

Exercise 1

Junsong Tang

junsong.tang@stat.ubc.ca

1 sampling from a joint distribution

1. Since flips are independent, then the joint pmf: $p(x, y_1, y_2, y_3, y_4) = p(y_1, y_2, y_3, y_4 | x) \cdot p(x) = \prod_{i=1}^4 p(y_i | x) \cdot p(x)$.

Let $g(X, Y_1, Y_2, Y_3, Y_4) = (1 + Y_1)^X$, then by LOTUS,

$$\begin{aligned}
 \mathbb{E}(1 + Y_1)^X &= \mathbb{E}g(X, Y_1, Y_2, Y_3, Y_4) \\
 &= \sum_{x \in \{0,1,2\}} \sum_{y_1 \in \{0,1\}} \sum_{y_2} \sum_{y_3} \sum_{y_4} g(x, y_1, y_2, y_3, y_4) \cdot p(x, y_1, y_2, y_3, y_4) \\
 &= \sum_x \sum_{y_1} (1 + y_1)^x \cdot p(y_1 | x) \cdot p(x) \sum_{y_2} p(y_2 | x) \sum_{y_3} p(y_3 | x) \sum_{y_4} p(y_4 | x) \\
 &= \sum_x \sum_{y_1} (1 + y_1)^x \cdot p(y_1 | x) \cdot p(x) \\
 &= \frac{13}{6}
 \end{aligned}$$

```

21 library(extraDistr)
2  set.seed(2024)
3  # theoretical expectation
4  X = c(0,1,2)
5  Y = c(0,1)
6  p_mtx = matrix(c(1,1/2,0,0,1/2,1), byrow = FALSE, nrow=3)
7  sum = 0
8  for (i in (1:length(X))) {
9    x = X[i]
10   for (j in (1:length(Y))) {
11     y = Y[j]
12     p = p_mtx[i,j]
13     sum = sum + (1+y)^x * p * (1/3)
14   }
15 }
16 sum #2.166666666666667
17
18 # sample function
19 forward_sample = function() {
20   x = rdunif(1, 0, 2)
21   Y = rbinom(4, 1, x/2)
22   return (c(x, Y))
23 }
24 forward_sample()
25 # 1 1 0 1 0
26
27 # function of (x, Y)
28 f_eval = function(v) ((1 + v[2])^v[1])
29
30 # compute expectation
31 mean(replicate(f_eval(forward_sample()), n = 100000))
32 # 2.16713
33

```

3. we use the sample mean of the function values of $g = (1 + y_1)^x$ to approximate the expectation by the LLN
4. From the simulation results, if the number of simulation gets larger, then the sample mean is closer to the true expectation.

2 computing a conditional

1. If $p = \frac{1}{2}$, then $X = 1$, so we want: $\mathbb{P}(X = 1 | (Y_1, Y_2, Y_3, Y_4) = (0, 0, 0, 0))$
2. Note that

$$\mathbb{P}(Y_i = 0, \forall i | X = 0) = 1$$

and

$$\mathbb{P}(Y_i = 0, \forall i | X = 1) = \left(\frac{1}{2}\right)^4$$

and

$$\mathbb{P}(Y_i = 0, \forall i | X = 2) = 0$$

, so by Bayes rule and total probability:

$$\begin{aligned} & \mathbb{P}(X = 1 | (Y_1, Y_2, Y_3, Y_4) = (0, 0, 0, 0)) \\ &= \frac{\mathbb{P}(Y_i = 0, \forall i | X = 1) \cdot \mathbb{P}(X = 1)}{\sum_{j=0}^2 \mathbb{P}(Y_i = 0, \forall i | X = j) \cdot \mathbb{P}(X = j)} \\ &= \frac{\left(\frac{1}{2}\right)^4 \cdot \frac{1}{3}}{1 \cdot \frac{1}{3} + \left(\frac{1}{2}\right)^4 \cdot \frac{1}{3} + 0 \cdot \frac{1}{3}} \\ &= \frac{1}{17} \end{aligned}$$

3 non uniform prior on coin types

- 1.

$$\begin{aligned} X &\sim \text{Categorical}((0, 1, 2) | (\frac{1}{100}, \frac{98}{100}, \frac{1}{100})) \\ Y_i | X &\sim \text{Bernoulli}(\frac{X}{2}) \end{aligned}$$

- 2.

$$\begin{aligned} & \mathbb{P}(X = 1 | (Y_1, Y_2, Y_3, Y_4) = (0, 0, 0, 0)) \\ &= \frac{\mathbb{P}(Y_i = 0, \forall i | X = 1) \cdot \mathbb{P}(X = 1)}{\sum_{j=0}^2 \mathbb{P}(Y_i = 0, \forall i | X = j) \cdot \mathbb{P}(X = j)} \\ &= \frac{\left(\frac{1}{2}\right)^4 \cdot \frac{98}{100}}{1 \cdot \frac{1}{100} + \left(\frac{1}{2}\right)^4 \cdot \frac{98}{100} + 0 \cdot \frac{1}{100}} \\ &= \frac{98}{114} \end{aligned}$$

4 a first posterior inference algorithm

```

1_ posterior_given_four_heads = function(rho) {
2_   K = length(rho)
3_   posterior = NULL
4_   sum = 0

```

```

5   for (k in (1 : K)) {
6     p_k = (1-(k-1)/(K-1))^4 * rho[k] #joint p(Y, X = k)
7     sum = sum + p_k
8     posterior = c(posterior, p_k)
9   }
10  # normalizing
11  posterior = posterior / sum
12  return (posterior)
13 }
14
15
16

```

```

21 posterior_given_four_heads(c(1/100,98/100,1/100))[2] - 98/114
2   # 0
3

```

```

31 rho = seq(1,10,1)
2   rho = rho / sum(rho)
3   posterior_given_four_heads(rho)
4   # 0.201845869866174, 0.252022765728349, 0.221596677434241, 0.159483156437471,
5   # 0.0961390555299185, 0.0472542685740655, 0.0174434702353484, 0.00393785571450546,
6   # 0.000276880479926166, 0

```

From the posterior distribution, we can infer that when we have four heads: (0,0,0,0), then it is most likely that $X = 2$. i.e. type 2 coin.

5

1.

$$X \sim$$

```

21 posterior = function(rho, n_heads, n_observations) {
2   posterior = NULL
3   K = length(rho) - 1
4   # normalize rho
5   rho = rho / sum(rho)
6   sum = 0
7   for (k in (0:K)) {
8     p_k = dbinom(n_heads, n_observations, 1-k/K) * rho[k+1]
9     sum = sum + p_k
10    posterior = c(posterior, p_k)
11  }
12  # normalizing
13  posterior = posterior / sum
14  return (posterior)
15 }
16

```

```

31 rho = seq(from=1, to=10, by=1)
2   n_heads = 2
3   n_observations = 10
4   posterior(rho, n_heads, n_observations)
5

```