0.2574684284198875 |
| 750 | 100 | 0.24911899520477043 |
| | 1000 | 0.2687289119332303 |
| 1500 | 100 | 0.28066283942057557 |
| | 1000 | 0.2622935268454715 |
| 3000 | 100 | 0.26459162967251004 |
| | 1000 | 0.2716451740382342 |

### e)

We modify the code so that we can use the code directly to calculate the expectation:

```
ni = [50, 250, 750, 1500, 3000]

for n in ni
    x = 0.5
    a = -5
    b = 5
    if n <= 1000
        y_grid = collect(range(a, length=n, stop=b));
        newa = a;
    elseif n > 1000 && n <= 2000
        nm = 1000;
        na=round(Int, (n-nm)/2);
        l = (b-a)/(nm-1);
        newa = a-l*na;
        y_grid = collect(range(newa, step=l , length=n));
    else n > 2000
        nm=round(Int, n/2);
        na=round(Int, (n-nm)/2); l = (b-a)/(nm-1);
        newa = a-l*na;
        y_grid = collect(range(newa, step=l , length=n));
    end
    dy = y_grid[3] - y_grid[2]
    unnormalized_posterior = Array{Float64, 1}(undef, n)
    for i in 1:n
        unnormalized_posterior[i] = y_grid[i]
            * exp(-(x-y_grid[i])^2/2) * (1/(1+y_grid[i]^2)) * dy
    end
    C = 1/1.549
    println(sum(unnormalized_posterior) * C)
end
```

And we get the result

| Grid Size $n$ | $\mathbb{E}[Y \mid X = 0.5]$ |
|---|---|
| 50 | 0.2661755641372646 |
| 250 | 0.26617525294322397 |
| 750 | 0.26617519291542174 |
| 1500 | 0.26617612339045527 |
| 3000 | 0.26617612339069857 |