SOLUTIONS TO EXERCISES 4

Solution 4.1. This follows from the standard Bayes rule:

$$\begin{aligned} p(x|y,z) &= \frac{p(x,y,z)}{p(y,z)}, \\ &= \frac{p(y|x,z)p(x|z)\cancel{p(z)}}{p(y|z)\cancel{p(z)}}, \\ &= \frac{p(y|x,z)p(x|z)}{p(y|z)}. \end{aligned}$$

Solution 4.2. This is just writing down the Bayes theorem and then using the definition of conditional independence: $p(y,x|z) = p(y|z)p(x|z)$

$$p(x|y,z) = \frac{p(y,x|z)}{p(y|z)} = \frac{p(y|z)p(x|z)}{p(y|z)} = p(x|z).$$

Same derivation is true if we swap x and y.

Solution 4.3. For the first step, we know from Example 3.2 that, given

$$\begin{aligned} p(x) &= \mathcal{N}(x;\mu_0,\sigma_0^2), \\ p(y_1|x) &= \mathcal{N}(y;x,\sigma^2) \end{aligned}$$

the posterior density is a Gaussian $p(x|y_1) = \mathcal{N}(x;\mu_1,\sigma_1^2)$, where

$$\begin{aligned} \mu_1 &= \frac{\sigma_0^2 y_1 + \sigma^2\mu_0}{\sigma_0^2+\sigma^2}, \\ \sigma_1^2 &= \frac{\sigma_0^2\sigma^2}{\sigma_0^2+\sigma^2}. \end{aligned}$$

For explicitness, let us show the case for μ_2 and σ_2^2. Using $p(x|y_1)$ as a Gaussian prior, we can update it with $p(y_2|x)$ to obtain the posterior $p(x|y_{1:2})$. We know that

$$p(x|y_{1:2}) = \frac{p(y_2|x)p(x|y_1)}{p(y_2|y_1)}.$$

Applying the same formula above for μ_2 and σ_2^2, we get

$$\begin{aligned} \mu_2 &= \frac{\sigma_1^2 y_2 + \sigma^2\mu_1}{\sigma_1^2+\sigma^2}, \\ \sigma_2^2 &= \frac{\sigma_1^2\sigma^2}{\sigma_1^2+\sigma^2}. \end{aligned}$$

Let us now plug the expressions of μ_1 and σ_1^2 into these update formulas. Let us start with μ_2, by writing

$$\begin{aligned} \mu_2 &= \frac{\sigma_1^2 y_2 + \sigma^2\mu_1}{\sigma_1^2+\sigma^2}, \\ &= \frac{\frac{\sigma_0^2\sigma^2}{\sigma_0^2+\sigma^2} y_2 + \sigma^2 \frac{\sigma_0^2 y_1 + \sigma^2\mu_0}{\sigma_0^2+\sigma^2}}{\frac{\sigma_0^2\sigma^2}{\sigma_0^2+\sigma^2} + \sigma^2}. \end{aligned}$$

At this point, notice that all terms have a common factor σ^2. Cancelling these give

$$\begin{aligned}
\mu_2 &= \frac{\frac{\sigma_0^2}{\sigma_0^2+\sigma^2} y_2 + \frac{\sigma_0^2 y_1 + \sigma^2 \mu_0}{\sigma_0^2+\sigma^2}}{\frac{\sigma_0^2}{\sigma_0^2+\sigma^2} + 1}, \\
&= \frac{\sigma_0^2 y_2 + \sigma_0^2 y_1 + \sigma^2 \mu_0}{\sigma_0^2 + \sigma_0^2 + \sigma^2}, \\
&= \frac{\sigma_0^2 (y_2 + y_1) + \sigma^2 \mu_0}{2\sigma_0^2 + \sigma^2}.
\end{aligned}$$

To derive σ_2^2, we write

$$\begin{aligned}
\sigma_2^2 &= \frac{\sigma_1^2 \sigma^2}{\sigma_1^2 + \sigma^2}, \\
&= \frac{\frac{\sigma_0^2\sigma^2}{\sigma_0^2+\sigma^2} \sigma^2}{\frac{\sigma_0^2\sigma^2}{\sigma_0^2+\sigma^2} + \sigma^2}, \\
&= \frac{\sigma_0^2 \sigma^2}{2\sigma_0^2 + \sigma^2},
\end{aligned}$$

after employing similar simplifications.

We now assume that, as an induction hypothesis, the posterior distribution $p(x|y_{1:n-1})$ is

$$p(x|y_{1:n-1}) = \mathcal{N}(x; \mu_{n-1}, \sigma_{n-1}^2).$$

with

$$\begin{aligned}
\mu_{n-1} &= \frac{\sigma_0^2 \sum_{i=1}^{n-1} y_i + \sigma^2 \mu_0}{(n-1)\sigma_0^2 + \sigma^2}, \\
\sigma_{n-1}^2 &= \frac{\sigma_0^2 \sigma^2}{(n-1)\sigma_0^2 + \sigma^2}.
\end{aligned}$$

Now we have a likelihood $p(y_n|x) = \mathcal{N}(y_n; x, \sigma^2)$ and a prior $p(x|y_{1:n-1})$. By writing the rule again for μ_n and σ_n^2:

$$\begin{aligned}
\mu_n &= \frac{\sigma_{n-1}^2 y_n + \sigma^2 \mu_{n-1}}{\sigma_{n-1}^2 + \sigma^2}, \\
\sigma_n^2 &= \frac{\sigma_{n-1}^2 \sigma^2}{\sigma_{n-1}^2 + \sigma^2},
\end{aligned}$$

we can start deriving these updates. For μ_n,

$$\begin{aligned}
\mu_n &= \frac{\sigma_{n-1}^2 y_n + \sigma^2 \mu_{n-1}}{\sigma_{n-1}^2 + \sigma^2}, \\
&= \frac{\frac{\sigma_0^2\sigma^2}{(n-1)\sigma_0^2+\sigma^2} y_n + \sigma^2 \frac{\sigma_0^2 \sum_{i=1}^{n-1} y_i + \sigma^2 \mu_0}{(n-1)\sigma_0^2+\sigma^2}}{\frac{\sigma_0^2\sigma^2}{(n-1)\sigma_0^2+\sigma^2} + \sigma^2}.
\end{aligned}$$

Simply the same way above, we can cancel σ^2 factors and get

$$\mu_n = \frac{\sigma_0^2 \sum_{i=1}^n y_i + \sigma^2 \mu_0}{n\sigma_0^2 + \sigma^2}.$$

For the variance, we write

$$\begin{aligned}
\sigma_n^2 &= \frac{\sigma_{n-1}^2 \sigma^2}{\sigma_{n-1}^2 + \sigma^2}, \\
&= \frac{\frac{\sigma_0^2\sigma^2}{(n-1)\sigma_0^2+\sigma^2}\sigma^2}{\frac{\sigma_0^2\sigma^2}{(n-1)\sigma_0^2+\sigma^2} + \sigma^2}, \\
&= \frac{\sigma_0^2\sigma^2}{n\sigma_0^2 + \sigma^2}.
\end{aligned}$$

Since we proved this case for $n = 2$ and we have shown that the induction hypothesis holds for $n - 1$, we have proved the result for all n.

Solution 4.4. We want to estimate

$$\lambda = \mathbb{P}(0 < X < 2) = \int_0^2 \frac{1}{\pi(1+x^2)} \mathrm{d}x.$$

We can choose a uniform on $[0, 2]$, i.e.,

$$p(x) = \frac{1}{2} \qquad \text{for } x \in (0, 2),$$

and

$$\varphi(x) = \frac{2}{\pi(1+x^2)}.$$

The variance of the Monte Carlo estimate is given by

$$\begin{aligned}
\frac{\mathsf{var}_p(\varphi)}{N} &= \frac{1}{N}\left(\mathbb{E}[\varphi^2(X)] - \mathbb{E}[\varphi(X)]^2\right) \\
&= \frac{1}{N}\left(\frac{4}{\pi^2}\frac{1}{2}\int_0^2 \frac{1}{(1+x^2)^2}\mathrm{d}x - (\frac{1}{2} - 0.1476)^2\right),
\end{aligned}$$

where the last line uses $\lambda = \frac{1}{2} - I$ and note $\mathbb{E}[\varphi(X)] = \lambda$. Computing this, we arrive at

$$\frac{\mathsf{var}_p(\varphi)}{N} = \frac{0.0285}{N},$$

almost half of the best exampled discussed during the course.

The variance is this estimate determines the variance of I, of course, as they are connected up to a constant and addition/substraction of constants will not change variance.

Solution 4.5. The code is given as follows.

```
import numpy as np
import matplotlib.pyplot as plt

def phi(x):
```

```
    return np.sqrt((1 - x**2))

I = np.pi / 4

N_max = 100000

U = np.random.uniform(0, 1, N_max)
I_est = np.zeros(N_max - 1)
I_var = np.zeros(N_max - 1)
I_var_correct = np.zeros(N_max - 1)

fig = plt.figure(figsize=(10, 5))

K = np.array([])
k = 0

for N in range(1, N_max, 1):
    # print(N)

    I_est[k] = (1/N) * np.sum(phi(U[0:N]))
    I_var[k] = (1/(N**2)) * np.sum((phi(U[0:N]) - I_est[k])**2)

    k = k + 1

    K = np.append(K, N)

    if (N-1) % 200 == 0:
        plt.clf()
        plt.semilogx(K, I_est[0:k], 'k-', label='MC estimate')
        plt.plot(K, I_est[0:k] + np.sqrt(I_var[0:k]), 'r', label='${\\
                                                    sigma}_{\\varphi,N}$',
                                                    alpha=1)
        plt.plot(K, I_est[0:k] - np.sqrt(I_var[0:k]), 'r', alpha=1)
        plt.plot([0, N_max], [I, I], 'b--', label='True Value', alpha=
                                                    1, linewidth=2)
        plt.legend()
        plt.xlabel('Number of samples')
        plt.ylabel('Estimate')
        plt.xlim([0, N_max])
        plt.show(block=False)
        plt.pause(0.01)
```